INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2020IPPAT031

Thèse de doctorat

Similarity Ranking for Biometrics: Theory and Practice

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Télécom Paris

École doctorale n°574 mathématiques Hadamard (EDMH)
Spécialité de doctorat : Mathématiques appliquées

Thèse présentée et soutenue à Palaiseau, le 9 octobre 2020, par

ROBIN VOGEL

Composition du Jury :

Florence d'Alché-Buc Professeure, Télécom Paris	Présidente
Marc Sebban Professeur, Université Jean Monnet	Rapporteur
Robert C. Williamson Professor, Australian National University	Rapporteur
Odalric-Ambrym Maillard Chargé de recherche, Inria	Examinateur
Isabel Valera Research scientist, Max-Planck Institute	Examinatrice
Stephan Clémençon Professeur, Télécom Paris	Directeur de thèse
Aurélien Bellet Chargé de recherche, Inria	Co-directeur de thèse
Stéphane Gentric Research unit manager, IDEMIA	Invité
Vincent Despiegel Research team leader, IDEMIA	Invité

Contents

List of Figures

List of Tables

List of Publications

- Learning Fair Scoring Functions:
  Fairness Definitions, Algorithms and Generalization Bounds for Bipartite Ranking.
  Authors: Robin Vogel, Aurélien Bellet, Stephan Clémençon.
  (preprint)

- A Multiclass Classification Approach to Label Ranking.
  Authors: Stephan Clémençon, Robin Vogel.
  (AISTATS 2020)

- Weighted Empirical Risk Minimization:
  Sample Selection Bias Correction based on Importance Sampling.
  Authors: Robin Vogel, Mastane Achab, Stephan Clémençon, Charles Tiller.
  (ESANN 2020)

- Weighted Empirical Risk Minimization:
  Transfer Learning based on Importance Sampling.
  Authors: Robin Vogel, Mastane Achab, Stephan Clémençon, Charles Tiller.
  (ICMA 2020)

- Trade-offs in Large-Scale Distributed Tuplewise Estimation and Learning.
  Robin Vogel, Aurélien Bellet, Stephan Clémençon, Ons Jelassi, Guillaume Papa.
  (ECML PKDD 2019)

- On Tree-based Methods for Similarity Learning.
  Authors: Stephan Clémençon and Robin Vogel.
  (LOD 2019)

- A Probabilistic Theory of Supervised Similarity Learning for Pointwise ROC Curve Optimization.
  Authors: Robin Vogel, Aurélien Bellet and Stephan Clémençon.
  (ICML 2018)

Remerciements

Je tiens ici à remercier tous ceux qui ont contribué au succès de ce projet doctoral.

Mes premiers remerciements vont à mon maître de thèse, Stephan, qui m'a guidé dans mes premiers pas dans le monde de la recherche. Les compétences que j'ai acquises pendant ces quatre années à Télécom Paris, ainsi que la qualité du travail de recherche effectué, sont en grande partie le fruit de son enseignement, son soutien et sa direction. Je te remercie particulièrement pour ta disponibilité au quotidien, ainsi que pour ta présence à des moments inoubliables de ma thèse.

Je remercie aussi tout particulièrement Aurélien, qui malgré la distance qui nous sépare a suivi de très près ce projet doctoral, et a été une force motrice essentielle de ce projet. Je ne peux qu'admirer ta capacité et ta détermination à toujours réévaluer avec justesse le travail accompli, et j'espère emporter cette aptitude avec moi pour mes projets futurs. Je n'ai aucun doute qu'elle sera promesse de succès, et j'espère l'appliquer dès que possible sous ton œil averti.

Je remercie Stéphane (avec un e), qui a assuré la continuité du suivi industriel du projet. Par son dynamisme, il a porté la dimension industrielle de ma thèse, et m'a enseigné des compétences professionnelles qui ne s'apprennent pas dans les livres, qui me seront sûrement très utiles à l'avenir. Je dois aussi saluer ta curiosité et ta constance, qui ont été le moteur de nombreuses réunions de suivi.

Finalement, je tiens à remercier formellement mes supérieurs immédiats en entreprise, dans l'ordre chronologique: Julien Bohné, Anouar Mellakh et Vincent ici présent. Merci Julien pour ton enthousiasme, et pour m'avoir mis le pied à l'étrier dans ma première expérience en apprentissage profond dans l'industrie. Merci Anouar pour ta présence rassurante et tes conseils. Vincent, je te remercie pour ta bienveillance et ta curiosité, qui je le crois contribuent à créer un environnement propice au bon déroulement d'une thèse CIFRE. Je vous souhaite à tous les trois une très bonne continuation dans vos projets professionnels futurs.

Je tiens aussi à remercier Anne Sabourin, pour m'avoir enseigné une certaine rigueur scientifique, que ce soit en réunion de suivi de thèse, ou en donnant de ton temps aux tableaux blancs de Rue Barrault. Je ne peux pas nier ta contribution à ma thèse, et à ma formation.

I want to thank Marc Sebban and Robert Williamson, for having reviewed the thesis. Their advice contributed to the quality of the final version of the manuscript, and to the permanent improvement of my writing skills. I also thank Florence d'Alché Buc, for teaching me machine learning in the classroom five years ago, as well as for accepting to be the president of my thesis jury today. Last, but not least, I want to thank Oldaric-Ambrym Maillard and Isabel Valera for accepting to join my jury.

Merci aux anciens doctorants (Anna, Albert, Igor, Guillaume P., Nicolas, ...) du labo, qui m'auront beaucoup aidé directement en début de thèse, et qui ont ensuite souvent incarné un exemple à suivre.

Merci aux doctorants de "ma promotion" (Hamid, Pierre L., ...) pour ces débats animés sur des sujets variés tels que le sport, les sujets de société, la mode, en passant par les mathématiques. Je remercie particulièrement Mastane, sans qui l'école d'été à Buenos Aires n'aurait pas aussi été ce *road-trip* légendaire. Je retiens ton invitation dans cette ville hispanophone un peu plus

animée que Neuquen.

Merci aux nouveaux doctorants (Pierre C., Émile (not on mute), Kimia, Amaury, Guillaume, Anas, Myrto, ...) du labo, sans qui ma fin de thèse n'aurait pas été la même. Malgré la distance qui nous sépare du point zéro, je suis convaincu qu'entre ces murs nous sommes au bon endroit.

Merci à mes collègues en entreprise, que ce soit les nouveaux (Baptiste, Arnaud, Damien, Dora, ...) et les moins nouveaux (Emine, Jonathan, Éric, ...). Je serais toujours partant pour un petit afterwork à la Défense. Je n'oublie bien sûr pas Richard, mon "camarade de CIFRE", auquel je souhaite bon courage pour la dernière ligne droite.

Merci à mes amis passés par la thèse (Yannick, Guillaume C., ...), ainsi qu'aux doctorants des quatre coins du monde rencontrés en école d'été. Leur authenticité m'a permis de mettre ma propre expérience en perspective dans les moments difficiles. Leur enthousiasme a été communicatif.

Je tiens à remercier ma famille et l'ensemble de mes proches, pour leur soutien durant ma thèse, et plus généralement durant l'ensemble de mon aventure francilienne.

Chapter 1

Summary

1.1 Context

The thesis originates from a collaboration between the french *grande école* Télécom Paris and the multinational company IDEMIA. Precisely, the project relies on a CIFRE contract (Industrial Agreements for Training through Research), a type of contract introduced in 1981 by the french government to strengthen the ties between research institutions and private companies. The research work is thus supervised by both parties, which is made possible in our case by frequent interactions.

Télécom Paris is one of the top French public institutions of higher education and research in engineering in France, and is a member of the *Institut Mines-Télécom* (IMT) and the *Institut Polytechnique de Paris* (IP Paris). The IP Paris is a public higher education and research institution, that brings together five prestigious French engineering schools: École Polytechnique, ENSTA Paris, ENSAE Paris, Télécom Paris and Télécom SudParis. Under the auspices of the Institute, they share their expertise to develop training programs of excellence and cutting-edge research. The research involved in this thesis was done within the Signal, Statistics and Learning (S2A) team of the Information Processing and Communication Laboratory (LTCI). The academical supervision team consisted of Stephan Clémençon and Anne Sabourin, both members of the S2A team, as well as Aurélien Bellet, researcher at INRIA (National Institute for Research in Digital Science and Technology).

IDEMIA is the leading company in biometric identification and security, as well as secure payments. The company is a merger of the companies Morpho and Oberthur Technologies, which was achieved in 2017. Oberthur technologies was a dominant player in digital security solutions for the mobile world, while Morpho was considered the worldwide leader of biometric recognition. IDEMIA has the aim of converging technologies developed for the public sector (by the former Morpho) and those for the private sector (by Oberthur Technologies). In the private sector, the major customers of the company come from banking, telecom and connected objects. The thesis started in 2017 with Safran Identity and Security (formerly Morpho) before the merger, back when Morpho was a subsidiary of the large aeronautics and defense company Safran. Throughout the thesis, Stéphane Gentric assumed the continued industrial supervision of this project. The managers of the Advanced Machine Learning team: sequentially Julien Bohné, Anouar Mellakh and Vincent Despiegel, contributed significantly to that supervision.

1.2 Introduction

Biometrics is the discipline of distinguishing individuals based on physical or behavioral attributes such as fingerprints, face, irises and voice. In the modern world, biometrics has many essential applications, such as border crossing, electronic commerce and welfare disbursement. While its mainstream usage is recent, the discipline is not new. Indeed, in the late 19th century, the French law enforcement officer Alphonse Bertillon proposed a personal identification system based on the measurement of specific bony parts of the body (Jain et al., 2011).

Today, the most widespread biometric measurement is fingerprint recognition, followed by face and iris recognition. All of them rely on the acquisition of images of specific parts the body. Hence, while biometrics is seen by many authors as a distinct field of science, its history and development is tightly related with that of *computer vision*, the interdisciplinary scientific field that seeks to enable computers to gain high-level understanding of digital images.

In the early 2010s, the performance of computer vision systems started to dramatically improve (Goodfellow et al., 2016), due to the development of general-purpose computing on graphical processing units (GPGPU). It enabled the widespread adoption of neural networks — statistical models composed of layers that summarize information — as their training strongly benefits from very fast matrix multiplication. Training neural networks consists in finding the parameters of the network that minimize a loss function with gradient descent algorithms. Those modify iteratively the network parameters, by adding a small quantity that is negatively proportional to the gradient at each step. The growing interest for neural networks has fueled a huge corpus of literature. Most papers propose a better architecture for the model, suggest an improvement of the optimization method or introduce a better loss function for a particular use case.

Recent literature has proposed many loss functions for biometrics, built on the intuition that a more stringent separation of identities in an embedding space leads to improvements in performance (Wang and Deng, 2018). The flagship problem of biometrics is 1:1 verification, which seeks to verify the identity of an individual by comparing a live measurement with reference data using a similarity measure. For example, the entry of an individual to a restricted area can require the conformity of the measurement with a personal identification card. Performance in 1:1 verification is evaluated using the ROC curve, a functional criterion that summarizes the quality of a similarity measure. For a set of tests, the ROC curve gives all false acceptance and false rejection rates attainable by thresholding the similarity. It is the golden standard for evaluating score functions. In the thesis, we argue for using the bipartite ranking literature to design loss functions for 1:1 verification, which is seen as scoring on pairs of observations. While both scoring and pairwise learning are addressed in the literature, their simultaneous examination is new and raises distinct challenges.

The recent dramatic improvements in accuracy of many machine learning applications foreshadow the emergence of new markets, issued from the maturation of formerly very experimental technologies. One such market is facial recognition, which has recorded, and is expected to maintain, exponential growth. Its development brought media coverage about the potential misuses and systemic biases of the technology, on top of the usual privacy concerns. In that context, practitioners and governmental agencies have recorded differences in accuracy between ethnicities in face recognition (Grother and Ngan, 2019). One common explanation is that available face databases for training face recognition systems fail to represent the general population. The performance discrepancy raises the broader issue of fairness, a common concern with automated decisions that has received increasing attention in the recent machine learning literature (Barocas et al., 2019). Some observers commented that predictive algorithms run the risk of being merely "opinions embedded in mathematics". In their view, practitioners should not focus on predictive performance alone, but also enforce the conformity of their system with a set of moral values. Biometrics is also concerned with fairness. Indeed, even when the representativeness of the training database in terms of gender is accounted for, women still account for more false positives than men, possibly due to societal norms regarding appearance. This may lead to systemic discrimination, notably when considering systems that flag people of interest. In the thesis, we first propose to correct biases using importance weights, which adjusts the distribution of the training data. The correction covers several cases where train and test data do not match, and assumes the availability of auxiliary information about their relationship. The level of generality that we propose is new, and covers many important applications in biometrics. Then, we propose to modify loss functions to incorporate explicitly fairness considerations when learning a score for bipartite ranking. While fairness in classification has received a lot of attention, fair bipartite ranking has not. Considering our scoring perspective on 1:1 verification, this work is an intermediary step to explicitly incorporate fairness constraints in this biometric problem.

In general, many issues from the machine learning literature arise simultaneously in the design of biometric systems. The objective of this thesis is to identify and address several of those from the perspective of statistical machine learning, in the hope of providing security guarantees under probabilistic assumptions, and proposing sensible solutions to biometric systems manufacturers.

Notation	Description
w.r.t.	With respect to
s.t.	Subject to
r.h.s.	Right-hand side
l.h.s.	Left-hand side
a.s.	Almost-surely
r.v.	Random variable
i.i.d.	Independent and identically distributed
c.d.f.	Cumulative distribution function
p.d.f.	Probability density function
$:=$	Definition of a variable
$\forall$	Universal quantifier
$\mathcal{X} \to \mathcal{Y}$	Map of $\mathcal{X}$ to $\mathcal{Y}$
$A^\top$	Transpose of the matrix A
$\varnothing$	Empty set
$\binom{n}{k}$	Binomial coefficient
$\mathfrak{S}_n$	Permutation of $\{1, \dots, n\}$
$A \cup B$ (resp. $A \cap B$)	Set union (resp. intersection) between the sets A and B
$A \Delta B$	Symmetric difference between the sets A and B
$\#A$	Cardinal of the set A
A^c	Complementary of a set A
$\mathcal{P}(A)$	Set of all parts of a set A
$\subset$	Set inclusion
$\mathbb{I}\{\cdot\}$	Indicator function
$\mathrm{Im}(f)$	Image of the function f
$\mathrm{sgn}(\cdot)$	Sign function, $\mathrm{sgn}(x) = 2\mathbb{I}\{x \geqslant 0\} - 1$
$\log(\cdot)$	Natural logarithm function
$O(\cdot)$	"Big O": Asymptotic order of a quantity
$\mathbb{P}[\cdot]$	Probability of event
$\mathbb{E}[\cdot]$	Expectation of a random variable
$\mathrm{supp}(\mu)$	Support of the distribution μ
$X \sim \mu$	The *r.v.* X follows the distribution μ
$\mu \otimes \nu$	Product measure of μ and ν
δ_x	Dirac mass at point x
$\bar{F}$	Survival function for *c.d.f.* F, $\bar{F} = 1 - F$
F^{-1}	Generalized inverse of a càdlàg function
$\mathbb{N}$ (resp. $\mathbb{R}$)	Natural (resp. real) numbers
$\mathbb{N}^* = \mathbb{N}\{0\}, \quad \mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geqslant 0\}, \quad \mathbb{R}_+^* = \mathbb{R}_+ \backslash \{0\}$	

Table 1.1: Summary of notations.

In that regard, the usual statistical machine learning literature deals with simple problems, such as classification or regression (Boucheron et al., 2005). However, the problems tackled in biometrics involve both pairwise learning and a functional criterion, and thus require a specific analysis.

Chapter outline. The current chapter summarizes the contributions of the main parts II and III of the thesis, and skips the remaining part (Part I). Part I is in the thesis for clarity reasons and only contains technical preliminaries to the theoretical results of the other parts. The current chapter is organized as follows: firstly, Section 1.3 extends on the ideas presented in the short introduction above and provides a detailed outline for the thesis. Secondly, Section 1.4 focuses on Part II and discusses similarity ranking. Thirdly, Section 1.5 summarizes the contributions of Part III on the broad topic of reliable machine learning. Finally, Section 1.6 details the perspectives of the thesis.

Section 1.3 is a developed introduction that focuses on important aspects of biometrics that are addressed in the machine learning literature. More precisely, it first presents the relationship between biometrics and metric learning, as well as the impact of deep learning on both fields.

It then delves into the nature of bias in facial recognition, and details potential dangers of bias through the lens of algorithmic fairness.

Section 1.4 is a short summary of Part II. It focuses on the idea of considering similarity learning as scoring on a product space. We name this view *similarity ranking*. In that regard, it begins with theoretical guarantees for that problem. Then, it proposes strategies with guaranteed statistical accuracy to reduce the computational complexity of similarity ranking, and ends with several practical, gradient-based, approaches.

Section 1.5 is a short summary of Part III, which deals with reliability in machine learning. As such, it first proposes a strategy to predict an ordered list of probable classes from multiclass classification data, instead of focusing on classification precision alone. Then, it gives strategies to deal with the lack of representativeness of databases. It ends with a proposal to enforce fairness for the scoring problem.

Section 1.6 presents the perspectives of the thesis. Specifically, it stresses the importance of practical illustrations of this work for biometric practitioners, and discusses several possible extensions of our analyses.

The notations adopted throughout the thesis are summarized in Table 12.1.

1.3 Recent Challenges in Biometrics

The recent advances in deep learning have brought rapid changes to the state-of-the-art in biometrics. In that regard, the new subfield of deep learning metric has proposed many derivable losses, but none of them relate to the ranking-based evaluation of biometrics systems. At the same time, the advances have stirred public debate about biometric technologies, and in particular on face recognition. While most issues concern the usage of facial recognition algorithms, one is its recently-measured racial bias. The machine learning research community can propose technical solutions for that problem.

1.3.1 Introduction to Biometrics

Societal value. Biometrics responds to the need to establish the identity of a person with high confidence. It has become crucial in the modern world, since one interact with an always increasing number of people. However, its roots can be traced as early as the late 19th century, with the first recorded uses of fingerprints for identification (Jain et al., 2011, Section 1.8). Today, biometrics are widely used in forensics and other governmental applications, as well as by various industries, such as the banking sector. One example of large-scale biometrics application is the Aadhaar project, managed by the Unique IDentification Authority of India (UIDAI), which has assigned national identification numbers and recorded the biometrics (irises, faces and fingerprints) of over one billion people (Jain et al., 2011, Section 1.6).

Formal objectives. The objective of biometric systems is to compare two measurements (x, x') in an input space $\mathcal{X}$, *e.g.* two fingerprints or two faces, and to decide whether both of them originate from the same individual. It is generally done by means of a pairwise similarity function $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$, that quantifies the likelihood that x and x' originate from the same person. The decision is taken by thresholding the similarity value, *i.e.* the pair (x, x') is a match if $s(x, x') > t$, where t is a threshold in $\mathbb{R}_+$. Two flagship problems can be identified in biometrics: verification and identification (Jain et al., 2011, Section 1.3). The verification problem is also referred to as 1:1 matching or 1:1 authentication. It is illustrated by the use case of border crossing, where an official compares a document x' with a live measurement x. As such, it consists in making a decision on a pair (x, x'). On the other hand, identification is illustrated by the use case of automatic surveillance, where a live measurement x is compared to a database. Precisely, it consists in finding the existence of a match to x in a database of $N \in \mathbb{N}$ observations $\mathcal{D}_N = \{x_i\}_{i=1}^N \subset \mathcal{X}$. If a match exists, it needs to return the relevant elements in $\mathcal{D}_N$. Identification is also referred to as 1:N matching or 1:N authentication. The number of enlisted people N can be large, for example in the millions.

Operational steps. To compare an observation x with a large database $\mathcal{D}_N$, one needs the

capacity to match quickly low-memory representations of elements in $\mathcal{X}$. It requires the derivation of efficient intermediary representations of the input data. Biometric systems can usually be split in three distinct processes: 1) the acquisition of the input data, called *enrollment*, 2) the feature extraction, sometimes referred to as the *encoding* of the data, 3) and the matching of the encodings. See Jain et al. (2011) (Section 1.2) for more details. In the context of fingerprint recognition, the enrollment phase covers the acquisition of the raw input, post-processing steps, as well as quality verifications on the final image. Feature extraction consists in applying usual computer vision techniques, followed by domain-specific techniques to extract specific points in the fingerprint image. For example, *Gabor filters* (Szeliski, 2011, Section 3.4.1) are used to derive *ridge orientation maps* (Jain et al., 2011, Chapter 2) from raw fingerprint images. Characteristic points called *minutiae* are then extracted from that intermediate representation. Finally, matching relies on the evaluation of a distance between the point clouds of two images. We refer to Section 2 of Jain et al. (2011) for more details.

Feature extraction. The module that has received the most attention in biometrics research is the feature extraction module. For example, research in automatic fingerprint recognition has dedicated many man-hours to finding as much discriminative information as possible in the images. In the context of facial recognition, the feature extraction part was first based on EigenFaces (Turk and Pentland, 1991), an application of the principal component analysis (PCA) to natural images of faces. Then, it combined common computer vision descriptors — such as Local Binary Patterns (LBP) and Scale-Invariant Feature Transform (SIFT) image descriptors — with dimensionality reduction techniques. Finally, it now relies on end-to-end — models that perform a task using raw input — deep convolutional neural networks, a type of neural network well suited for images (Wang and Deng, 2018).

Metric learning algorithms serve to train the feature extraction module. Recently, the advent of deep learning algorithms has completely shifted the focus of biometric researchers from a combination of domain-specific feature extraction and linear metric learning, to end-to-end deep metric learning. Researchers in biometrics thus have to follow closely recent developments in machine learning — and especially deep learning — to stay competitive.

1.3.2 Deep Metric Learning for Biometrics

Metric learning or similarity learning is the machine learning problem that that seeks to learns how similar two objects are. We refer to Bellet et al. (2015a) for a survey. In biometrics, the supervision for those algorithms originates from a database of n images $\{x_i\}_{i=1}^n \subset \mathcal{X}$, with each image x_i having an associated identity $y_i \in \{1, \dots, K\}$ with $K \leqslant n$ and $(n, K) \in \mathbb{N}^2$.

Linear metric learning. The first metric learning algorithms and a large part of the literature focus on *linear metric learning*. It refers to distances or similarity functions $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ that are linear functions of their inputs. Those mainly rely on the use of a *Mahalanobis distance* — a distance function $\mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$ for $d \in \mathbb{N}$ parameterized by a positive semidefinite matrix $M \in \mathbb{R}^{d \times d}$ — with a few other linear metric learning methods using other combinations of M and the input x, x'. The Mahalanobis distance d_M between the points x and x' satisfies:

$$d_M(x, x') = \sqrt{(x - x')^\top M (x - x')}.$$

The Cholesky decomposition (Petersen and Pedersen, 2008, Section 5.5.1) implies that one can write $M = LL^\top$, where L is a lower triangular matrix. This result justifies seeing Mahalanobis distances as computing a simple Euclidean distance over a transformation of the inputs, since $d_M(x, x') = \|Lx - Lx'\|_2$, where $\|\cdot\|_2$ is the standard Euclidean distance. Notable approaches for learning Mahalanobis distances include the Mahalanobis Metric for Clustering (MMC) algorithm in 2002 (Xing et al., 2002), the Neighborhood Component Analysis (NCA) algorithm in 2004 (Goldberger et al., 2004), and the Large Margin Nearest Neighbor (LMNN) algorithm in 2009 (Weinberger and Saul, 2009). Several authors have considered the extension of linear metric learning algorithms. For instance, they proposed the kernelization of linear metric learning methods, as well as the use of local linear metrics (Bellet et al., 2015a, Section 5). These extensions turned out to be useful for facial recognition practitioners. For exemple, Bohné et al. (2014) considered applying the algorithm MM-LMNN (Weinberger and Saul, 2009), which learns a local linear metric, to facial recognition.

Deep metric learning (DML). Due to the development of general-purpose computing on graphical processing units (GPGPU), the training and use of very deep neural networks became practical. It has had the effect of dramatically improving the performance of computer vision, notably for the task of large-scale classification on the challenge ILSVRC (ImageNet Large Scale Visual Recognition Challenge). The most salient breakthrough for that challenge happened in 2012 and is presented in Krizhevsky et al. (2012). Results with deep networks led to important advances in face recognition as early as 2014 (Taigman et al., 2014). Deep facial recognition encodes raw data x to a vector $e(x)$, where $e : \mathcal{X} \to \mathbb{R}^d$ is a non-linear function and corresponds to the output of a neural network. Then, a simple distance is computed between $e(x)$ and $e(x')$ to decide whether x and x' match. The encoding e is optimized by gradient descent to minimize a loss function. On the contrary to other popular biometries such as fingerprints and irises, finding manually the important distinctive features of a face is difficult. Therefore, an end-to-end gradient-based approach is very well-suited to face recognition. Still, authors have already proposed deep metric learning approaches for other biometries (Minaee et al., 2019).

Loss functions for DML. Deep metric learning has replaced the sequence of hand-crafted features and dimensionality reduction by end-to-end models, as illustrated in the very influential paper of Schroff et al. (2015). Therefore, a lot of research focuses on better network architectures. Simultaneously, the advent of gradient-based learning has paved the way for extensive research in the design of loss functions. Early facial recognition systems used the usual softmax cross-entropy loss, an usual classification loss in deep learning that seeks to separates identities (Goodfellow et al., 2016, Section 3.13). Since then, many other loss functions have been proposed, such as ArcFace (Deng et al., 2019). We refer to Figure 5 of Wang and Deng (2018) for an overview of losses for facial recognition. Their objective is either to: increase the margin between the identities to increase inter-group variance, group all observations of each identity together to decrease intra-group variance, or combine both approaches as does the triplet loss (Schroff et al., 2015). Practicioners have reported that summing different losses, while adjusting the proportion of each loss was necessary to optimize for performance (Parkhi et al., 2015).

Ranking-based evaluation. The performance of facial recognition systems is measured on the ROC curve, as shown in the evaluations of commercial face recognition systems by Grother and Ngan (2019). Those were conducted by the National Institute of Standards and Technology (NIST), an agency of the United States Department of Commerce. The ROC curve is a standard for evaluating scores functions for bipartite ranking, a problem that seeks to assign higher scores to elements associated with a label $+1$ than to elements with label -1. We refer to Menon and Williamson (2016) for a survey of bipartite ranking. In that context, similarity functions can be seen as score functions on the product space. That observation suggests that exploring the vast corpus of research on bipartite ranking is a justified approach to find better loss functions for facial recognition, which is what we undertake in both Part II and in Chapter 8 of Part III.

Facial recognition has observed rapid improvements in accuracy and does not require the individual to be cooperative. Therefore, the technology has recently gathered attention in the media and in public opinion. On top of the usual privacy concerns, commentators have expressed growing concerns over the possible unreliability or unfairness of facial recognition.

1.3.3 Reliability of Biometrics

The recent advances in facial recognition have confirmed the maturation of the technology, which foreshadows its deployment and has provoked widespread debate about it. Gates (2011) warned about the tendency of the public to extrapolate about the ubiquity of such systems, which creates an illusion of surveillance that changes behavior. However, the technical hurdles enunciated by Gates (2011) seem much weaker today. Precisely, papers such as Schroff et al. (2015) show the ability of facial recognition systems to handle very loosely controlled acquisition conditions. Also, observers have forecasted a compound annual growth rate (CAGR) — *i.e.* an average geometric progression — of 14.5 % per year between 2020 and 2027 for the global facial recognition market. In that context, the issues regarding the deployment of facial recognition and other machine learning technologies belong to the domain of the legislator, but the decisions of the model are of the responsibility of the machine learning practicioner.

Bias in facial recognition. In the specific case of facial recognition, the NIST has precisely

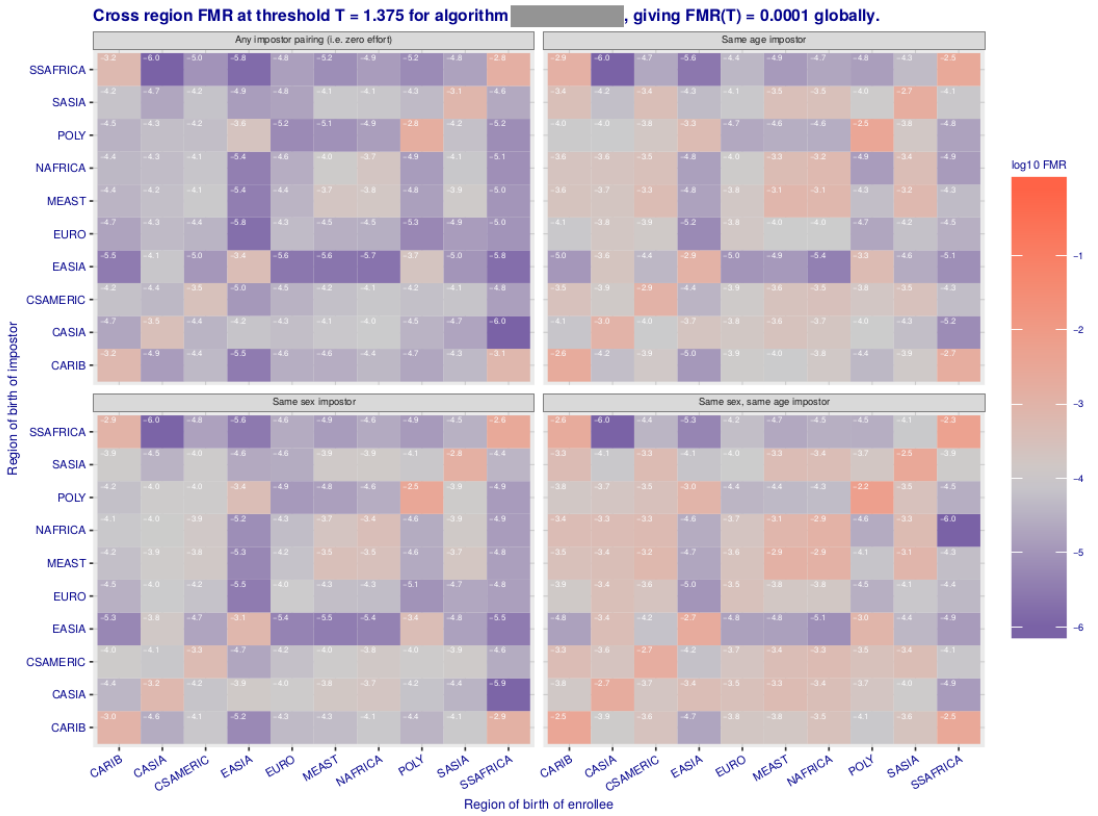


Figure 1.1: This graph shows that fixing a threshold for a fixed false positive rate (called False Matching Rate (FMR) in facial recognition) of $\alpha = 10^{-5}$ on the general population can give higher false positive rate when used on another type of population. Specifically, it gives $\alpha = 10^{-2.8}$ for the population that originates from Polynesia.

quantified important negative differentials in recognition performance for people of color, with Caucasians as the reference class. Figure 1.1 illustrates that and is extracted from Grother and Ngan (2019). The observation was echoed by many news outlets and referred to as "racial bias" in 2019. The main justification was that the face databases used for the training of such systems are generally composed of European and northern American personalities, which are mostly Caucasians and do not represent the general population. Authors interpreted the observation as an "other-race effect" for automatic face recognition, an idea introduced in Furl et al. (2002) and Phillips et al. (2011), that states that humans generally struggle to recognize ethnicities different from their own. Some authors have presented strategies to correct this specific issue explicitly (Wang et al., 2019). More generally, a broad literature on bias in machine learning can be invoked to tackle that issue (Bolukbasi et al., 2016; Zhao et al., 2017; Hendricks et al., 2018; Liu et al., 2016; Huang et al., 2006). Chapter 9 of Part III contributes to that effort, by providing a general reweighting scheme that addresses usual representativeness problems in biometrics.

Limitations of dataset representativeness. While having a database that represents the target population is important, it cannot be expected to correct for inherent biases in the training data. Precisely, even if a social group identified by protected attributes such as race or religion is significantly poorer on average than another social group, it can be deemed as immoral to refuse a loan on the basis that an applicant belongs to the former group. In that context, observers have bluntly qualified predictive algorithms of "opinions embedded in math" (O'Neil, 2016).

Algorithmic fairness. A large body of research (Agarwal et al., 2018; Woodworth et al., 2017; Zafar et al., 2017a,b, 2019; Menon and Williamson, 2018; Bechavod and Ligett, 2017) has emerged on the topic of fairness in machine learning — also called algorithmic fairness — which is a new field of study. It seeks to add explicit constraints to the training phase of machine learning models, so that blunt optimizations for accuracy do not lead to a reproduction of systemic societal biases. Early influential works date all the way back to 2012 (Dwork et al., 2012). More recently,

authors have worked on a textbook dedicated to the subject (Barocas et al., 2019).

Fairness in facial recognition. In facial recognition, including fairness constraints when learning models can correct for social groups being harder to identify than others. It is a necessity in many practical cases. For example, if a system designed to flag persons of interest has a higher rate of false acceptance for a specific ethnicity, it may interpreted as automatic racial profiling.

Fairness literature. The literature in fairness for classification is broad, but there is little work in specific settings, such as ranking or similarity learning. Notable exceptions include Beutel et al. (2019) for ranking, which only modifies a score function with a post-processing step to satisfy a criterion of fairness. This suggests an opportunity for new approaches to fairness that are specifically tailored for those important problems, which we address in Chapter 10 of Part III.

As a whole, the thesis participates in the dialogue between the machine learning community and the biometrics community. It proposes a stylized and theory-oriented view of challenges of biometrics, which leverages recent literature to study the specific — *i.e.* pairwise and functional — criteria used in biometrics. We hope that our perspective on biometric problems will have valuable impacts on practice, and underline that the derivation of statistical guarantees for biometric systems constitutes an important tool to ensure their security.

1.3.4 Thesis outline

The thesis is divided in three parts. Part I of the thesis contains technical preliminaries, which provide all of the intermediary results necessary to derive the theoretical contributions of the thesis. It is featured in the thesis for clarity reasons. Part II and Part III focus on our contributions. Part II delves into the idea of viewing similarity learning as a scoring problem on a product space. Part III engages with the broad idea of reliable machine learning.

Part I is divided in three chapters. The first chapter (Chapter 2) is a quick introduction to statistical learning theory. It presents the necessary results to derive generalization guarantees in the easy setting of binary classification. Precisely, it details the derivation of finite-sample bounds on the excess error of the empirical minimizer. Most of our theoretical contributions can be interpreted as extensions of those results, but our contributions concern more complicated problems. The second chapter (Chapter 3) deals with all of the necessary results that relate to the idea of ranking in machine learning. Our contributions build on two topics related to rankings: bipartite ranking and ranking aggregation. Indeed, our work extends existing bipartite ranking guarantees to the similarity ranking setting presented in Part II, ranking aggregation is used for bagging ranking trees, and the guarantees of the first chapter of Part III are built on a parametric model for rankings, which can be considered as a tool for ranking aggregation. Finally, the third and last chapter (Chapter 4) features a short introduction to important results in U-statistics, an essential component of all pairwise learning problems. As such, Chapter 4 is a prerequisite to our similarity ranking guarantees, and is involved in the estimation of risk functionals in standard bipartite ranking.

Part II explores the idea of considering the biometrics verification problem as ranking pairs of instances. It is divided in three chapters. The first chapter (Chapter 5) presents formally the idea of seeing similarity learning through the lens of ROC optimization. It proposes novel guarantees for the problem of pointwise ROC optimization, which seeks to optimize for true positive rates under an upper bound on the false positive rate. This analysis paves the way for an extension of the guarantees of the TreeRank algorithm to similarity learning, that we deliver. Using numerical simulations, we provide the first empirical illustration of fast learning rates, tailored here to the specific case of pointwise ROC optimization for similarity ranking. Due to the prohibitive number of pairs involved in the computations, the propositions of the first chapter are not practical for any large-scale application. To correct this, statisticians proposed sampling approximations for U-statistics, which is an useful approach in similarity ranking. The second chapter (Chapter 6) extends that proposition to settings where the dataset is distributed. Finally, the third chapter (Chapter 7) is rather prospective and proposes simple numerical toy experiments on similarity ranking, that address the optimization aspect of the problem. The extension of those experiments will be the subject of future work.

Part III is the last part of the thesis. It revolves around the idea of reliable machine learning and

is also divided in three chapters. The first chapter (Chapter 8) derives learning guarantees for predicting an ordering over possible classes with only multiclass classification data, which relies on the use of a One-Versus-One (OVO) strategy. That problem often arises in noisy problems, *i.e.* those for which the top-1 ranked class has a good chance to be a false positive. One is then interested in the most likely classes, as is often the case in forensics. The second chapter (Chapter 9) proposes techniques to correct bias between the train and test sample, with auxiliary information on the difference between the two. It relies on an application of the well-known principle of importance sampling in statistics. Finally, the third chapter (Chapter 10) proposes an unification of a class of fairness constraints, as well as a new and more restrictive fairness constraint, that is more suited to practical situations. It also features theoretical guarantees and gradient-based approaches for learning under both types of constraints.

The last chapter of the thesis (Chapter 11) contains a summary of the contributions of Part II and Part III, as well as a detailed account of the most promising directions for future work. It ends with a general conclusion on the thesis.

The next two sections of this chapter — Section 1.4 and Section 1.5 — summarize the contributions of the thesis. Each section focuses respectively on the first and second part of the thesis, and is divided in subsections which summarize each chapter of the part. Finally, Section 1.6 sums up the perspectives of the thesis.

1.4 Similarity Ranking

Similarity learning plays a key role in many machine learning problems such as clustering, classification or dimensionality reduction. It is especially important when one considers open-world problems, — *e.g.* situations when a model encounters classes after deployment that were not available during training (Chen et al., 2018) — which is the case for any biometric application. In this section, we consider metric learning from the perspective of scoring pairs of instances, which is coherent with the evaluation of many systems based on metric learning techniques.

1.4.1 Similarity Ranking Theory

Bipartite ranking/scoring considers a set of elements associated to a binary label, and seeks to rank those with label $+1$ higher than those with label -1. To derive an order on an input space $\mathcal{X}$, bipartite ranking is generally tackled by learning a score function $s : \mathcal{X} \to \mathbb{R}$ (Menon and Williamson, 2016). On the other hand, the field of metric/similarity learning (Bellet et al., 2015a) is the task of learning a similarity — or equivalently, a distance — $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ on the product space $\mathcal{X} \times \mathcal{X}$. While metric learning algorithms were originally evaluated *w.r.t.* their relevance for a clustering task (Xing et al., 2002), today practitioners use performance indicators derived from the ROC curve, the golden standard for evaluating scoring functions in bipartite ranking. Therefore, our work introduces as *similarity ranking* the idea of learning similarities for a ranking objective.

A functional criterion: the ROC **curve.** In the multi-class classification setting, introduce a random pair $(X, Y) \in \mathcal{X} \times \{1, \dots, K\}$, with $K \in \mathbb{N}$ the number of classes, as well as an independent copy (X', Y') of (X, Y). Then, one can define a random variable $Z = 2 \cdot \mathbb{I}\{Y = Y'\} - 1$ that is equal to 1 if both pairs belong to the same class and -1 otherwise. The ROC curve of a similarity function is then equal to the PP-plot $t \in \mathbb{R} \mapsto (\bar{H}_s(t), \bar{G}_s(t))$, where, for all $t \in \mathbb{R}$:

$$\bar{H}_s(t) := \mathbb{P}\{s(X, X') > t \mid Z = -1\} \qquad \text{and} \qquad \bar{G}_s(t) := \mathbb{P}\{s(X, X') > t \mid Z = +1\}.$$

$\bar{H}_s(t)$ and $\bar{G}_s(t)$ are respectively the false positive and true positive rate associated to the similarity s. Under continuity assumptions, the ROC curve writes as the graph of the function $\alpha \in (0, 1) \mapsto \mathrm{ROC}_s(\alpha) = \bar{G}_s(t) \circ \bar{H}_s^{-1}(\alpha)$. Previous approaches for similarity learning optimize an empirical evaluation of the Area Under the ROC Curve (AUC) of the similarity function s (McFee and Lanckriet, 2010; Huo et al., 2018).

Pointwise ROC **optimization (p**ROC**).** The AUC is a global summary of the ROC curve which penalizes ranking errors regardless of their position in the list (Clémençon et al., 2008,

Proposition B.2). Other criteria focus on the top of the list (Clémençon and Vayatis, 2007; Huo et al., 2018), and their study is the subject of the *Ranking the Best* literature (Menon and Williamson, 2016, Section 9). In our work, we consider optimizing the true positive rate attained by a similarity under an upper bound $\alpha \in (0, 1)$ on its false positive rate. This setting is relevant in biometric applications, as security guarantees are specified with a limit on the false positive rate of the system. We refer to this problem as *pointwise* ROC *optimization* (pROC). Considering the risks:

$$R^-(s) := \mathbb{E}[s(X, X') \mid Z = -1] \qquad \text{and} \qquad R^+(s) := \mathbb{E}[s(X, X') \mid Z = +1],$$

with $\mathcal{S}$ a proposed family of similarities, the pROC problem writes:

$$\max_{s \in \mathcal{S}} R^+(s) \qquad \textit{subject to} \qquad R^-(s) \leqslant \alpha. \tag{1.1}$$

We denote a solution of Eq. (1.1) as s^*. Clémençon and Vayatis (2010) studied the equivalent of Eq. (1.1) in bipartite ranking. This problem is analogous to Neyman-Pearson classification (Scott and Nowak, 2005), and bears close resemblance to the minimum-volume set problem (Scott and Nowak, 2006). When $\mathcal{S}$ is the class of all measurable functions, a solution of Eq. (1.1) writes as a super-level set of the posterior probability $\eta : x, x' \mapsto \mathbb{P}\{(X, X') = (x, x') \mid Y = Y'\}$, which is a consequence of the Neyman-Pearson fundamental lemma (Lehmann and Romano, 2005, Theorem 3.2.1).

Pairwise estimators. The analysis of Clémençon and Vayatis (2010) relies on the fact that natural estimators of $R^-(s)$ and $R^+(s)$ are standard empirical means in the case of bipartite ranking, However, it is not true in similarity ranking. Consider a sample $\mathcal{D}_n = \{(X_i, Y_i)\}_{i=1}^n$ composed of n *i.i.d.* copies of the pair (X, Y), then the natural estimators of $R^-(s)$ and $R^+(s)$ based on $\mathcal{D}_n$ write:

$$R_n^-(s) := \frac{1}{n_-} \sum_{i<j} \mathbb{I}\{Y_i \neq Y_j\} \cdot s(X_i, X_j), \tag{1.2}$$

$$R_n^+(s) := \frac{1}{n_+} \sum_{i<j} \mathbb{I}\{Y_i = Y_j\} \cdot s(X_i, X_j), \tag{1.3}$$

where $n_+ := \sum_{i<j} \mathbb{I}\{Y_i = Y_j\}$ and $n_- := n(n-1)/2 - n_+$. The quantities in Eq. (1.2) and Eq. (1.3) are not sums of independent random variables, hence the analysis of Clémençon and Vayatis (2010) breaks down. However, they are ratios of the well-known U-statistics (Lee, 1990; de la Pena and Giné, 1999).

Generalization guarantees for pROC**.** The empirical counterpart of pROC (Eq. (1.1)) writes:

$$\max_{s \in \mathcal{S}} R_n^+(s) \qquad \textit{subject to} \qquad R_n^-(s) \leqslant \alpha + \phi, \tag{1.4}$$

where $\phi \geqslant 0$ is a term, which tolerates the variations of $R_n^-(s)$ around its expectation $R^-(s)$. We denote by s_n a solution of Eq. (1.4). The generalization of standard concentration inequalities to U-statistics enables us to extend the uniform guarantees of Clémençon and Vayatis (2010). Precisely, we simultaneously ensure with high probability: that $R^+(s^*) - R^+(s_n)$ is upper-bounded by a quantity of order $n^{-1/2}$ and that $R^-(s_n) \leqslant \alpha + \phi_n$ with $\phi_n = O(n^{-1/2})$. In summary, we show that both excesss risk are bounded with the standard learning speed in $n^{-1/2}$ without assumptions on the distribution of the data.

Fast generalization speeds for pROC**.** In the case of binary classification, Mammen and Tsybakov (1995) have shown that, under a noise assumption parameterized by $a \in (0, 1)$ on the data distribution, fast convergence speeds in $O(n^{-1/(2-a)})$ hold. The fast speed is a consequence of an upper-bound on the variance of the excess risk, that is derived from the noise assumption. The analysis of Clémençon and Vayatis (2010) built on those ideas to propose an upper bound on $R^+(s^*) - R^+(s_n)$ in $O(n^{-(2+a)/4})$ with similar guarantees in $O(n^{-1/2})$ for R^-, for pROC in bipartite ranking. Compared to the binary classification setting, pROC has lower learning speeds, which comes from the bilateral nature of pROC. Our work extend the fast speeds of Clémençon and Vayatis (2010) from bipartite ranking to the case of similarity ranking. Incidentally, the result is true under a much weaker assumption. Precisely, it relies on the second Hoeffding's decomposition for U-statistics (Hoeffding, 1948), which implies that the deviation of the excess

risk consists mainly in that of its Hájek projection. Since the Hájek projection is a variance-reducing transformation of a U-statistic (van der Vaart, 2000, Section 11), weaker assumptions imply the upper-bound on the variance required for fast learning speeds. Clémençon et al. (2008) featured that usage of the properties of U-statistics to derive fast convergence speeds, but never applied it to problems that feature a random constraint.

Empirical illustration of fast speeds. Our work also contains the first experimental illustration of fast speeds of convergence, on our specific similarity ranking problem. The illustration relies on the generation of data that satisfies the noise assumption for different noise parameters $a \in (0, 1)$, followed by a comparison of their empirical learning rates. We chose the data distribution and the proposed family $\mathcal{S}$ so that the optimal similarity s^* is known and the empirical minimizer s_n can be found exactly. For that matter, we set $\mathcal{S}$ as the decision stumps on a fixed transformation of the data.

Limitations of pROC**.** While the pROC problem echoes practical considerations in biometrics, where systems are deployed to work at a fixed rate of false positives α, its empirical resolution is hard in practice. Few exceptions rely on fixed partitioning of the input space (Scott and Nowak, 2006). In many situations, the false positive rate α for a system is unknown in advance, thus optimizing for the wrong α may not yield satisfying outcomes at deployment.

TreeRank for bipartite ranking. The TreeRank algorithm for bipartite ranking was introduced in Clémençon and Vayatis (2009). TreeRank learns a piecewise constant score function s_{D_n}, built to provide an adaptive piecewise linear estimation of the optimal ROC curve. As implied by optimal solutions of Eq. (1.1), the optimal ROC curve ROC^* is that of the posterior probability η. TreeRank recursively splits the input space $\mathcal{X}$ and optimizes greedily the AUC at each split, thus forming a binary tree of depth D_n of nested partitions of the input space $\mathcal{X}$. Under specific assumptions, Clémençon and Vayatis (2009) have proven uniform bounds in supremum norm between the optimal ROC curve and that of s_{D_n} when $D_n \sim \sqrt{\log(n)}$, *i.e.* that with large probability:

$$\sup_{\alpha \in [0,1]} \left| \mathrm{ROC}_{s_{D_n}}(\alpha) - \mathrm{ROC}^*(\alpha) \right| \leqslant \exp(-\lambda \sqrt{\log(n)}), \tag{1.5}$$

where λ is a constant specified by the user.

TreeRank for similarity ranking. Our work proposes an extension of the TreeRank algorithm for learning similarities, by considering recursive splits of the product space $\mathcal{X} \times \mathcal{X}$. To ensure that the similarity s is symmetric, we consider only symmetric splits with respect to the two arguments in the input space $\mathcal{X} \subset \mathbb{R}^d$, by splitting on the following simple reparametrization of $\mathcal{X} \times \mathcal{X}$:

$$f : (x, x') \mapsto \begin{pmatrix} |x - x'| \\ x + x' \end{pmatrix}.$$

Using the same extensions of classic concentrations inequalities to U-statistics as before, we extended the proof of Eq. (1.5) to similarity ranking. Our analysis provides a theoretically-supported approach to learning similarities that approach the optimal ROC curve in supremum norm.

While we have proven theoretical guarantees for approaches to similarity ranking, the estimators involved in the computation of the risk functionals require performing sums of very large numbers of terms. The induced computational cost makes the practical application of such approaches prohibitive. For example, calculating of $R_n^-(s)$ require summing n_- terms, which is quadratic in n when K is constant. In typical biometric applications, the number of samples per class is fixed. Hence, the proportion of negative pairs n_- over all pairs n^2 is even higher than the case K constant. The next section exploits recent analyses in the approximation of U-statistics to alleviate that problem.

1.4.2 Distributed U-Statistics

Most biometric applications learn on large-scale datasets. For facial recognition, the largest dataset released to the public contains 8.2 million images (Guo et al., 2016) and private datasets are much larger. The scale of facial recognition datasets justifies the computational concerns

described in Section 1.4.1, as the number of negative pairs is higher than 50 trillions (10^{12}) for Guo et al. (2016). Besides this restriction on the number of operations, these datasets often cannot be contained in the random-access memory (RAM) of a single machine. In our work, we proposed an approach to tackle the estimation of U-statistics in a distributed environment, which deals with those two limitations simultaneously.

Incomplete U-statistics. The idea of alleviating the computational complexity of U-statistics is not new, as Blom (1976) proposed summing over a small finite number of B pairs selected by sampling with replacement to in the set of all pairs to form *incomplete U-statistics* in 1976. Clémençon et al. (2016) derived an upper bound on the deviation between an incomplete U-statistic U_B and the complete U-statistic U_n that holds with high probability. In the case of one sample U-statistics of degree two — averages of all possible pairs formed with a sample —, the bound implies that using the estimator U_B with only $B = n$ pairs suffices to recover a usual learning rate of order $n^{-1/2}$, instead of summing all $n(n-1)/2$ pairs as for U_n. This result implies that one can extend the proofs for similarity ranking presented above so that they work with incomplete U-statistics, which makes the setting practical in large-scale learning applications.

Distributed environments. In cases where the data does not fit on a single machine, the recent technological progresses regarding distributed databases and parallel computation made the deployment of distributed machine learning accessible, largely due to the development of frameworks for cluster computing, such as Apache Spark (Zaharia et al., 2010) or Petuum (Xing et al., 2015). These frameworks abstracted away the network and communication aspects of distributed algorithms. As such, they eased the deployment of distributed algorithms, but have restricted the types of operations that can be efficiently achieved, generally in order to guarantee algorithmic properties. At the same time, Jordan (2013) urged statisticians to guide practitioners of large-scale machine learning, by studying the implications of distribution on estimation, particularly to put in perspective the gains in computational time with the potential losses in statistical accuracy. Our work addresses this issue, by proposing several estimators for U-statistics in a distributed setting and comparing their variances. In that context, we propose time versus variance tradeoffs.

Probabilistic framework. Introduce two independent *i.i.d.* samples $\mathcal{D}_n = \{X_1, \ldots, X_n\} \subset \mathcal{X}$ and $\mathcal{Q}_m = \{Z_1, \ldots, Z_m\} \subset \mathcal{Z}$ of respectively $n \in \mathbb{N}$ and $m \in \mathbb{N}$ elements, such that $\mathcal{D}_n$ and $\mathcal{Q}_m$ can have different distributions. The complete two-sample U-statistic of kernel $h : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ associated with those samples writes as:

$$U_{\mathbf{n}}(h) := \frac{1}{nm} \sum_{k=1}^{n} \sum_{l=1}^{m} h(X_k, Z_l), \tag{1.6}$$

with $\mathbf{n} = (n, m)$. On the other hand, the incomplete counterpart of $U_n(h)$ based on B pairs writes as:

$$U_B(h) := \frac{1}{B} \sum_{(k,l) \in \mathcal{D}_B} h(X_k, Z_l), \tag{1.7}$$

where $\mathcal{D}_B$ is a set of B elements selected at random in the set of all pairs $\{(k, l) \mid (k, l) \in \{1, \ldots, n\} \times \{1, \ldots, m\}\}$. For large-scale data, the full datasets $\mathcal{D}_n$ and $\mathcal{Q}_m$ can not fit on a single machine, which makes the direct computation of Eq. (1.6) and Eq. (1.7) impossible. In that context, the standard approach is to distribute the data on $N \in \mathbb{N}$ workers. For a standard mean, the simple computation of the average of the local means for each worker yields the same estimator as in the centralized setting. The computation is not that simple for U-statistics, since each worker can only form pairs with the local sample without network communication.

Distributed estimators for U-statistics. We first introduced two simple estimators in the distributed setting that do not require network communication: the average $U_{\mathbf{n},N}$ of N complete U-statistics on each local sample, and the average $U_{\mathbf{n},N,B}$ of N incomplete U-statistics formed with B randomly selected pairs on each local sample. Using the second Hoeffding decomposition, we derive an analytical formulation for the variances of those estimators, following the steps of Hoeffding (1948) for $U_{\mathbf{n}}$. Their expression shows that the estimators $U_{\mathbf{n},N}$ and $U_{\mathbf{n},N,B}$ have a limited accuracy, *i.e.* a minimum variance, that can be significantly higher than that of $U_{\mathbf{n}}$ for a specific h and specific distributions of X_1 and Z_1.

Repartition for distributed estimators. This difference in variance between distributed estimators and U_n comes from the fact that many pairs involved in the computation of the latter are not involved in that of the former. To counterbalance this effect, we propose to average estimators computed between repartitioning procedures, that reassign the observations to each of the clusters at random, so that every pair involved in $U_{\mathbf{n}}$ has a chance to be seen. We propose two estimators with repartitioning: $U_{\mathbf{n},N,T}$ (resp. $U_{\mathbf{n},N,B,T}$) that averages T estimators $U_{\mathbf{n},N}$ (resp. $U_{\mathbf{n},N,B}$) computed on T different partitioning of the data. As T grows, the variance of those estimators approach the variance of $U_{\mathbf{n}}$ by above.

Relative variance of our estimates. We provide analytical expressions for the variance of the estimators $U_{\mathbf{n}}, U_B, U_{\mathbf{n},N}, U_{\mathbf{n},N,B}, U_{\mathbf{n},N,T}$ and $U_{\mathbf{n},N,B,T}$, for several ways of distributing the data on the workers. In that regard, we first consider sampling without replacement (SWOR), which is relevant when splitting all data on several machines for space constraints. Then, we consider with replacement (SWR), which is relevant when selecting batches of data to compute several estimates in parallel, *e.g.* in minibatch gradient descent. We assume for both settings that each worker contains n/N elements of the sample $\mathcal{D}_n$ and m/N elements of $\mathcal{Q}_m$, a setting we call *prop*. Relaxing that assumption implies that there is a nonzero probability to have no elements of either $\mathcal{D}_n$ or $\mathcal{Q}_m$ in a worker. In that case, one has to provide a default value for the estimator of that worker. Hence, the variance has no simple interpretable analytic form, so we provide empirical evidence that the observed variances are of the same order. Additionally, we characterize the parameters h, n, m and the distributions of X_1, Z_1 for which repartitioning is important.

Learning with distributed U-statistics. Papa et al. (2015) studied stochastic gradient descent for incomplete U-statistics. While we do not extend their analysis to our distributed estimators, our work considers the minimization of U-statistics with stochastic gradient descent in a distributed environment, by estimating the gradient with $U_{\mathbf{n},N,B}$ and repartitioning the data every n_r timesteps. We provide empirical evidence that lowering n_r gives a better solution to the optimization process on average. Also, the variance of the loss of the final solution is much smaller, which shows the increased robustness obtained by repartitioning the data.

While Section 1.4.1 and this section account respectively for the generalization and scalability aspect of similarity ranking, the next section focuses on the optimization aspect. As such, the next section is more prospective and gives optimization strategies for similarity ranking problems on toy examples.

1.4.3 Practical Similarity Ranking

The development of the deep metric learning literature is motivated by a need for criteria that align with the requirements of biometric identification, and can be optimized by gradient descent. Most of those criteria focus on better separating the identities with some heuristic, such as that all instances associated to a same identity should be mapped to the same point in a representation space, see the center loss (Wen et al., 2016). While those criteria are sensible, they are poorly connected with the evaluation of biometric systems, which is based on the ROC curve. The ROC curve is a functional made to assess the capacity of a scoring function to distinguish positive from negative instances, or in biometrics the capacity of a similarity function to distinguish matching instances from non-matching instances. The connection between bipartite ranking and biometrics motivates us to propose practical approaches for learning similarity functions, that optimize a measure derived from the ROC curve.

Previous analyses of similarity ranking. Viewing similarity learning as pairwise bipartite ranking is extensively discussed in other parts of the thesis under the name similarity ranking. Indeed, we provided generalization results for the pointwise ROC optimization (pROC) problem, sometimes referred to as Neyman-Pearson classification (Scott and Nowak, 2006). The pROC problem consists in finding a score s and a threshold t for it, that give the highest possible true positive rate under an upper-bound on the false positive rate. Additionally, we provided an extension of the analysis and methodology of the TreeRank algorithm, which was originally shown to learn score functions s that approximate the optimal ROC in supremum norm (Clémençon and Vayatis, 2009), to learn similarity functions. Finally, we motivated theoretically such methods, as we prove that one can replace the computationally intensive estimators involved by incomplete

U-statistics, which dealt with the computational limitations associated to similarity ranking.

Limitations of previous analyses. Our theoretical results justify considering the optimization of the pROC problem or the use of the TreeRank algorithm. However, they do not imply obvious approaches to optimize for the pROC problem, nor guarantee the empirical performance of our extension of the TreeRank algorithm. This chapter of the thesis provides illustrations of possible approaches, that are demonstrated to work on simple synthetic datasets. Deriving an extensive empirical evaluation of those approaches is an interesting direction for future work.

Solving pROC **for linear similarities.** While many papers discuss the pROC problem (Scott and Nowak, 2006; Clémençon and Vayatis, 2010; Rigollet and Tong, 2011), there lacks practical approaches to it, with the exception of a few propositions based on recursive partitioning, see *e.g.* Scott and Nowak (2006). Introduce a sample of n data points as $\mathcal{D}_n := \{(x_i, y_i)\}_{i=1}^n \subset \mathcal{X} \times \{1, \dots, K\}$, where y_i is the identity of an observation $x_i \in \mathcal{X}$ and $\mathcal{X} \subset \mathbb{R}^d$. In the case of similarity ranking, pROC at level $\alpha \in (0, 1)$ over a class of proposed functions $\mathcal{S}$, with $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ for any $s \in \mathcal{S}$, writes:

$$\max_{s \in \mathcal{S}} \frac{1}{n_+} \sum_{i<j} \mathbb{I}\{y_i = y_j\} \cdot s(x_i, x_j) \quad \text{subject to} \quad \frac{1}{n_-} \sum_{i<j} \mathbb{I}\{y_i \neq y_j\} \cdot s(x_i, x_j) \leqslant \alpha. \tag{1.8}$$

In the general case, Eq. (1.8) can be very hard to solve. For example, if $\mathcal{S}$ is composed of indicators of sets in $\mathcal{X} \times \mathcal{X}$, then Eq. (1.8) might not be differentiable, nor continuous. However, if the family $\mathcal{S}$ has a simple form, solving Eq. (1.8) may be much easier. For example, we propose an explicit analytical solution when $\mathcal{S}$ is the set of all bounded bilinear similarities $(x, x') \mapsto x^\top A x$ with $A \in \mathbb{R}^{d \times d}$ and $\|A\|_F \leqslant 1$ where $\|\cdot\|_F$ is the Frobenius norm. While we show that this approach gives sensible solutions to pointwise ROC optimization for very specific distributions, the general case requires more flexible families of functions.

Solving pROC **with gradient descent.** To address pROC with more complex proposed families, we propose a gradient descent based approach to minimize a counterpart of the excess error for minimum volume set estimation of Scott and Nowak (2006), that is adapted for bipartite ranking. While we demonstrate its efficiency with a simple linear classifier and a toy example, the approach is flexible enough to accomodate for more complex models, as well as an extension to similarity ranking. However, we will have to demonstrate its performance empirically on more eloquent examples. Introduce as H and G the distributions of respectively $X \mid Y = -1$ and $X \mid Y = +1$, as well as the optimal rejection region for pointwise ROC optimization as R^*_α, then our relaxation of the excess error writes:

$$\max_{(w,b) \in \mathbb{R}^d \times \mathbb{R}} \left(G(R^*_\alpha) - \widehat{G}(w, b) \right)_+ + \left(\widehat{H}(w, b) - \alpha \right)_+ \qquad \text{s.t.} \qquad \|w\|_2^2 + b^2 \leqslant 1. \tag{1.9}$$

where $\widehat{G}$ and $\widehat{H}$ are relaxed empirical versions of respectively G and H, and $(x)_+ := \max(x, 0)$ for any $x \in \mathbb{R}$. By minimizing the objective of Eq. (12.9) and projecting the weights on the unit ball, our method recovers optimal rejection regions by gradient descent, for our toy example. Though the quantity $G(R^*_\alpha)$ is unknown, we conjecture that the results are not extremely sensitive to this value, and that reasonable approximations can be proposed for it in most applications. The pROC problem is a principled approach to learning similarities with a ranking objective, but does not address ranking as a global problem. Precisely, it focuses on recovering a single level set of the score function, as demonstrated by the fundamental lemma of Neyman-Pearson, whereas bipartite ranking consists in recovering an order relation between any two points of the input space.

TreeRank for similarity learning. The TreeRank algorithm of Clémençon and Vayatis (2009) deals with bipartite ranking as a global problem. Precisely, it tackles bipartite ranking with a recursive splitting procedure, that solves binary classification problems with asymmetrically weighted errors between positives and negatives instances. As presented before, our work extended the algorithm to the case of similarity ranking, as well as the supremum norm guarantees for the distance of the ROC of the learned score to the optimal ROC. To build intuition on our TreeRank variant for similarity ranking, we illustrate visually the shape of our proposed symmetric regions for splitting the space.

Practical considerations for TreeRank. Two drawbacks of TreeRank are : 1) its dependence on the initial splits of the space, which jeopardize performance if the proposed splitting

family is too limited, 2) and its discontinuous nature, which is inconsistent with natural hypotheses of many practical settings. A first response to these issues is to extend the idea of random forests proposed in Breiman (2001) to TreeRank, which is proposed in Clémençon et al. (2013). We show on a toy dataset with continuous likelihood ratio dG/dH that averaged ranking trees can correct for both the initial mispecification of the proposed family and the discrete nature of ranking trees. Precisely, the averaged score function that we obtain gives a ROC curve that is almost undistinguishable from the optimal ROC, despite both its finite — but numerous — set of values, and the inadequacy of each tree with respect to the true likelihood ratio.

These proposals sketch a way for new algorithms specifically designed for solving the similarity ranking problem, which is a principled approach to dealing with the biometric identification problem. However, they are merely illustrations on very simple problems. Finding new approaches for practical similarity ranking, ideally for large-scale experiments that correspond to operational scenarios in biometrics, is an exciting track for future work.

1.5 Reliable Machine Learning

Besides the similarity learning aspect, biometrics and particularly facial recognition incarnate many important issues in machine learning, as shown in the reports of the NIST (National Institute of Standards and Technology) on industrial facial recognition benchmarks (Grother and Ngan, 2019). Precisely, facial recognition is confronted with issues regarding the robustness of predictions, bias in the training data, as well as algorithmic fairness. The subsections of this section tackle all of those issues sequentially. While there is extensive literature on those general topics, our work deals with challenging and underexamined settings that directly apply to biometric problems. Precisely: 1) for prediction robustness, we consider learning an ordered list of identities, as do particular biometric identification problems, 2) for training data bias, we reweight training instances using high-level information, which can be nationality in the context of border control, 3) for fairness, we focus on score functions as a gateway to similarity functions, as done in Part II.

1.5.1 Ranking the Most Likely Labels

In biometrics systems for law enforcement, an human expert often analyzes several of the most likely suspects proposed by a system. More generally, hard classification problems focus on the most likely labels, such as the ILSVRC (ImageNet Large Scale Visual Recognition Challenge) challenge, where Krizhevsky et al. (2012) and subsequent papers consider the accuracy at top-5 alongside the usual (top-1) prediction accuracy. In our work, we propose an approach to tackle predicting an ordered list of labels from classification data. We refer to that problem by the name of *label ranking* (LR). Precisely, we propose to use the well-known One-versus-One approach to classification, and derive guarantees for that approach.

Probabilistic framework for label ranking (LR). As usual in the multiclass classification setting, consider a random pair $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ with $\mathcal{Y} = \{1, \dots, K\}$, as well as the risk $L(g) = \mathbb{P}\{g(X) \neq Y\}$ associated to a classifier $g : \mathcal{X} \to \mathcal{Y}$. The optimal classifier for $L(g)$ in the class of all measurable functions is the well-known Bayes classifier g^*, defined as follows:

$$g^*(x) := \operatorname*{arg\,max}_{k \in \{1,\dots,K\}} \eta_k(x),$$

where $\eta_k : x \mapsto \mathbb{P}\{Y = k \mid X = x\}$ is the posterior probability of class k for any $k \in \{1, \dots, K\}$. We refer to the task of finding an ordered list of the most likely labels as *label ranking* (LR). It amounts to associating to any $x \in \mathcal{X}$ a permutation $\sigma_x^* \in \mathfrak{S}_K$, such that:

$$\eta_{\sigma_x^{*-1}(1)} > \eta_{\sigma_x^{*-1}(2)} > \cdots > \eta_{\sigma_x^{*-1}(K)}. \tag{1.10}$$

We denote by σ_X^* the corresponding random permutation, *i.e.* σ_X^* satisfies, for any $\sigma \in \mathfrak{S}_K$, $\mathbb{P}\{\sigma_X^* = \sigma\} = \mathbb{P}\{X \in A_\sigma\}$ with $A_\sigma = \{x \in \mathcal{X} \mid \sigma_x^* = \sigma\}$.

On ranking median regression (RMR). Another well-known problem in statistical literature that concerns predicting a ranking over a set of labels is *ranking median regression* (RMR)

(Tsoumakas et al., 2009; Vembu and Gärtner, 2010; Clémençon et al., 2018). RMR considers a random pair $(X, \Sigma) \in \mathcal{X} \times \mathfrak{S}_K$, hence Σ is a random permutation. RMR learns from data a ranking rule $s : \mathcal{X} \to \mathfrak{S}_K$ that minimizes the risk:

$$R(s) := \mathbb{E}[d(\Sigma, s(X))], \tag{1.11}$$

where $d : \mathfrak{S}_K \times \mathfrak{S}_K \to \mathbb{R}_+$ is a symmetric loss function. The distance d quantifies the distance between rankings. The most well-known distance is the *Kendall τ distance* d_τ, which is equal to the number of pairwise disagreements between the two permutations.

Optimal solution of RMR. Previous work on RMR (Clémençon et al., 2018) has shown that: the optimal minimizer of Eq. (1.11) for measurable ranking rules has a simple analytical formulation for the distance d_τ, under an assumption called the Strict Stochastic Transitivity (SST). SST assumes that the pairwise probabilities $p_{k,l}(x) := \mathbb{P}\{\Sigma(k) < \Sigma(l) \mid X = x\}$ and $p_{l,k}(x) := 1 - p_{k,l}(x)$ for any $1 \leqslant k < l \leqslant K$ satisfy: for all $x \in \mathcal{X}$ and $(i, j, k) \in \{1, \ldots, K\}^2$ with $i \neq j$, we have $p_{i,j}(x) \neq 1/2$ and:

$$p_{i,j}(x) > 1/2 \qquad \textit{and} \qquad p_{j,k}(x) > 1/2 \qquad \Rightarrow \qquad p_{i,k}(x) > 1/2.$$

Under the SST assumption, the optimal ranking rule for d_τ writes:

$$s^*_X(k) = 1 + \sum_{l \neq k} \mathbb{I}\{p_{k,l}(X) < 1/2\}. \tag{1.12}$$

LR as RMR with partial information. While the characterization of the optimal element is a welcome extension of usual learning theory (Devroye et al., 1996) to the RMR problem, the whole ranking Σ is not available when dealing with classification data. However, our work shows that if we consider the random permutation Σ to be generated by a conditional BLTP model (Korba, 2018) with preference vector $\eta(X) = (\eta_1(X), \ldots, \eta_K(X))$, then it is possible to build Σ so that it satisfies $Y = \Sigma^{-1}(1)$ almost surely. Based on this observation, we propose to consider LR as a RMR problem with the partial information $\Sigma^{-1}(1)$ about the full random permutation Σ.

Optimal solutions of LR with One-versus-One (OVO). One can calculate the expressions of the pairwise probabilities $p_{k,l}(x)$'s under the conditional BTLP model with preference vector $\eta(x)$. Precisely, we have $p_{k,l}(x) = \eta_k(x)/(\eta_k(x) + \eta_l(x))$ for any $x \in \mathcal{X}$ and $k < l$. Remark that $p_{k,l}(x)$ corresponds to the probability of predicting k against l for the One-versus-One (OVO) problem of classifying k against l. The One-versus-One (OVO) approach is a well-studied approach (Hastie and Tibshirani, 1997; Moreira and Mayoraz, 1998; Allwein et al., 2000; Fürnkranz, 2002) to tackle multiclass classification using binary classification algorithms. OVO consists in learning $K(K-1)/2$ decision functions, *i.e.* a classifier for each class k against l with $k < l$, and taking the majority vote of the $K(K-1)/2$ classifiers. The Bayes classifier for the (k, l) OVO problem is $g^*_{k,l} : x \mapsto 2 \cdot \mathbb{I}\{p_{k,l}(x) \geqslant 1/2\} - 1$. Hence, Eq. (1.12) reduces to:

$$s^*_X(k) = 1 + \sum_{l \neq k} \mathbb{I}\{g^*_{k,l}(X) = -1\}. \tag{1.13}$$

Notice that s^*_X corresponds to σ^*_X in Eq. (1.10), as soon as all of the $\eta_k(X)$ are distinct. We showed that a combination of the optimal solutions of all of the $K(K-1)/2$ OVO problems imply an optimal ranking rule s^*_X. Hence, we can probably derive a good solution of the LR problem from good solutions of all of the OVO problems.

Guarantees for LR with OVO. We propose a solution to LR, that uses a combination of the solutions of all empirical OVO classification problems. Then, we derive theoretical guarantees for that solution. Introduce a sample $\mathcal{D}_n = \{(X_i, Y_i)\}_{i=1}^n$ of n *i.i.d.* copies of the random pair (X, Y), as well as the notation $Y_{k,l,i} = \mathbb{I}\{Y_i = l\} - \mathbb{I}\{Y_i = k\}$ for any $k < l$ and any $i \in \{1, \ldots, n\}$. The empirical risk $\widehat{L}_{k,l}$ of $g : \mathcal{X} \to \{-1, +1\}$ for OVO classification of k versus l writes:

$$\widehat{L}_{k,l}(g) := \frac{1}{n_k + n_l} \sum_{i : Y_i \in \{k,l\}} \mathbb{I}\{g(X_i) \neq Y_{k,l,i}\},$$

where $n_k = \sum_{i=1}^n \mathbb{I}\{Y_i = k\}$ for any $k \in \{1, \dots, K\}$. We denote the minimizer of $\widehat{L}_{k,l}$ over the fixed proposed class $\mathcal{G}$ of binary classifiers as $\widehat{g}_{k,l}$. Following Eq. (1.13), an empirical solution of LR writes:

$$\widehat{s}_X(k) := 1 + \sum_{k \neq l} \mathbb{I}\{\widehat{g}_{k,l}(X) = -1\}.$$

A simple union bound implies:

$$\mathbb{P}\{\widehat{s}_X \neq s^*_X\} \leqslant \sum_{k<l} \mathbb{P}\{\widehat{g}_{k,l}(X) \neq g^*_{k,l}(X)\}. \tag{1.14}$$

Eq. (1.14) shows that the sum of the probabilities of not predicting the optimal class for each OVO problem bounds the probability of not predicting the optimal list of labels in LR. At the same time, a consequence of usual hypotheses for the derivation of fast generalization speeds — first presented in Mammen and Tsybakov (1995) and reviewed in Boucheron et al. (2005) — is an upper-bound of the *r.h.s.* quantity in Eq. (1.14) by the excess errors of the k versus l classification problems. Under a standard noise assumption, we provide usual fast convergence bounds in $O(n^{-1/(2-a)})$ for the excess error of each k versus l classification problem, where $a \in (0, 1)$ is a noise parameter. From Eq. (1.14), these results imply a convergence bound in $O(n^{-a/(2-a)})$ for the quantity $\mathbb{P}\{\widehat{s}_X \neq s^*_X\}$, which is slower than classification learning speeds, due to the inherent complexity of label ranking.

Implications of the analysis. The counterpart of the RMR error Eq. (1.11) for LR would be the following risk:

$$\mathcal{R}(s) := \mathbb{E}[d(s(X), \sigma^*_X)], \tag{1.15}$$

for a ranking rule $s : \mathcal{X} \to \mathfrak{S}_K$. Note that, for any bounded distance d:

$$d(\sigma, \sigma') \leqslant \mathbb{I}\{\sigma \neq \sigma'\} \times \max_{\sigma_0, \sigma_1 \in \mathfrak{S}_K} d(\sigma_0, \sigma_1), \tag{1.16}$$

for any $\sigma, \sigma' \in \mathfrak{S}_K$. Eq. (1.16) implies an extension of our guarantees for $\mathbb{P}\{\widehat{s}_X \neq s^*_X\}$ to the risk Eq. (1.15) of the empirical minimizer. Incidentally, our analysis of LR provides the first generalization bounds for the OVO approach to multiclass classification, by considering the specific case $k = 1$ for the top-k classification guarantees that we provide.

In conclusion, we proposed the new yet natural label ranking (LR) problem, which consists in learning to predict an ordered list of most likely labels from multiclass classification data. While Korba et al. (2018) and Brinker and Hüllermeier (2019) give practical approaches to ranking median regression (RMR) with partial information, our theoretical guarantees are new. Our analysis fits nicely into the usual empirical risk minimization framework and exploits recent results on RMR. A byproduct of our analysis is the first generalization bounds for the OVO approach to multiclass classification.

1.5.2 Selection Bias Correction

In statistical learning problems, the distribution P' of the training data $Z'_1, \dots, Z'_n$ may differ from that of the testing data P. This setup constitutes a particular case of transfer learning (Pan and Yang, 2010; Ben-David et al., 2010; Storkey, 2009; Redko et al., 2019). Notably in facial recognition problems, the training population is frequently not representative of the testing population, as underlined in Wang et al. (2019). Auxiliary information in the form of high level characteristics is often available, such as the nationality associated to a portrait in facial recognition. Our work addresses learning with biased data from the lens of Empirical Risk Minimization (ERM). In that regard, we propose an approach based on importance sampling that deals with: 1) classification problems where class probabilities differ between the training and testing step, 2) situations where the data originates from stratified populations that are represented differently between training and testing, 3) PU learning, the problem of learning with a sample of positive and unlabeled data (du Plessis et al., 2014), 4) and learning with censored data (Fleming and Harrington, 2011). Our analysis is supported by strong empirical evidence for

classification on the ImageNet database (Russakovsky et al., 2014), for which we formed strata information from higher level concepts than the predicted classes.

Weighted Empirical Risk Minimization (WERM). The goal of learning algorithms is generally to find a parameter $\theta \in \Theta$, that minimizes the expected risk $\mathcal{R}(\theta) = \mathbb{E}_P[\ell(\theta, Z)]$ on the testing data, with $\ell : \Theta \times \mathcal{Z} \to \mathbb{R}_+$ is a measurable loss function. To approximate $\mathcal{R}(\theta)$, we propose a weighted estimator $\widetilde{\mathcal{R}}_{w,n}$ on the training data:

$$\widetilde{\mathcal{R}}_{w,n}(\theta) := \frac{1}{n} \sum_{i=1}^{n} w_i \cdot \ell(\theta, Z'_i),$$

where $w = (w_1, \ldots, w_n)$ is a weight vector. When $P' = P$ and $w = (1, \ldots, 1)$, then $\widetilde{\mathcal{R}}_{w,n}(\theta)$ is the usual empirical risk, as well as an unbiased estimator of the expected test risk $\mathcal{R}(\theta)$. When $P' \neq P$ and P is absolutely continuous with respect to P', the importance sampling method (Wasserman, 2010, Section 25.3) introduces optimal weights $w^* := (w_1^*, \ldots, w_n^*)$ that satisfy $w_i^* := \Phi(Z_i)$ for any $i \in \{1, \ldots, n\}$, where $\Phi(z) := (dP/dP')(z)$ for any $z \in \mathcal{Z}$. Since the function Φ denotes the likelihood ratio between P' and P, $\widetilde{\mathcal{R}}_{w^*,n}$ is an unbiased estimator of $\mathcal{R}$.

Generalization guarantees for WERM. We then propose usual generalization guarantees in $O(n^{-1/2})$ that depend on the supremum norm $\|\Phi\|_\infty$ of the likelihood ratio over the input space $\mathcal{Z}$. Our guarantees show that generalization is better when the two distributions P' and P are similar. In the general case, the likelihood ratio Φ is unknown, which limits the applicability of the technique. Additionally, Φ is a function over the input space $\mathcal{Z}$, which makes its estimation impractical. Our work presents situations for which the likelihood function Φ has a simple form, under the assumption that auxiliary information about the relationship between the distributions P' and P is available.

WERM for strata frequencies. In the context multiclass classification, *i.e.* when $Z = (X, Y) \in \mathcal{X} \times \mathcal{Y}$ with $\mathcal{Y} = \{1, \ldots, K\}$ and $K \in \mathbb{N}$ is the number of classes, Φ has a simple form when the proportion $p_k = \mathbb{P}_{Z \sim P}\{Y = k\}$ of each class k in the test dataset is known. In that context, the optimal weights w^* satisfy $w_i^* = p_{Y_i}/p'_{Y_i}$ for any $i \in \{1, \ldots, n\}$, where $p'_k = \mathbb{P}_{P'}\{Y = k\}$ is the train set proportion of class k for any $k \in \{1, \ldots, K\}$. Note that the p'_k's can be estimated from the training data. This reweighting strategy does not depends on a classification objective, but applies to any case with known: proportions of each strata for some stratification in the test distribution, and stratum associated to each sample in the training set.

WERM for Positive Unlabeled (PU) learning. Positive Unlabeled (PU) learning has received increasing attention in the statistical learning literature recently (du Plessis and Sugiyama, 2014; du Plessis et al., 2014, 2015; Kiryo et al., 2017; Bekker et al., 2019). PU learning considers a binary classification problem, *i.e.* $Z = (X, Y) \in \mathcal{X} \times \{-1, +1\}$, and learns a classification decision with a sample of positive instances and one of unlabeled instances. The unlabeled sample is a mixture of both negative and positive data with fixed proportions. Our work shows that, by reweighting the instances of those samples, we obtain an unbiased estimator of the risk $\mathcal{R}$. We provide statistical guarantees for the minimizer of our estimated risk.

Experiments on the ImageNet dataset. Finally, we provide convincing numerical evidence of the effectiveness of weighted ERM on the ImageNet dataset, a database used to benchmark large-scale visual classification algorithms (Russakovsky et al., 2014). ImageNet classes are based on the WordNet lexical database for english (Fellbaum, 1998). As such, they can be regrouped in several higher-level concepts, which constituted the strata of our weighted classification experiment. For example, the class "flamingo" would belong to the stratum "bird". Using high-level information to reweight the training data greatly increases performance on the testing set, measured in terms of top-1 and top-5 classification accuracy.

Ensuring the representativeness of a database may help to obtain fair predictions, but there is growing interest in mechanisms that correct explicitly for inherent biases of training data.

1.5.3 Learning Fair Scoring Functions

Learning a classifier under fairness constraints has received a great deal of attention (Dwork et al., 2012; Zafar et al., 2017a; Donini et al., 2018; Barocas et al., 2019; Williamson and Menon, 2019;

McNamara et al., 2019). However, proposed approaches for fairness in ranking either only correct the scores with a post-processing step (Borkan et al., 2019; Beutel et al., 2019; Zehlike et al., 2017; Celis et al., 2018), or tackle original notions of fairness, such as fairness in exposure for the sequential presentation of rankings (Singh and Joachims, 2018, 2019). Many fairness for ranking papers have proposed different constraints based on the Area Under the ROC Curve (AUC), a standard measure of performance in ranking. Our work first proposes: an unified framework for AUC-based fairness constraints, generalization guarantees for the minimization of a loss that incorporates any of those constraints, as well as a practical optimization procedure for that loss based on gradient descent. Then, we show the limitation of AUC-based fairness constraints, and propose stronger ROC-based constraints. Finally, we prove generalization guarantees and an extension of our gradient-based optimization procedure to learning with the new ROC-based constraints.

Framework for fair bipartite ranking. The standard fairness framework for binary classification considers a triplet of random variables $(X, Y, Z) \in \mathcal{X} \times \{-1, 1\} \times \{0, 1\}$, where X is the input random variable Y is the binary output random variable, and Z encodes the membership to a protected group. In bipartite ranking, one learns a score function $s : \mathcal{X} \to \mathbb{R}$ and evaluates it with respect to where it projects the negatives $Y = -1$ relatively to the positives $Y = +1$ on the real line. In the context of fairness, the influence of Z on the distribution of the scores is important. Hence, we introduce the following conditional distributions of a given score s for any $z \in \{0, 1\}$:

$$H_s(t) := \mathbb{P}\{s(X) \leqslant t \mid Y = -1\} \qquad \text{and} \qquad H_s^{(z)}(t) := \mathbb{P}\{s(X) \leqslant t \mid Y = -1, Z = z\},$$
$$G_s(t) := \mathbb{P}\{s(X) \leqslant t \mid Y = +1\} \qquad \text{and} \qquad G_s^{(z)}(t) := \mathbb{P}\{s(X) \leqslant t \mid Y = +1, Z = z\}.$$

While the ROC curve serves to evaluate the ranking performance of a score function, it is also a general tool to assess the differences between two distributions functions h and g over $\mathbb{R}$. In this context, the ROC curve is known as the probability-probability (PP) plot of h and g. The Area Under the ROC Curve (AUC) is a scalar summary of the ROC curve, that is omnipresent in the ranking literature. It generally serves to evaluate the performance of bipartite ranking algorithms (Clémençon et al., 2008). Formally, the ROC and AUC between the two *c.d.f.*'s h and g, writes:

$$\mathrm{ROC}_{h,g} : \alpha \in [0, 1] \mapsto 1 - g \circ h^{-1}(1 - \alpha) \qquad \text{and} \qquad \mathrm{AUC}_{h,g} := \int_0^1 \mathrm{ROC}_{h,g}(\alpha)\, d\alpha.$$

Differences in the distributions of the positives (or negatives) between the protected groups leads to discrepancies in the error rates of those groups, as observed for facial recognition in Grother and Ngan (2019).

Unified criterion for AUC-based fairness. To correct discrepancies of error rates between protected groups, many authors — principally from the recommendation systems community — have proposed fairness constraints based on the AUC (Beutel et al., 2019; Borkan et al., 2019; Kallus and Zhou, 2019). With $D(s) := (H_s^{(0)}, H_s^{(1)}, G_s^{(0)}, G_s^{(1)})^\top$, those constraints write:

$$\mathrm{AUC}_{\alpha^\top D(s), \beta^\top D(s)} = \mathrm{AUC}_{\alpha'^\top D(s), \beta'^\top D(s)}, \tag{1.17}$$

for different values of $(\alpha, \beta, \alpha', \beta') \in (\mathcal{P})^4$, where $\mathcal{P} = \{v \mid v \in \mathbb{R}_+^4, \mathbf{1}^\top v = 1\}$ denotes the 4-simplex. Our work shows that: if the fairness constraint Eq. (1.17) is satisfied when the distribution $X|Y = y, Z = z$ does not depend on $z \in \{0, 1\}$ for both $y = -1$ and $y = +1$, then Eq. (1.17) writes as the linear combination $\Gamma^\top C(s) = 0$ of five elementary fairness constraints $C(s) = (C_1(s), \ldots, C_5(s))$ with $\Gamma \in \mathbb{R}^5$. Our general definition for AUC-based fairness constraints englobes all proposed measures of fairness based on the AUC and can serve to to derive new ones. More importantly, it paves the way for considering flexible fair scoring approaches based on AUC constraints.

Learning under AUC-based constraints. We integrate the AUC-based fairness constraint as a penalization in the objective function, and maximize the objective:

$$\max_{s \in \mathcal{S}} \mathrm{AUC}_{H_s, G_s} - \lambda \left| \Gamma^\top C(s) \right|, \tag{1.18}$$

on a family of scoring functions $\mathcal{S}$. The parameter λ dictates the trade-off between ranking accuracy and fairness. We provide theoretical guarantees on the generalization error of the

criterion in Eq. (1.18), which are proven with concentration inequalities on the deviations of U-statistics. We propose a gradient-based algorithm that simply optimizes a relaxed version of Eq. (1.18). Introduce the notation $\widetilde{\cdot}$ to denote the relaxation of an empirical estimation AUC with the logistic function $\sigma : x \mapsto 1/(1+e^{-x})$, our relaxation of the loss with a specific AUC constraint writes as:

$$\widetilde{L}_\lambda(s) := \widetilde{\mathrm{AUC}}_{H_s,G_s} - \lambda \cdot c \left(\widetilde{\mathrm{AUC}}_{H_s^{(0)},G_s^{(0)}} - \widetilde{\mathrm{AUC}}_{H_s^{(1)},G_s^{(1)}} \right), \tag{1.19}$$

where $c \in [-1,1]$ is a parameter that changes during the learning process. The parameter c is modified every fixed n_{adapt} gradient steps by a fixed value, depending on whether the difference in AUC's in the constraint of Eq. (1.19) is evaluated as positive or negative on a validation set.

Limitations of AUC**-based constraints.** The equality between two AUC's constrains the distributions involved. Specifically, from the mean value theorem, it imposes an equality point for the ROC curves involved. However, that point is unknown a priori. Many applications — and in particular biometrics — focus on the performance of the system for small false positive rates, *i.e.* specific regions of the ROC curve. Enforcing the equality of ROC curves in those regions implies that the classifiers obtained by thresholding the score are fair in a classification sense. To enforce ROC's to be equal in a specific region, we can focus on a few selected points, which is motivated by discrete approximation results. For these reasons, we introduce ROC-based constraints, which enforce the equality of two ROC curves at specific points.

Learning under ROC**-based constraints.** Consider the ROC curves between the negatives and positives of each sensitive group, *i.e.* $\mathrm{ROC}_{H_s^{(0)},H_s^{(1)}}$ and $\mathrm{ROC}_{G_s^{(0)},G_s^{(1)}}$, their deviation to the diagonal writes:

$$\Delta_{F,\alpha}(s) := \mathrm{ROC}_{F_s^{(0)},F_s^{(1)}}(\alpha) - \alpha,$$

for $F \in \{H,G\}$. Instead of AUC-based fairness constraints, we enforce $\Delta_{F,\alpha}(s) = 0$ with specific values $\alpha_F = [\alpha_F^{(1)}, \ldots, \alpha_F^{(m_F)}]$ of α for any $F \in \{H,G\}$. For that matter, introduce a loss L_Λ that incorporates those as constraints with strength $\lambda_F = \left[\lambda_F^{(1)}, \ldots, \lambda_F^{(m_F)}\right]$ for $F \in \{H,G\}$, as:

$$L_\Lambda(s) := \mathrm{AUC}_{H_s,G_s} - \sum_{k=1}^{m_H} \lambda_H^{(k)} \left| \Delta_{H,\alpha_H^{(k)}}(s) \right| - \sum_{k=1}^{m_G} \lambda_G^{(k)} \left| \Delta_{G,\alpha_G^{(k)}}(s) \right|,$$

where $\Lambda := (\alpha, \lambda_H, \lambda_G)$. We extend our generalization guarantees to ROC-based fairness constraints with an uniform control of the ROC curves. To prove the result, we consider an of empirical processes indexed by the family of points $\alpha \in [0,1]$. We also propose a strategy for optimization that is analogous to the one used for Eq. (1.19). Our strategy features threshold parameters, that are modified in the same way as c.

Experimental results. We provide strong empirical evidence of our approach. Fig. 1.2 gives an overview of the fairness versus accuracy trade-offs achieved with our methods on specific data. As more restrictive fairness constraints are introduced in our experiments, we observe simultaneously that: 1) the area under ROC_{H_s,G_s} — *i.e.* the AUC as a ranking accuracy measure — decreases, and 2) the conditional distributions of the score change coherently with our fairness constraints. Notice that, for high scores, the difference in the conditional distribution of the score between sensitive groups is much higher with AUC-based constraints than with ROC-based constraints.

In conclusion, we have proposed new approaches to tackle algorithmic fairness in ranking. First, we regrouped AUC-based constraints under a single general definition. Then, we proposed theoretical guarantees, and a practical method for learning with any of those constraints by gradient descent. Afterwards, we pointed at the limitations of AUC-based constraints, and proposed a ROC-based approach that corresponds better to operational settings. Finally, we extended our generalization guarantees and practical optimization method to the new and more flexible ROC-based constraint.

1.6 Perspectives

In conclusion, the thesis addresses important problems in biometrics from the point of view of statistical learning theory. Our work proposes original ideas for these problems, and supports

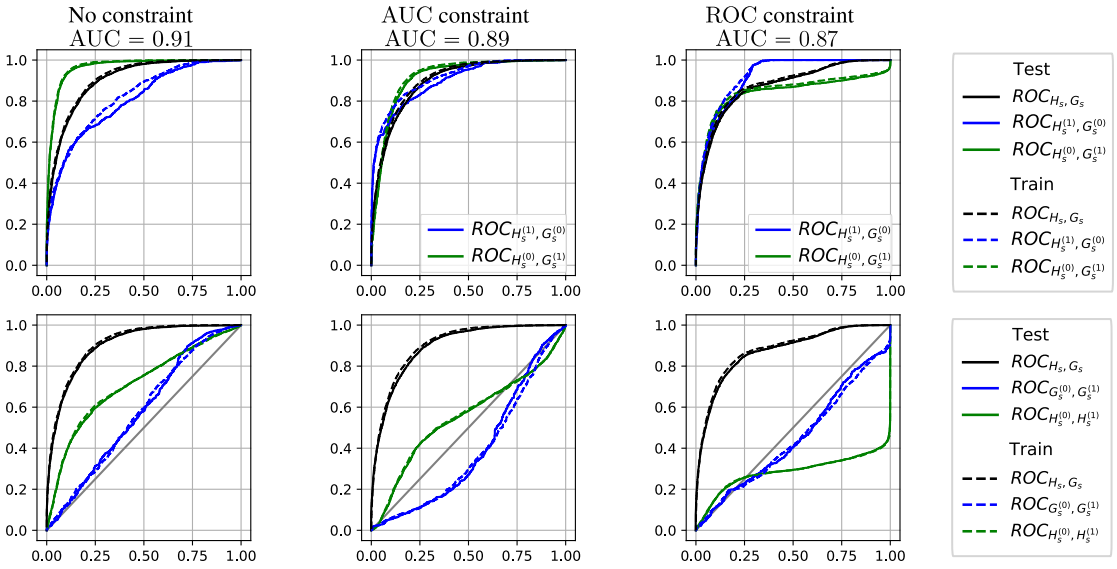


Figure 1.2: ROC curves obtained respectively: by learning a score without constraint, with the AUC-based constraint $\mathrm{AUC}_{H_s^{(0)},G_s^{(1)}} = \mathrm{AUC}_{H_s^{(1)},G_s^{(0)}}$, and with a ROC-based constraint parameterized to obtain $\mathrm{ROC}_{F_s^{(0)},F_s^{(1)}}(\alpha) = \alpha$ for any $\alpha \in [0, 1/4]$ and any $F \in \{H, G\}$.

those ideas with theoretical results. These results can be interpreted as security guarantees, that hold under probabilistic assumptions. Our work is a much-needed answer to the rapidly increasing volume of experimental machine learning literature that biometrics researchers have to follow. Biometric identification, and facial recognition in particular, incarnate many machine learning topics simultaneously, such as pairwise learning, sample bias or ranking. For this reason, we considered stylized versions of those problems, as their simultaneous examination would obscure our discourse, and runs the risk of being disregarded as anecdotal by the machine learning community. The richness of the topics tackled by the thesis is a result of this imperative. Broadening this spectrum could be envisaged, for example by considering the extension of ranking the best criteria (Menon and Williamson, 2016, Section 9) to the similarity ranking problem presented above.

From a biometrics perspective, the most important perspective in this thesis is to realize the potential impact of the methods presented, by providing strong empirical evidence of their relevance in practical settings. Indeed, while the rapid adoption of machine learning techniques by private companies has boosted the growth of the field, it also directed most of the attention to papers that propose unequivocal solutions to specific industrial problems. A notorious example for face recognition is Schroff et al. (2015). In that context, the promotion of this work will require finding and presenting pedagogically large-scale experiments that address precisely practical use-cases, which is a promising direction for future work.

Finally, we could extend the different topics considered in the thesis. In the context of fairness for ranking, one of the limitations of our analysis comes from the absence of an analytical expression for the optimal score function under a fairness condition. However, it is provided in the case of fair regression by Chzhen et al. (2020) for example. In bipartite ranking, overcoming this hurdle would pave the way for an extension of the partition-based algorithms of Clémençon and Vayatis (2009) to learning under fairness constraints. Another possibility concerns the extension of the techniques presented here to the case of similarity ranking. Indeed, that extension gives a framework that matches operational considerations in biometrics very closely, and would be justified by the current interest in methods to explicitly correct biases for facial recognition. The experimental component of that work would be supported by the availability of well-suited face databases (Wang et al., 2019). Another possibility is to address the limitations of our work on weighted empirical risk minimization, by considering cases where the nature of the difference between the train and test set is not covered by our work. For example, in Sugiyama et al.

(2007), the authors propose to estimate the likelihood ratio using a small sample of the test set as auxiliary information. Besides the above examples, more extensions of each topic in the thesis could be considered.

In conclusion, the richness of the issues that arise in biometrics is fertile grounds for new theory and new practices in machine learning. This richness spurred the creation of this thesis and can inspire future research.

Part I

Preliminaries

Chapter 2

Statistical Learning Theory

Summary: This chapter is a short introduction to statistical learning theory, which focuses on the derivation of finite-sample bounds on the generalization error for the binary classification problem. All of the generalization guarantees proven in the thesis are modeled after those presented in this chapter. Precisely, understanding this chapter is essential to derive all of the theoretical results of Part III (Chapter 8, Chapter 9 and Chapter 10), to all bounds of Part I (Chapter 3 and Chapter 4), and to our similarity ranking guarantees (Part II - Chapter 5). In the chapter, we first present the probabilistic setting associated to binary classification and the associated risks. To bound the excess risk of the empirical minimizer, we introduce: a few basic concentration inequalities, as well as strategies to relate the excess risk to the complexity of a proposed class. Those results enable us to derive generalization bounds that are independent of the distribution of the data. Next, we address the looseness of those bounds with more involved concentration inequalities. Combined with noise assumptions on the distribution of the data, those inequalities imply fast and distribution-dependent bounds on the generalization error. As the problems introduced in the thesis can not be studied in the framework presented in this chapter, we delve into the necessary notions to extend those results. Finally, we detail the involvement throughout the thesis of the results presented here. We refer the reader to Bousquet et al. (2003), Boucheron et al. (2005) and Györfi (2002) for more detailed introductions on statistical learning theory.

2.1 Introduction

Statistical learning theory is described in Bousquet et al. (2003) (Chapter 1) as a mathematical framework for studying the problem of inference, that is of gaining knowledge about some modeled phenomenon from data. It can be considered as a mathematical tool that justifies the soundness of performing machine learning, *i.e.* of using data collected in some specific context, *e.g.* in the past, to make decisions in another context, *e.g.* in the future. For that to be sensible, one has to make assumptions as to how the data collected — called *training data* — relates to the *testing data*, and those assumptions are referred to as *inductive bias*. Those assumptions express naturally in a probabilistic language. A common probabilistic assumption is that all observations are independent and originate from the same distribution (*i.i.d.*). Since risk measures in machine learning are usually mathematical expectations of a loss function, the *i.i.d.* assumption implies that natural estimators of that true risk write as averages of the loss on the training data.

Statistical estimation has extensively studied the convergence of standard averages. While classic statistical literature focuses on their asymptotic properties as the sample size grows to infinity (van der Vaart, 2000), the limitation of that analysis for finite samples has lead authors to quantify the random fluctuations of finite averages of *i.i.d.* variables with *concentration inequalities* (Boucheron et al., 2013). As a result, recent theoretical results in statistical machine learning originate from bounds on the deviation between a finite-sample estimate and a target

value, that hold with some probability (Boucheron et al., 2005).

In machine learning, authors have shown that the difference between the true risk of a minimizer of the empirical risk and that of a reference solution can also be bounded using concentration inequalities. In this thesis, we refer to those bounds as learning bounds/rates, or as *generalization bounds/guarantees/rates*, This chapter details the derivation of such bounds for binary classification, the flagship problem in machine learning.

Firstly, Section 2.2 details the context and mathematical framework necessary to introduce learning bounds. Secondly, Section 2.3 shows how to derive an uniform learning bound, *i.e.* bounds that do not depend on the distribution of the data. Those bounds hold for the most complicated distributions P possible, but tend to be quite loose for smooth distributions. Thirdly, Section 2.4 responds to this weakness by deriving tighter bounds under mild assumptions on the data. Finally, Section 2.5 discusses the extension of such results to the more complicated statistics involved in the thesis and indexes the uses of the tools and results of this chapter in the thesis.

2.2 A Probabilistic Setting for Binary Classification

Statistical learning considers data that belongs to a input space $\mathcal{X}$, also called *feature space*, that is often a d-dimensional Euclidean space, *i.e.* $\mathcal{X} = \mathbb{R}^d$. That information serves to gain knowledge about some value that belongs to an output space $\mathcal{Y}$. To study the relationship between the collected *features* and the output value, those are modeled as a pair of dependent random variables *r.v.* $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ that follow some unknown joint probability distribution P defined on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$. We refer to the two first chapters of Shorack (2000) for a thorough definition of these mathematical objects. The random variable Y is referred to as the *output r.v.*, while X is referred to as the *input r.v.*. We introduce the distribution of X as F, called the *marginal distribution* of X (Papoulis, 1965, Chapter 6, page 171). Mathematically, statistical learning focuses seeks to recover some information about distribution of the output *r.v.* Y for any possible value of the input *r.v.* X, *i.e.* about the conditional random variable $Y|X = x$ for any $x \in A$, with $A \subset \mathcal{X}$ such that (*s.t.*) $\mathbb{P}\{X \in A\} = 1$. The probability distribution P can be decomposed as $F \times P_{Y|X}$, with $P_{Y|X}$ the conditional distribution of Y given X. The conditional distribution $P_{Y|X}$ is referred to as the *posterior probability* of Y, and its existence is justified in Shorack (2000) (Chapter 8, Theorem 5.2).

Learning with binary output Y. Many learning problems assume that the output random variable Y is binary, *i.e.* $\mathcal{Y} = \{-1, +1\}$, such as binary classification or bipartite ranking, that last problem being the subject of Chapter 3. Under that assumption, the posterior probability $P_{Y|X}$ is summarized by the mapping η with:

$$\begin{aligned} \eta : \ & \mathcal{X} \to [0, 1], \\ & x \mapsto \mathbb{P}\{Y = +1 \mid X = x\}, \end{aligned}$$

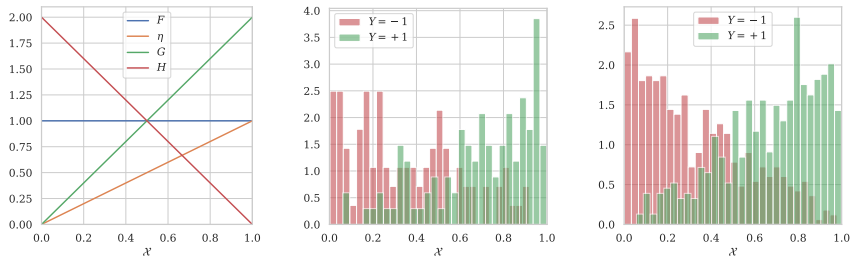
which is referred to as the *regression function* or as the *Bayes score* in the bipartite ranking literature. The distribution P is completely characterized by the pair (F, η). Another way to specify P is to define the proportion of positive instances $p = \mathbb{P}\{Y = +1\}$ and the distribution of the conditional random variables $X|Y = y$, denoted as G for $y = +1$ and as H for $y = -1$. Then, the two characterizations are related by the identities $G = \eta F/p$, $H = (1 - \eta)F/(1 - p)$ and $p = \int \eta(x) dF(x)$. They imply $F = pG + (1 - p)H$ and $\eta/(1 - \eta) = pG/((1 - p)H)$.

Example 2.1. *To illustrate this section, we provide an example with $\mathcal{X} = [0, 1]$. We introduce a posterior probability η, that depends on the parameter $a \in (0, 1)$: $\forall x \in [0, 1]$,*

$$\eta(x) = \frac{1}{2} + \frac{1}{2} sgn(2x - 1)|1 - 2x|^{\frac{1-a}{a}}.$$

See Fig. 2.1(a) for a representation of the characterizations of P for $a = 1/2$.

Binary classification. We now focus on the binary classification problem, and refer to Section 2.5 for a discussion on the extension of those results to other settings. The goal of binary classification



(a) Representation of P (b) Dist. $n = 200$ (c) Dist. $n = 1000$

Figure 2.1: Representation of Example 2.1 with $a = 1/2$. the left-hand side figure represents the different possible parametrizations of P, namely (F, η) and (p, G, H) (we omit $p = 1/2$ here). As F, H, G are all absolute continuous measures, we represented the Radon-Nikodym derivative *w.r.t.* the Lebesgue measure of those distributions (Shorack, 2000, Section 2, Chapter 4). On the other hand, the two right-hand side figures represent the approximations of G and H that we obtain using histograms of n independent realizations of (X, Y) with $n = 200$ and $n = 1000$.

is to find a function $g : \mathcal{X} \to \mathcal{Y} = \{-1, +1\}$ in a class of measurable candidate functions $\mathcal{G}$ that minimizes the misclassification error $R(g)$, with:

$$R(g) := \mathbb{P}\{g(X) \neq Y\}.$$

The following proposition characterizes the best classifier in the set of all measurable functions.

Definition 2.2 (Bayes classifier)**.**
The Bayes classifier *is defined as the function $g^* : \mathcal{X} \to \mathcal{Y}$ s.t.:*

$$\forall x \in \mathcal{X}, \quad g^*(x) := 2 \cdot \mathbb{I}\{\eta(x) > 1/2\} - 1.$$

For any measurable $g : \mathcal{X} \to \mathcal{Y}$, we have $R(g^*) \leqslant R(g)$.
Furthermore, we have the following decomposition of the excess risk:

$$R(g) - R(g^*) = \mathbb{E}[|2\eta(X) - 1| \cdot \mathbb{I}\{g(X) \neq g^*(X)\}]. \tag{2.1}$$

Proof. Note that:

$$R(g) = \mathbb{E}[\mathbb{E}[\mathbb{I}\{g(X) \neq Y\}|X]] = \mathbb{E}[\eta(X) + (1 - 2\eta(X))\mathbb{I}\{g(X) = +1\}].$$

Since: for any $x \in \mathcal{X}$ and $g : \mathcal{X} \to \mathcal{Y}$,

$$(1 - 2\eta(x))\left(\mathbb{I}\{g(x) = +1\} - \mathbb{I}\{g^*(x) = +1\}\right) = |2\eta(x) - 1| \cdot \mathbb{I}\{g(x) \neq g^*(x)\},$$

we have that $R(g) \geqslant R(g^*)$ for any measurable g. □

The Bayes classifier makes classification mistakes unless the classes are separable. Indeed, note that $R(g^*) = \mathbb{E}[\min(\eta(X), 1-\eta(X))]$, which means that g^* has a positive risk unless $\eta(X) \in \{0, 1\}$ *a.s.*. A sensible goal when selecting a candidate function in $\mathcal{G}$ is thus to find $g \in \mathcal{G}$ with $R(g)$ as close as possible to $R(g^*)$, which is measured by the notion of *excess risk*, defined as:

$$E(g) := R(g) - R(g^*).$$

Remark 2.3. *The Bayes classifier for Example 2.1 is the simple function $x \mapsto \mathbb{I}\{x \geqslant 1/2\}$. In that case, it belongs to the simple proposed class of functions $\mathcal{G}_{stump}$ that consists of all of the* decision stumps *on $[0, 1]$, i.e. $\mathcal{G}_{stump} = \{x \mapsto a \cdot sgn(x - h) \mid h \in [0, 1], a \in \{-1, +1\}\}$.*

Approximation error. Unlike in Remark 2.3, in usual settings, the class $\mathcal{G}$ is not large enough to contain the Bayes classifier g^*. Introduce any minimizer $g^\dagger$ of R in $\mathcal{G}$, *i.e.* $g^\dagger \in \arg\min_{g \in \mathcal{G}} R(g)$.

Then, the gap in performance between $g^\dagger$ and g^* is quantified by the *approximation error*, defined as:

$$R(g^\dagger) - R(g^*) \geqslant 0.$$

Controlling the approximation error generally requires an assumption on the smoothness of the target function. To introduce this type of assumption, the regression function η can for example be assumed to belongs to a *Sobolev space* (Jost, 2011, Appendix A.1). While results exist in the case of regression Smale and Zhou (2003), bounding the approximation error in classification remains an open problem.

Generalization error. Since P is unknown, we cannot directly measure the risk R. The general idea of statistical learning is to use a notion of empirical risk to approximate R. Introduce a sequence of n *i.i.d.* pairs with the same probability distribution as (X, Y), *i.e.* $\mathcal{D}_n = \{(X_i, Y_i)\}_{i=1}^n \overset{i.i.d.}{\sim} P$, we define the empirical risk R_n on the data $\mathcal{D}_n$ as:

$$R_n(g) := \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\{g(X_i) \neq Y_i\}. \tag{2.2}$$

We denote by g_n the minimizer of R_n over $\mathcal{G}$. The precision of the approximation of R by R_n — and as a result, the choice of g_n — depends on the difference between the empirical distribution and the true distribution, illustrated in Fig. 2.1. A quantity of interest is the risk $R(g_n)$ of the minimizer of the empirical risk, as it quantifies the performance of the classifier learned from data. Observe, that, since $R_n(g_n) \leqslant R_n(g^\dagger)$ by definition of g_n, we have:

$$\begin{aligned} E(g_n) &\leqslant R(g^\dagger) - R(g^*) + R(g_n) - R_n(g_n) + R_n(g^\dagger) - R(g^\dagger), \\ &\leqslant R(g^\dagger) - R(g^*) + 2 \sup_{g \in \mathcal{G}} |R_n(g) - R(g)|. \end{aligned} \tag{2.3}$$

Eq. (2.3) shows that one can bound the excess of risk of the empirical minimizer by the approximation error plus the uniform deviation $\mathcal{E}_n(\mathcal{G}) := \sup_{g \in \mathcal{G}} |R_n(g) - R(g)|$ of $R_n(g) - R(g)$ over $\mathcal{G}$, which quantifies the *generalization error*. The bound can be seen as a formalization of the bias-variance tradeoff presented in Domingos (2012), where the bias term corresponds to the approximation error, while the variance corresponds to the generalization error. When we increase the size of the proposed family $\mathcal{G}$, the approximation error decreases, while the generalization error increases.

PAC-bounds. PAC (*probably approximately correct*)-style bounds are bounds that are satisfied for any outcome $\omega \in A_\delta$, for some $A_\delta \subset \Omega$ with $\mathbb{P}\{A_\delta\} \geqslant 1 - \delta$. The learning bounds presented throughout the thesis consist in PAC bounds for similar types of uniform deviations over diverse families of functions. All of the bounds of the thesis concern the generalization error.

2.3 Uniform Learning Bounds

Most learning bounds are based on the *Chernoff bound*, which bounds *the tail* $\mathbb{P}\{Z > t\}$ of a *r.v.* Z using its *moment generating function* $\lambda \mapsto \mathbb{E}[e^{\lambda Z}]$, as shown below.

Proposition 2.4 (The Chernoff bound)**.** *(Boucheron et al., 2013, section 2.1)*
For any real-valued r.v. Z and any $t \in \mathbb{R}$,

$$\mathbb{P}\{Z > t\} \leqslant \inf_{\lambda > 0} \mathbb{E}\left[e^{\lambda(Z-t)}\right] = \inf_{\lambda > 0} \left(\mathbb{E}\left[e^{\lambda(Z - \mathbb{E}[Z])}\right] \cdot e^{\lambda(\mathbb{E}[Z]-t)}\right). \tag{2.4}$$

Proof. The proof is a simple combination of an increasing transformation with Markov's inequality. Let $\lambda > 0$, then $t \mapsto e^{\lambda t}$ is increasing and the simple bound $\mathbb{I}\{x > 1\} \leqslant x$ holds for any $x \in \mathbb{R}$, hence:

$$\mathbb{P}\{Z > t\} = \mathbb{P}\left\{e^{\lambda Z} > e^{\lambda t}\right\} = \mathbb{E}\left[\mathbb{I}\{e^{\lambda(Z-t)} > 1\}\right] \leqslant \mathbb{E}\left[e^{\lambda(Z-t)}\right], \tag{2.5}$$

and minimizing the right-hand side of Eq. (2.5) gives the result. □

By substituting Z with $\mathcal{E}_n(\mathcal{G})$ in Eq. (2.4), we see that controlling the uniform deviation can relies on a combination of two separate results. Specifically, we rely on a result on $\mathbb{E}[Z]$, a term that measures the expressivity of the proposed class $\mathcal{G}$, and on a term $Z - \mathbb{E}[Z]$, a term that measures the robustness of the process of selecting g from data. For sensible families $\mathcal{G}$, both of these quantities decrease to zero when n increases, but with different convergence speeds. Typically, the parameter λ is selected to balance the speeds of convergence of the two terms.

Section 2.3.1 presents basic concentration inequalities, that are required to study the terms $\mathbb{E}[Z]$ and $Z - \mathbb{E}[Z]$ where $Z = \mathcal{E}_n(\mathcal{G})$. The convergence of $\mathbb{E}[Z]$ results from the limited complexity of the proposed class of functions $\mathcal{G}$, and we present the related mathematical formalization in Section 2.3.2. Finally, Section 2.3.3 exploits the results of Section 2.3.2 and the McDiarmid's inequality of Section 2.3.1 to conclude. It presents a first generalization bound.

2.3.1 Basic Concentration Inequalities

Three distinct but related types of presentation exist for any result of Section 2.3.1. First, two equivalent formulations exist for presenting bounds on the tail $\mathbb{P}\{Z > t\}$ of a *r.v.* Z. Precisely, consider a bound on the tail of a *r.v.* Z that writes $\mathbb{P}\{Z > t\} \leqslant f(t)$, and define t_δ as the solution in t of $\delta = f(t)$ with some $\delta > 0$. The bound $\mathbb{P}\{Z > t\} \leqslant f(t)$ is then equivalent to a simple inequality of the form $Z \leqslant t_\delta$, that holds with probability (*w.p.*) larger than ($\geqslant$) $1 - \delta$. For example, Corollary 2.6 below is provided as such. Another type is as a bound on the *moment generating function* $t \mapsto \mathbb{E}[e^{tZ}]$ of a *r.v.* Z, which relates directly to the tail of Z with Chernoff's bound (Proposition 2.4). The moment generating bound implies tail bounds, and all theorems can be stated in any of the three forms.

The most famous concentration inequality is *Hoeffding's inequality* (Györfi, 2002, Theorem 1.2), a PAC bound of the order $O(n^{-1/2})$ on the deviation of the mean of bounded *i.i.d. r.v.*'s from their expectation. It stems from a simple bound on the tail $\mathbb{P}\{Z > t\}$ of a bounded random variable Z, a property introduced in Györfi (2002) (Lemma 1.2) and presented in Lemma 2.5 below. The proof of Lemma 2.5 is the beginning of the proof of the Hoeffding inequality of Györfi (2002) (Lemma 1.2).

Lemma 2.5 (almost Hoeffding's inequality)**.**
Let Z be a random variable with $\mathbb{E}[Z] = 0$ and $a \leqslant Z \leqslant b$ a.s.. Then for $t > 0$:

$$\mathbb{E}\left[e^{tZ}\right] \leqslant e^{t^2(b-a)^2/8}.$$

Corollary 2.6 (Hoeffding's inequality)**.**
Let $Z_1, \dots Z_n$ be $n \in \mathbb{N}^*$ i.i.d. r.v. such that $a \leqslant Z_1 \leqslant b$. Denote their mean by $\bar{Z}_n = (1/n)\sum_{i=1}^n Z_i$. Then for any $\epsilon > 0$, we have with probability (w.p) at least $1 - \epsilon$,

$$\bar{Z}_n - \mathbb{E}[\bar{Z}_n] \leqslant (b-a)\sqrt{\frac{\log(1/\delta)}{2n}}.$$

Proof. By Chernoff's bound, see Proposition 2.4, we have:

$$\mathbb{P}\{\bar{Z}_n - \mathbb{E}[\bar{Z}_n] > t\} \leqslant \inf_{\lambda > 0} e^{-t\lambda}\mathbb{E}\left[e^{\lambda(\bar{Z}_n - \mathbb{E}[\bar{Z}_n])}\right]. \tag{2.6}$$

The independence between the Z_i's, see Shorack (2000) (Chapter 8, Theorem 1.1), followed by Lemma 2.5, implies that:

$$\mathbb{E}\left[e^{\lambda(\bar{Z}_n - \mathbb{E}[\bar{Z}_n])}\right] = \prod_{i=1}^n \mathbb{E}\left[e^{(\lambda/n)(Z_1 - \mathbb{E}[Z_1])}\right] = e^{\lambda^2(b-a)^2/(8n)}.$$

Minimizing the *r.h.s.* of Eq. (2.6) in λ gives:

$$\mathbb{P}\{\bar{Z}_n - \mathbb{E}[\bar{Z}_n] > t\} \leqslant e^{-2t^2 n/(b-a)^2}. \tag{2.7}$$

Inverting that bound, *i.e.* solving in t such that δ is equal to the *r.h.s.* of Eq. (2.7) implies the result. □

We can bound the tail of $\left|\bar{Z}_n - \mathbb{E}[\bar{Z}_n]\right|$ using the *union bound* — also referred to as the *subadditivity of measures* or *Boole's inequality* (Shorack, 2000, Chapter 1, Proposition 1.2) — between Corollary 2.6 and its application to the family $-Z_1, \dots, -Z_n$. It writes:

$$\mathbb{P}\{\left|\bar{Z}_n - \mathbb{E}[\bar{Z}_n]\right| > t\} \leqslant \mathbb{P}\{\bar{Z}_n - \mathbb{E}[\bar{Z}_n] > t\} + \mathbb{P}\{\mathbb{E}[\bar{Z}_n] - \bar{Z}_n > t\} \leqslant 2e^{-2t^2 n/(b-a)^2}.$$

The Hoeffding inequality on its own implies guarantees for finite families of functions $\mathcal{G}$, as presented in Bousquet et al. (2003) (section 3.4) and shown in Corollary 2.7 below:

Corollary 2.7. *(Bousquet et al., 2003, section 3.4)*
Assume that $\mathcal{G}$ is of cardinality N. Then, we have that: for any $\delta > 0$ and $n \in \mathbb{N}^*$, w.p. $\geqslant 1 - \delta$,

$$R(g_n) - R(g^\dagger) \leqslant \sqrt{\frac{2}{n} \log\left(\frac{2N}{\delta}\right)}.$$

Proof. The union bound implies:

$$\begin{aligned}\mathbb{P}\left\{\max_{g\in\mathcal{G}} |R_n(g) - R(g)| > t\right\} &= 1 - \mathbb{P}\left\{\bigcup_{g\in\mathcal{G}} \left(|R_n(g) - R(g)| \leqslant t\right)\right\}, \\ &\geqslant 1 - \sum_{g\in\mathcal{G}} \mathbb{P}\left\{|R_n(g) - R(g)| \leqslant t\right\}.\end{aligned}$$

Applying Corollary 2.6 gives:

$$\begin{aligned}\mathbb{P}\{\max_{g\in\mathcal{G}} |R_n(g) - R(g)| \leqslant t\} &\leqslant \sum_{g\in\mathcal{G}} \mathbb{P}\{|R_n(g) - R(g)| \leqslant t\}, \\ &\leqslant 2Ne^{-2t^2 n},\end{aligned}$$

and inverting the bound gives the result. □

The result presented in Lemma 2.5 implies bounds on the expectation of the maximum of a family of random variables, as shown in Lemma 2.8 below. It is useful when considering minimizers of the empirical risk, to upper-bound the value of $\mathbb{E}[\mathcal{E}_n(\mathcal{G})]$ presented in Section 2.1.

Lemma 2.8. *(Györfi, 2002, Lemma 1.3)*
Let $\sigma > 0$, and assume that $Z_1, \dots, Z_m$ are $m \in \mathbb{N}^*$ real-valued r.v.'s such that for any $t > 0$ and $1 \leqslant j \leqslant m$, $\mathbb{E}[e^{tZ_j}] \leqslant e^{t^2\sigma^2/2}$, then:

$$\mathbb{E}\left[\max_{j\leqslant m} Z_j\right] \leqslant \sigma\sqrt{\log(2m)}.$$

We refer to Györfi (2002) (Lemma 1.3) for the proof, as well as the remark that the tail of $\max_{j\leqslant m}|Z_j|$ is bounded by applying Lemma 2.8 to the family $\{Z_1, -Z_1, \dots, Z_m, -Z_m\}$ of $2m$ elements. Now that we have presented inequalities to deal with the term $\mathbb{E}[\mathcal{E}_n(\mathcal{G})]$, we introduce one that controls the deviation of $\mathcal{E}_n(\mathcal{G})$ from its mean, and refer to Györfi (2002) (Lemma 1.4) for the proof.

Theorem 2.9 (McDiarmid's inequality)**.** *(Györfi, 2002, Lemma 1.4)*
Let $Z_1, \dots, Z_n$ be a collection of $n \in \mathbb{N}^*$ i.i.d. random variables. Assume that some function $g : Im(Z_1)^n \to \mathbb{R}$ satisfies the bounded difference assumption:

$$\sup_{\substack{z_1,\dots,z_n \in Im(Z_1) \\ z_i' \in Im(Z_1)}} \left|g(z_1, \dots z_n) - g(z_1, \dots, z_i', \dots, z_n)\right| \leqslant c_i, \qquad 1 \leqslant i \leqslant n,$$

then, for all $t > 0$:

$$\mathbb{P}\{g(Z_1, \dots, Z_n) - \mathbb{E}[g(Z_1, \dots, Z_n)] > t\} \leqslant \exp\left\{-\frac{2t^2}{\sum_{i=1}^n c_i^2}\right\}.$$

2.3.2 Complexity of Classes of Functions

This section presents several ways to control the complexity of a class of proposed functions $\mathcal{G}$, referred to as capacity measures in Boucheron et al. (2005). Those include *Rademacher averages*, the concept of *shatter coefficient* and *VC dimension*, as well as *covering numbers*. First, we introduce the notion of Rademacher average and relate it to the notion of VC dimension. For an introduction to covering numbers for statistical learning theory, we refer to Boucheron et al. (2005) (Section 5.1).

Definition 2.10 (Rademacher average)**.**
Let $\mathcal{G}$ be a set of binary classifiers, i.e. $\forall g \in \mathcal{G}, g : \mathcal{X} \to \{-1, +1\}$. The Rademacher average of $\mathcal{G}$ is defined as:

$$\mathfrak{R}_n(\mathcal{G}) := \mathbb{E}\left[\sup_{g\in\mathcal{G}}\left|\frac{1}{n}\sum_{i=1}^{n}\sigma_i\mathbb{I}\{g(X_i)\neq Y_i\}\right| \mid (X_i,Y_i)_{i=1}^n\right],$$

where $\sigma = (\sigma_i)_{i=1}^n$ is a set of n i.i.d. Rademacher variables*, i.e. $\mathbb{P}\{\sigma_i = +1\} = \mathbb{P}\{\sigma_i = -1\} = 1/2$ for any $i \in \{1, \dots, n\}$.*

The Rademacher average measures the capacity of a family of functions to fit random noise, and bounds the expectation of $\mathcal{E}_n(\mathcal{G})$, as shown in Proposition 2.11 below.

Proposition 2.11. *For any class of binary classifiers $\mathcal{G}$ and any $n \in \mathbb{N}^*$, we have:*

$$\mathbb{E}\left[\sup_{g\in\mathcal{G}}|R_n(g) - R(g)|\right] \leqslant 2\mathbb{E}\left[\mathfrak{R}_n(\mathcal{G})\right].$$

Proof. The proof is based on a trick called *symmetrization*, as presented for example in Bousquet et al. (2003) (Section 4.4). It requires the introduction of an unobserved sample $\mathcal{D}'_n = (X'_i s, Y'_i s)_{i=1}^n \overset{i.i.d.}{\sim} P$, referred to as *ghost sample* or *virtual sample*. We denote by R'_n the estimate of the same form as Eq. (2.2), but estimated on $\mathcal{D}'_n$.

Using the simple property $\sup \mathbb{E}[\cdot] \leqslant \mathbb{E}[\sup(\cdot)]$, followed by the triangle inequality, gives:

$$\begin{aligned}
\mathbb{E}\left[\mathcal{E}_n(\mathcal{G})\right] &= \mathbb{E}\left[\sup_{g\in\mathcal{G}}\left|R_n(g) - \mathbb{E}[R'_n(g)]\right|_{\mathcal{G}}\right] \leqslant \mathbb{E}\left[\sup_{g\in\mathcal{G}}\left|R_n(g) - R'_n(g)\right|\right], \\
&\leqslant \mathbb{E}\left[\mathbb{E}\left[\sup_{g\in\mathcal{G}}\left|\frac{1}{n}\sum_{i=1}^{n}\sigma_i\left(\mathbb{I}\{g(X_i)\neq Y_i\} - \mathbb{I}\{g(X'_i)\neq Y'_i\}\right)\right| \mid (X_i,Y_i)_{i=1}^n\right]\right], \\
&\leqslant 2\mathbb{E}\left[\mathfrak{R}_n(\mathcal{G})\right].
\end{aligned}$$

□

Remark 2.12. *If X is a continuous random variable and $\mathcal{G}$ is the family of all measurable functions, we have: $\mathfrak{R}_n(\mathcal{G}) \geqslant 1/2$ a.s. It follows from the observation that: we have almost surely.*

$$\mathfrak{R}_n(\mathcal{G}) = \frac{1}{n}\cdot\mathbb{E}\left[\max\left(\sum_{i=1}^{n}\sigma_i, n - \sum_{i=1}^{n}\sigma_i\right)\right] \geqslant \frac{1}{2},$$

The result implies that the bound in Proposition 2.11 is uninformative.

On the other hand, an estimation of the Rademacher average of $\mathcal{G}_{stump}$ (see Remark 2.3) gives, by averaging over 200 random draws of the σ_i's, values of 0.13 *for $n = 100$ and* 0.04 *for $n = 1000$. It shows that, for simple families of functions, the Rademacher average decreases quickly as n grows.*

The notion of Rademacher average is a standard analysis tool and is involved in many important results. Those include the derivation of sharp bounds for the supremum of empirical processes (Györfi, 2002, Section 1.4.6) or the derivation of data-dependent bounds (Boucheron et al., 2005, Theorem 3.2). However, it does not relate directly to simple properties of the proposed family.

Introduce the *shatter coefficient* $\mathbb{S}_{\mathcal{G}}(n)$ of a family of functions $\mathcal{G}$ as the maximal number of ways to split any collection of n elements in $\mathcal{X} \times \mathcal{Y}$ with $\mathcal{G}$. Formally:

$$\mathbb{S}_{\mathcal{G}}(n) = \sup_{\{(x_i,y_i)\}_{i=1}^n \in (\mathcal{X}\times\mathcal{Y})^n} |\{(\mathbb{I}\{g(x_1) \neq y_1\}, \dots, \mathbb{I}\{g(x_n) \neq y_n\}) \mid g \in \mathcal{G}\}|. \tag{2.8}$$

Note that the shatter coefficient does not depend on the sample $\mathcal{D}_n$, and is bounded by 2^n. A set of n points is said to be shattered by a family $\mathcal{G}$ if its shatter coefficient is equal to 2^n, with the convention that the empty set is always shattered.

Lemma 2.8 relates the Rademacher average to the shatter coefficient, which in turn enables us to replace the supremum over $\mathcal{G}$ with a maximum over a set of $\mathbb{S}_{\mathcal{G}}(n)$ vectors, as detailed in Corollary 2.13 below.

Corollary 2.13 (Massart's Lemma)**.** *(Györfi, 2002, see the proof of Theorem 1.9)*
Let $\mathcal{G}$ be a class of measurable family of binary classifiers, i.e. for any $g \in \mathcal{G}$, $g : \mathcal{X} \to \{-1, +1\}$. For any $n \in \mathbb{N}^*$, we have:

$$\mathfrak{R}_n(\mathcal{G}) \leqslant \sqrt{\frac{2\log(2\mathbb{S}_{\mathcal{G}}(2n))}{n}}.$$

The combination of Proposition 2.11 and Corollary 2.13 is referred to as the *Vapnik-Chervonenkis inequality* (Györfi, 2002, Theorem 1.9, page 13). When the class $\mathcal{G}$ is finite of size N, then $\mathbb{S}_{\mathcal{G}}(n) \leqslant N$, which implies a bound on $\mathbb{E}\left[\sup_{g\in\mathcal{G}} |R_n - R|\right]$ that decreases in $O(n^{-1/2})$. The shatter coefficient gives a simple bound on the Rademacher average by removing the expectation. However, it depends on n and does not have an intuitive form for simple families of functions. On the other hand, the VC-dimension is intuitive and summarizes the complexity of a class of functions by a single coefficient.

Definition 2.14 (VC-dimension)**.** *(Bousquet et al., 2003, Definition 2, page 189)*
The VC-dimension V of a proposed class $\mathcal{G}$ is defined as the maximum number of points that a class can shatter, i.e. as the largest $n \in \mathbb{N}^*$ such that:

$$\mathbb{S}_{\mathcal{G}}(n) = 2^n.$$

Any class of functions with finite VC-dimension is called a VC-class of functions.

Remark 2.15. *The VC-dimension of the set of decision stumps introduced in Remark 2.3 is 2, as one can always separate two distinct points, but not the set $\{(0.2, +1), (0.3, -1), (0.4, +1)\} \subset \mathcal{X}\times\mathcal{Y}$. We refer to Mohri et al. (2012) (Section 3.3, pages 41 to 45) for many examples of proposed families and a discussion on their VC-dimension.*

The following lemma relates the VC-dimension to the shatter coefficient.

Lemma 2.16 (Vapnik-Chervonenkis-Sauer-Shelah lemma)**.**
(Bousquet et al., 2003, Lemma 1, page 190)
Assume that $\mathcal{G}$ is a VC-class of functions with VC-dimension V. Then: for all $n \in \mathbb{N}$,

$$\mathbb{S}_{\mathcal{G}}(n) \leqslant \sum_{k=0}^{V} \binom{n}{k} \leqslant (n+1)^V.$$

The last inequality of Lemma 2.16 is simply a consequence of the binomial theorem:

$$(n+1)^V = \sum_{k=1}^{V} \frac{n^k}{k!} \frac{V!}{(V-k)!} \geqslant \sum_{k=1}^{V} \frac{n^k}{k!} \geqslant \sum_{k=1}^{V} \frac{n!}{(n-k)!}\frac{1}{k!} = \sum_{k=1}^{n} \binom{n}{k},$$

as detailed in Györfi (2002) (Theorem 1.13, page 19). Even though Györfi (2002) also contains a proof of the first inequality, we refer here to the clearer proof of Shalev-Shwartz and Ben-David (2014) (page 74, lemma 6.10).

All of the results of Section 2.3.2 come into play to control the quantity $\mathbb{E}[\mathcal{E}_n(\mathcal{G})]$ introduced in Proposition 2.4. They are necessary to derive a first, basic learning bound, presented in the section below.

2.3.3 Uniform Generalization Bounds

By joining the results of Section 2.3.2 to control $\mathbb{E}[\mathcal{E}_n(\mathcal{G})]$ (Proposition 2.4) with Theorem 2.9 for the deviation of $\mathcal{E}_n(\mathcal{G})$ to its mean, we can derive a simple uniform learning bound.

First, combining Proposition 2.11, Corollary 2.13 and Lemma 2.16 gives Proposition 2.17 below.

Proposition 2.17. *Assume that $\mathcal{G}$ is a VC-class of functions with VC-dimension V, then, for any $n \in \mathbb{N}^*$, we have:*

$$\mathbb{E}[\mathcal{E}_n(\mathcal{G})] \leqslant \sqrt{\frac{8\log(2) + 8V\log(1+2n)}{n}}.$$

Proposition 2.17 gives a bound on $\mathbb{E}[\mathcal{E}_n(\mathcal{G})]$. We now deal with the other term in Proposition 2.4. The following corollary is a simple application of Theorem 2.9 to the random variable ϵ.

Corollary 2.18. *Let $\mathcal{G}$ be a class of binarybe classifiers. With $Z = \mathcal{E}_n(\mathcal{G}) = \sup_{g\in\mathcal{G}}|R_n(g) - R(g)|$, we have that, for any $n \in \mathbb{N}^*$:*

$$\mathbb{E}\left[e^{\lambda(Z-\mathbb{E}[Z])}\right] \leqslant e^{\lambda^2 n/8}.$$

Proof. Introduce R'_n as the estimator where for a fixed $i \in \{1,\dots,n\}$, (X_i, Y_i) was substituted by some value $(X'_i, Y'_i) \in \mathcal{X}\times\mathcal{Y}$ in the estimator of the risk R_n, as well as $\mathcal{E}'_n(\mathcal{G}) := \sup_{g\in\mathcal{G}}|R'_n(g) - R(g)|$. Simple triangle inequalities for the absolute value, combined with usual properties of the supremum imply:

$$\begin{aligned}\left|\mathcal{E}_n(\mathcal{G}) - \mathcal{E}'_n(\mathcal{G})\right| &\leqslant \sup_{g\in\mathcal{G}}\left|R_n(g) - R'_n(g)\right|,\\ &\leqslant \frac{1}{n}\sup_{g\in\mathcal{G}}\left|\mathbb{I}\{g(X_i)\neq Y_i\} - \mathbb{I}\{g(X'_i)\neq Y'_i\}\right| \leqslant \frac{1}{n},\end{aligned}$$

which implies the result with Theorem 2.9. □

Joining the results of Proposition 2.4 with those of Proposition 2.17 and Corollary 2.18 gives Proposition 2.19 below. Note the usual order in $O(n^{-1/2})$ for statistical learning theory, as the term in $\log(n)$ is negligible in front of the other terms. A more advanced technique for controlling the complexity of $\mathcal{G}$, called *chaining*, gives sharper bounds in $O(n^{-1/2})$ without the logarithmic term $\log(n)$. Here, we report that result without the proof, but refer to Györfi (2002) (Theorem 1.16 and 1.17) for it.

Proposition 2.19. *(Boucheron et al., 2005, Theorem 3.4)*
Let $\mathcal{G}$ be a class of binary classifiers. Assume that $\mathcal{G}$ has finite VC dimension V, then: for any $\delta > 0$ and $n \in \mathbb{N}^*$, w.p. $\geqslant 1-\delta$:

$$R(g_n) - R(g^\dagger) \leqslant \sqrt{\frac{2\log(1/\delta)}{n}} + 2\sqrt{\frac{8\log(2) + 8V\log(1+2n)}{n}}.$$

A more refined version of this inequality can be proven using a chaining argument, as explained in Boucheron et al. (2005) (Theorem 3.4 therein) and proven in Györfi (2002) (Section 1.4.6). It states that: for any $\delta > 0$ and $n \in \mathbb{N}^*$, w.p $\geqslant 1-\delta$,

$$R(g_n) - R(g^\dagger) \leqslant \sqrt{\frac{2\log(1/\delta)}{n}} + 2C\sqrt{\frac{V}{n}},$$

where $C > 0$ is an universal constant.

Proof. It follows from Eq. (2.3), that:

$$\mathbb{P}\{R(g_n) - R(g^\dagger) > t\} \leqslant \mathbb{P}\{2\cdot\mathcal{E}_n(\mathcal{G}) > t\}.$$

Combining Proposition 2.17 and Corollary 2.18, then minimizing the bound in Proposition 2.4 in λ gives, with $C_{V,n} = 8\log(2) + 8V\log(1+2n)$:

$$\mathbb{P}\{\mathcal{E}_n(\mathcal{G}) > t\} \leqslant \exp\left\{-2\left(\sqrt{n}t - \sqrt{C_{V,n}}\right)^2\right\}. \tag{2.9}$$

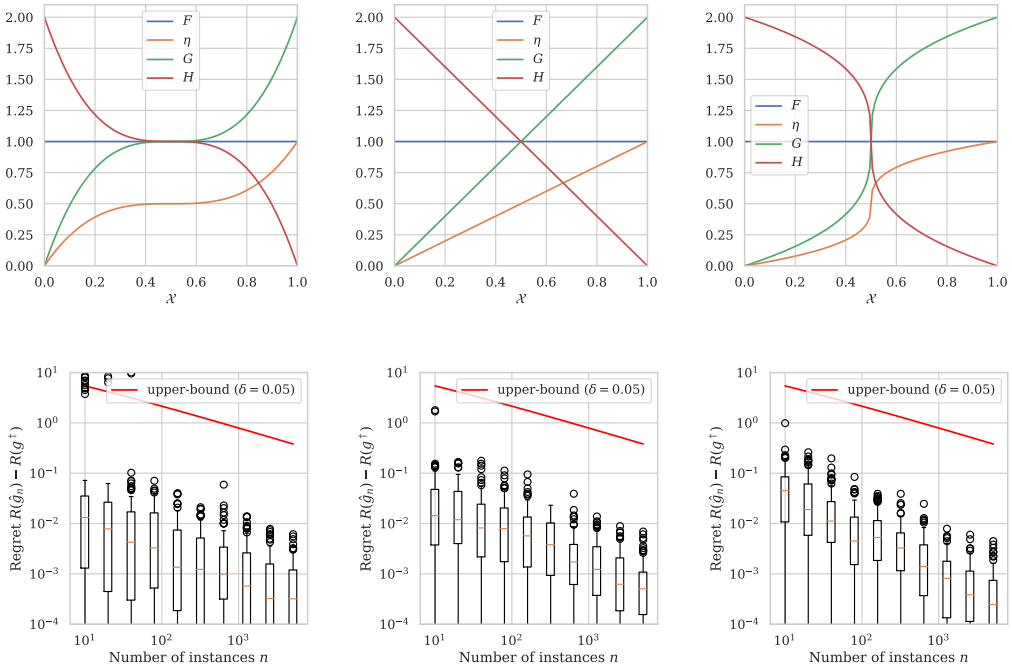
Inverting the bound gives the result. □

While the bound holds for any possible probability distributions P, it is loose for smooth distributions. Indeed, in many cases, the uniform deviation of the empirical risk $\sup_{g\in\mathcal{G}}|R_n(g) - R(g)|$ over $\mathcal{G}$ is a very poor bound for $R(g_n) - R(g^\dagger)$, as mentioned in Boucheron et al. (2005) (Section 5.2). The looseness of the bound of this section is illustrated in Remark 2.20.

Remark 2.20. *For the distributions defined in Example 2.1, we illustrate in Fig. 2.2 the bound of Proposition 2.19 for the family of decision stumps in $\mathbb{R}$ as proposed family $\mathcal{G}$.*

Firstly, observe that the bound is very loose. Our distribution is not very complicated and our proposed family $\mathcal{G}_{stump}$ is very favorable for the problem at hand, which may explain that fact.

Secondly, while the order of the slope is the same for any distribution P, the error rate seems to decrease faster when a is close to 1 than when a is close to 0. Precisely, the learning rate for easily separated distributions seems faster than $O(n^{-1/2})$. It is proven in Section 2.4.



(d) $a = 1/4$ (e) $a = 1/2$ (f) $a = 3/4$

Figure 2.2: Boxplot of the regret over 100 generations of a dataset $\mathcal{D}_n$, for three selected values of the parameter a and different values of n, compared to the bound derived in Proposition 2.19. See Remark 2.20 for more details.

2.4 Faster Learning Bounds

This section introduces the derivation of learning rates faster than $O(n^{-1/2})$ under a noise assumption on the distribution of the data. Firstly, we introduce concentration inequalities that involve the variance of random variables. Secondly, we introduce noise conditions, and explain how they induce an upper-bound on the variance of the excess loss. Finally, we show that combining these two first results gives a fixed-point inequality, which implies fast learning rates.

The methodology presented here is limited to finite classes of functions $\mathcal{G}$ and to problems for which the Bayes classifier g^* belongs to $\mathcal{G}$. In this section, we also discuss the generalization of those results to general classes of functions of controlled complexity, as well as to those that do not contain the Bayes classifier.

We introduce the family $\mathcal{F}$ as the family of the *excess losses* of all elements in $\mathcal{G}$. Formally, the

family of functions $\mathcal{F}$ satisfies that: for any $f \in \mathcal{F}$ there exist some $g \in \mathcal{G}$, such that,

$$f(X,Y) := \mathbb{I}\{g(X) \neq Y\} - \mathbb{I}\{g^*(X) \neq Y\}.$$

The bound of Proposition 2.19 relies on the uniform (for any $g \in \mathcal{G}$) boundedness of the random variable $f(X,Y)$. However, in many situations, as we choose a candidate g that approaches g^*, the variance of $f(X,Y)$ may be very small in front of the range of $f(X,Y)$. Hence, the next section presents concentrations inequalities that use the variance of random variables. They serve to derive tighter bounds based on hypotheses on the distribution of the data.

2.4.1 Sharper Concentration Inequalities

The derivation of fast learning bounds relies on concentration inequalities that involve the variance of random variables. The simplest one is Bernstein's inequality (Boucheron et al., 2013, Corollary 2.11). Its proof is featured in Boucheron et al. (2013) (Theorem 2.10). We recall the simplest formulation of the inequality here.

Theorem 2.21 (Bernstein's inequality)**.**
Let $Z_1, \ldots, Z_n$ be $n \in \mathbb{N}^*$ i.i.d. real-valued r.v.'s with zero mean, and assume that $|Z_i| \leqslant c$ a.s. Let $\sigma^2 := Var(Z_1)$ and $\bar{Z}_n := (1/n)\sum_{i=1}^n Z_i$. For any $\delta > 0$, we have that: w.p. $\geqslant 1-\delta$,

$$\bar{Z}_n \leqslant \frac{2c\log(1/\delta)}{3n} + \sqrt{\frac{2\sigma^2\log(1/\delta)}{n}}.$$

Popoviciu's inequality on variances gives an upper bound on the variance of a bounded random variable (Egozcue and García, 2018). Precisely, it states that the variance σ^2 of a *r.v.* Z such that $m \leqslant Z \leqslant M$ is smaller than $(M-m)^2/4$. Plugging this result into Theorem 2.21 gives a similar result as Corollary 2.6.

Bernstein's inequality does not give a guarantee on the supremum of a family of functions, but on an empirical mean, as does Hoeffding's inequality. A direct consequence of Hoeffding's inequality were guarantees on a finite family of functions, as shown in Corollary 2.7. Similarly, only the derivation of guarantees on a finite family of functions are a direct application of the combination of Bernstein's inequality and noise conditions on P.

Extending those results to more general classes of functions requires *Talagrand's inequality*, presented in Theorem 2.22 below. We refer to Boucheron et al. (2005) (Section 5.3, Theorem 5.4) for a simple presentation of the result, and to Boucheron et al. (2013) (Theorem 7.9, page 225) for the proof. We denote by P_n the empirical distribution associated to $\mathcal{D}_n$. For any $f \in \mathcal{F}$, we have $Pf := \mathbb{E}[f(X,Y)]$ and $P_n f := (1/n)\sum_{i=1}^n f(X_i, Y_i)$

Theorem 2.22 (Talagrand's inequality)**.**
Let $b > 0$ and set $\mathcal{F}$ a family of functions $\mathcal{X} \to \mathbb{R}$. Assume that, for any $f \in \mathcal{F}$, $Pf - f \leqslant b$. Then for all $n \in \mathbb{N}^*$: w.p $\geqslant 1-\delta$,

$$\sup_{f\in\mathcal{F}}(Pf - P_n f) \leqslant 2\mathbb{E}\left[\sup_{f\in\mathcal{F}}(Pf - P_n f)\right] + \sqrt{\frac{2(\sup_{f\in\mathcal{F}} Var(f))\log(1/\delta)}{n}} + \frac{4b\log(1/\delta)}{3n}.$$

The formulations of Bernstein's inequality and Talagrand's inequality are very similar. Their main difference is that the first deals with simple means, while the second handles the supremum of empirical processes. Note that the first term in the bound of Talagrand's inequality can be controlled in Section 2.3.2.

These inequalities will prove useful when combined with a clever upper bound of the variance of the excess loss, *i.e.* the variance of $f(X,Y)$. Conveniently, that quantity is upper-bounded by a function of the excess risk $R(g) - R(g^*)$, using the noise conditions introduced in the following section.

2.4.2 Noise Conditions

The simplest noise assumption to derive faster bounds is called *Massart's noise condition* (Boucheron et al., 2005, page 340). It assumes that there is always a clear separation between the

two classes for the classification decision. Formally, it assumes that $\eta(X)$ is always far from 1/2.

Assumption 2.23 (Massart's noise assumption)**.**
There exists $h > 0$, s.t.

$$|2\eta(X) - 1| \geqslant h \quad a.s.$$

A less restrictive assumption is called the *Mammen-Tsybakov noise condition*, see (Boucheron et al., 2005, page 340). It gives a bound on the proportion of $\mathcal{X}$, as measured by the distribution of X, for which the classification decision is hard to take. It is parameterized by a parameter $a \in [0, 1]$. If a distribution satisfies it for a close to 1, the two classes are easily separated. If it only satisfies the bound when a is close to 0, the two classes are hard to separate.

Assumption 2.24 (Mammen-Tsybakov's noise assumption)**.**
There exists $B > 0$ and $a \in [0, 1]$, such that:

$$\mathbb{P}\left\{|2\eta(X) - 1| \leqslant t\right\} \leqslant Bt^{\frac{a}{1-a}}.$$

Remark 2.25. *The distribution (F, η) introduced in Example 2.1, and parameterized by a, satisfies the Mammen-Tsybakov noise condition with parameters $B = 1$ and a.*

Observe that Mammen's noise condition is void when $a = 0$. Also, it is implied with $a = 1$ by Massart's noise condition. Both noise assumptions imply a convenient bound on the variance of a function $f \in \mathcal{F}$. as a consequence of a second moment bound (a bound of the form $\mathrm{Var}(Z) \leqslant \mathbb{E}[Z^2]$), combined with the decomposition of the excess risk presented in Eq. (2.1). Precisely, since:

$$\mathrm{Var}(f(X, Y)) \leqslant \mathbb{E}[\mathbb{I}\{g(X) \neq g^*(X)\}], \tag{2.10}$$

and $R(g) - R(g^*) = \mathbb{E}\left[|2\eta(X) - 1|\mathbb{I}\{g(X) \neq g^*(X)\right]$, Massart's inequality implies:

$$\mathrm{Var}(f(X, Y)) \leqslant \frac{1}{h}\mathbb{E}[|2\eta(X) - 1|\mathbb{I}\{g(X) \neq g^*(X)\}] = \frac{1}{h}\left(R(g) - R(g^*)\right).$$

Similarly, the Mammen-Tsybakov noise assumption implies the following proposition.

Proposition 2.26. *Assume that Assumption 2.24 is true, then, for any $f \in \mathcal{F}$, there exists c that depends on only of B and a, s.t.*

$$Var(f(X, Y)) \leqslant c(R(g) - R(g^*))^a$$

The proof relies on Eq. (2.10) and an equivalent formulation of the Mammen-Tsybakov noise assumption. The equivalent formulation states that: there exists $c > 0$ and $a \in [0, 1]$, such that:

$$\mathbb{P}\{g(X) \neq g^*(X)\} \leqslant c\left(R(g) - R(g^*)\right)^a. \tag{2.11}$$

Eq. (2.11) above is implied by Assumption 2.24, as proven in Bousquet et al. (2003) (Definition 7), which contains other equivalent formulations of Proposition 2.26.

Bousquet et al. (2003) (Section 5.2) contains or implies bot]h of the preceding results. Used in conjunction with the concentration inequalities of Section 2.4.1, those results enable us to derive fast and distribution-dependent generalization bounds by solving a fixed-point equation.

2.4.3 Distribution-dependent Generalization Bounds

The reasoning presented here can be found in Bousquet et al. (2003) (Section 5.2), but is more detailed here. The union bound of applications of Bernstein's inequality on each element of a finite family of N elements, gives that: for any $\delta > 0$, $w.p \geqslant 1 - \delta$, we have simultaneously, that for any $g \in \mathcal{G}$ that corresponds to $f \in \mathcal{F}$,

$$R(g) - R(g^*) \leqslant R_n(g) - R_n(g^*) + \frac{2c\log(N/\delta)}{3n} + \sqrt{\frac{2\mathrm{Var}(f(X, Y))\log(N/\delta)}{n}}. \tag{2.12}$$

Assume that g^* belongs to the proposed class, *i.e.* that $g^* \in \mathcal{G}$. Combining Eq. (2.12) with the implications of the Mammen-Tsybakov noise assumption (Proposition 2.26) for the empirical risk minimizer ($g = g_n$) implies that: for any $\delta > 0$, $w.p \geqslant 1 - \delta$,

$$R(g_n) - R(g^*) \leqslant \frac{2c\log(N/\delta)}{3n} + (R(g_n) - R(g^*))^{a/2}\sqrt{\frac{2c\log(N/\delta)}{n}}, \tag{2.13}$$

which is a fixed point inequality in $R(g_n) - R(g^*)$. To derive an upper-bound on $R(g_n) - R(g^*)$ from Eq. (2.13), we use Lemma 2.27 below. It relies on a generalization of Mignotte (1992) (Theorem 4.2), which gives an upper bound on the maximum of the absolute values of the roots of a polynomial.

Lemma 2.27. *(Cucker and Smale, 2002, Lemma 7)*
Let $(c_1, c_2, s, q) \in \mathbb{R}_+^{*4}$ be real positive numbers. Then the equation $x^s - c_1 x^q - c_2 = 0$ has a unique positive zero x^*. In addition, $x^* \leqslant \max\{(2c_1)^{1/(s-q)}, (2c_2)^{1/s}\}$.

Using Lemma 2.27, we can prove a fast learning bound for finite families of functions.

Proposition 2.28. *Assume that $\mathcal{G}$ is a finite family of N functions and that the Mammen-Tsybakov noise assumption Eq. (2.11) is satisfied with constant parameter c and noise parameter a. For any $\delta > 0$, we have that, for any $n \in \mathbb{N}^*$: w.p. $\geqslant 1 - \delta$,*

$$R(g_n) - R(g^*) \leqslant \left(\frac{8c\log(N/\delta)}{n}\right)^{1/(2-a)} + \frac{8c\log(N/\delta)}{3n}.$$

Proof. A straightforward application of Lemma 2.27 to Eq. (2.13), followed by the bound $\max(x, y) \leqslant x + y$ for any $x, y > 0$ implies:

$$\begin{aligned} R(g_n) - R(g^*) &\leqslant \max\left(\left(\frac{8c\log(N/\delta)}{n}\right)^{1/(2-a)}, \frac{8c\log(N/\delta)}{3n}\right), \\ &\leqslant \left(\frac{8c\log(N/\delta)}{n}\right)^{1/(2-a)} + \frac{8c\log(N/\delta)}{3n}. \end{aligned}$$

□

Proposition 2.28 gives a learning bound of the order $O(n^{-1/(2-a)})$, under a noise condition parameterized by $a \in [0, 1]$. When a is close to 0, we recover the order $O(n^{-1/2})$ derived in Section 2.3.3. On the other hand, when a is close to zero, we approach the fast rate $O(n^{-1})$. The fast bounds are illustrated in Remark 2.29. The generalization of Proposition 2.28 to non-finite families of functions is more involved, and is a consequence of Talagrand's inequality (Theorem 2.22). We refer to Boucheron et al. (2005) (Section 5.3) for that extension. The analysis presented there extends to the case where g^* does not belong to $\mathcal{G}$. The extension is detailed in Boucheron et al. (2005) (Section 5.3.5).

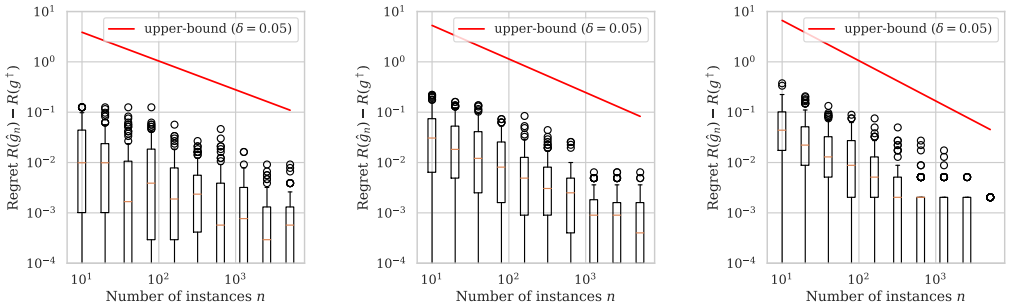
Remark 2.29. *With the distributions defined in Example 2.1, we illustrate the value of the bound in Proposition 2.28 for the finite proposed family of functions $\mathcal{G} = \{i/100 \mid i \in \{0, \ldots, 1\}$ in Fig. 2.2.*

Observe in Fig. 2.3 that the learning rate of the fast bound seems much more aligned with the empirical learning rate than the slow learning rate of Section 2.3.

In real problems, the distribution of the data is unknown. Hence, the satisfaction of a noise assumption can not be verified. Still, fast learning speeds justify the looseness of the bounds derived in Section 2.3, constitute strong evidence of the soundness of learning from empirical data, and give possible tighter generalization bounds.

2.5 Connections to Present Work

The results in this thesis are modeled after those of this chapter. However, the majority of our work — specifically Chapter 5 and Chapter 7 of Part II and Chapter 10 of Part III — tackles the



(a) $a = 1/4$ (b) $a = 1/2$ (c) $a = 3/4$

Figure 2.3: Boxplot of the regret over 100 generations of a dataset $\mathcal{D}_n$, for three selected values of the parameter a and different values of n, compared to the fast bound derived in Proposition 2.28. See Remark 2.29 for more details.

problems of scoring or similarity learning. Scoring concerns learning a score function $s : \mathcal{X} \to \mathbb{R}$, while similarity learning concerns learning a function $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Hence, both bipartite ranking and similarity learning focus on real-valued functions. To derive generalization results for real-valued functions, tools that control the complexity of a real-valued family of functions $\mathcal{S}$ are required. In this section, we present notions that extend the results of the chapter to real-valued functions, and detail the implications of the chapter for our results in the thesis.

VC-major classes of functions. To quantify the complexity of classes of real-valued functions, the notion of VC-major class is used throughout the thesis. It builds on top of the notion of superlevel-set at some level $t \in \mathbb{R}$, defined as $\{x \in \mathcal{X} \mid s(x) > t\}$ for some $s : \mathcal{X} \to \mathbb{R}$.

Definition 2.30 (VC-major class)**.** *(van der Vaart and Wellner, 1996, Section 2.4.6) A class of functions $\mathcal{S}$ such that $\forall s \in \mathcal{S}$, $s : \mathcal{X} \to \mathbb{R}$ is called VC-major if all superlevel-sets of all functions in $\mathcal{S}$ form a VC-class of sets. Formally, $\mathcal{S}$ is a VC-major class if and only if:*

$$\{\{x \in \mathcal{X} \mid s(x) > t\} \mid s \in \mathcal{F}, t \in \mathbb{R}\} \text{ is a VC-class of sets.}$$

Other notions to control the variations of empirical processes indexed by real-valued functions include the notion of VC-subgraph class (van der Vaart and Wellner, 1996, Section 2.6.2). A class of functions $\mathcal{S}$ is a VC-subgraph class simply if the subgraphs of all functions in $\mathcal{S}$, *i.e.* $\{(x,t) \mid t < s(x)\}_{s \in \mathcal{S}}$, form a VC-class of sets. Note that, unlike VC-major classes the set of all subgraphs is only indexed by $\mathcal{S}$. We refer to Dudley (1999) (Theorem 4.7.1) for more details on the relationship between VC-major classes and VC-subgraph classes.

Sauer's lemma for VC-major classes. If a major class of functions $\mathcal{S}$ is bounded, any function $s \in \mathcal{S}$ writes as a (possibly infinite) weighted sum of its major sets, as presented in van der Vaart and Wellner (1996) (Lemma 2.6.13). That observation implies the proposition below, an extension of Proposition 2.17 to VC-major classes.

Proposition 2.31. *Assume that $\mathcal{S}$ is a VC-major class of functions bounded by 1, of bounded VC-dimension V. For $n \in \mathbb{N}^*$, introduce P_n as the empirical measure associated to a sample $\{X_i\}_{i=1}^n \overset{i.i.d}{\sim} P$, i.e. $P_n := (1/n) \sum_{i=1}^n \delta_{X_i}$. Then:*

$$\mathbb{E}\left[\sup_{s \in \mathcal{S}} |P_n s - P s|\right] \leqslant \sqrt{\frac{8 \log(2) + 8V \log(1 + 2n)}{n}}.$$

The extension of VC-properties to general types of functions follows generally from permanence properties of the Rademacher average, presented in Boucheron et al. (2005) (Theorem 3.3).

Implications of this chapter. The theory presented in this chapter is the foundation for much of our theoretical contributions. Precisely, it is involved in the two other chapters of the preliminaries, *i.e.* Chapter 3 and Chapter 4, as well as in Chapter 5 of Part II and in all chapters of Part III.

In Part I, the uniform generalization bounds (Proposition 2.19) and the intermediary steps of the fast generalization bound (Proposition 2.28) are essential to the proof of the generalization results for the pointwise ROC optimization problem considered in Chapter 3, as well as to the guarantees for the TreeRank algorithm. Indeed, the results for bipartite ranking of Chapter 3 are modeled on the framework provided here for finite-sample statistical guarantees in usual statistical learning theory. In Chapter 4, the uniform generalization bound (Proposition 2.19) is extended to the case of U-statistics, as well as to their sampling-based approximation.

In Part II, Chapter 5 extends to the case of similarity ranking the pointwise ROC optimization and TreeRank guarantees proven for bipartite ranking in Chapter 3. The proofs are based on an extension for U-statistics of the results used for bipartite ranking in Chapter 3. Precisely, it is based on the variant for real-valued U-statistics of the learning bounds Proposition 2.19 and Proposition 2.28. That variant is implied by a combination of Proposition 2.31 and Chapter 4. Incidentally, the properties of U-statistics enable us to derive fast bounds with weaker assumptions than in the bipartite ranking case.

In Part III, the main result of our work on label ranking (Chapter 8) incorporates a variant of our fast learning bounds (Proposition 2.28) applied to a binary classification problem. Precisely, that variant is proven with Talagrand's inequality (Theorem 2.22) and thus holds for general classes of functions. The statistical guarantees of our work on weighted empirical minimization (Chapter 9) rely on the exact same tools as Proposition 2.19, but feature weighted samples. Finally, our work on fair scoring functions (Chapter 10) presents generalization bounds for learning real-valued score functions. To maximize pairwise functionals, those invoke results on the supremum deviation of U-statistics, an extension of Proposition 2.19 presented in Chapter 4. In other cases, it involves the usual results on standard empirical processes that imply Proposition 2.19.

Chapter 3

Selected Ranking Problems

Summary: This chapter is a short introduction to selected problems that deal with ranking data in machine learning. Its purpose is to provide the preliminaries to our theoretical results: on similarity ranking (Chapter 5) and on fair ranking (Chapter 10). Additionally, it puts in perspective the results involved in our analysis of label ranking (Chapter 8), and provides the intuitions for our propositions for practical similarity ranking (Chapter 7). Ranking data in a broad sense is for example encountered in: bipartite ranking/scoring, ranking aggregation and learning-to-rank. Bipartite ranking considers a set of elements associated to a binary label, and seeks to associate a score to each instance, such that those with label $+1$ have higher score than those with label -1. A typical application is credit scoring. Ranking aggregation is the problem of summarizing several individual rankings by a representative ranking. Finally, learning-to-rank associates an ordering over candidate elements to a query, and is the usual theoretical framework for studying search engines algorithms. In this chapter, the majority of the results focus on bipartite ranking. Those results are finite-sample bounds, and an extension of the analysis of Chapter 2. Precisely, we present at length the ROC criterion, pointwise ROC optimization and the theoretical guarantees for the TreeRank algorithm. Then, we briefly present the ranking aggregation problem, and precise its relationship to probabilistic models for ranking. Finally, we detail the implications throughout the thesis of the results presented here. We refer to Menon and Williamson (2016) for a more detailed overview of bipartite ranking.

3.1 Introduction

Rankings are a very natural way to provide information about a set of elements, which explains the ubiquity of machine learning problems related to finding ordered sets of those elements. The thesis relates to three different settings connected to that idea, with some papers making interesting connections between them. Those settings are: *bipartite ranking* or *scoring*, *ranking aggregation* and *learning-to-rank*.

The bipartite ranking/scoring problem considers a set of elements associated to a binary label, and seeks to rank those with label $+1$ higher than those with label -1. This is generally performed by learning a mapping from the input space to the real line, called a *score function*. For example, this setting applies to any situation with a fixed budget to assign to the observations that are the most likely to be positive, *e.g.* in credit scoring. On the other hand, the binary classification problem described in Chapter 2 recovers a set of relevant instances, but does not compare positive instances among one another. We refer to Menon and Williamson (2016) for an overview of bipartite ranking.

Ranking aggregation seeks a consensus between several observed orderings of items. It relies on finding an ordering that summarizes best all observed orderings. A flagship problem of ranking

aggregation is social choice theory, a theoretical framework for combining individual opinions. We refer to Korba (2018) (Part 1) for an introduction to ranking aggregation.

Finally, learning-to-rank refers to learning models that return an ordered list of instances in a database, that are relevant for a specific query. It can be performed by learning a notion of similarity between queries and candidate instances, and returning a list of items ordered by decreasing similarity with the query. The design of search engines is an example of direct application of this setting. We refer to Liu (2011) for an introduction to learning-to-rank.

In this section, we will focus first on bipartite ranking (Section 3.2) and then briefly introduce ranking aggregation (Section 3.3). Our introduction to bipartite ranking introduces the ROC curve, presents the problem of pointwise ROC optimization, also known as Neyman-Pearson classification, and finally recalls the theoretical guarantees for the TreeRank algorithm for bipartite ranking. The ranking aggregation section swiftly presents the problem and focuses on its relation to probabilistic models for ranking. Section 3.4 details the implications of the results of this chapter on the original work of the thesis.

3.2 Bipartite Ranking

3.2.1 Introduction

Bipartite ranking relies on the same probabilistic framework as binary classification, presented in Chapter 2. Specifically, one considers a random pair $(X, Y) \sim P$ in $\mathcal{X} \times \mathcal{Y}$ where $\mathcal{X}$ is the input space, and $\mathcal{Y} = \{-1, +1\}$ is the output space. Generally the input space $\mathcal{X}$ is a d-dimensional Euclidean space $\mathbb{R}^d$. As stated in Section 2.1, the distribution of the pair (X, Y) is completely summarized by the pair (F, η), where F is the marginal distribution of X and η is the posterior probability $\eta(x) := \mathbb{P}\{Y = 1 \mid X = x\}$, as well as by the triplet (p, H, G), where p is the proportion of positive instances $p = \mathbb{P}\{Y = +1\}$, H is the distribution of $X|Y = -1$ and G that of $X|Y = +1$. An usual approach to bipartite ranking is to project items on the real line with a score function $s : \mathcal{X} \to \mathbb{R}$, and to return the ordering of those items by decreasing score.

To quantify the capacity of the score s to split positive from negative instances, we introduce the distribution of the random variables $s(X)|Y = -1$ and $s(X)|Y = +1$ as, respectively:

$$H_s(t) := H(s(X) \leqslant t) = \mathbb{P}\{s(X) \leqslant t \mid Y = -1\},$$
$$G_s(t) := G(s(X) \leqslant t) = \mathbb{P}\{s(X) \leqslant t \mid Y = +1\}.$$

With $\bar{F} = 1 - F$ the survival function associated to a distribution F, the quantities $\overline{H_s}(t)$ and $\overline{G_s}(t)$ are known respectively as the *false positive rate (FPR)* and the *true positive rate (TPR)* at threshold t. Considering that an instance x is classified as positive when $s(x) > t$, they correspond respectively to the proportion of negative instances wrongly classified as positive and the proportion of positive instances rightly classified as positive.

The goal of bipartite ranking is to order items by a notion of quality, quantified by the probability that an item $x \in \mathcal{X}$ is positive, which is simply the posterior probability $\eta(x)$. While one could consider solving the bipartite ranking problem by constructing an approximation of η from data, bipartite ranking is only interested in recovering the order induced by the posterior probability, not in the values of the posterior probability. A widespread criterion to measure the accuracy of the order induced by a score function is the *Receiver Operating Characteristic curve* (ROC) (Definition 4 in Clémençon and Vayatis (2009)) a functional criterion defined as the parametric curve:

$$\begin{aligned} &\mathbb{R} \to [0, 1]^2, \\ &t \mapsto \left(\bar{H}_s(t), \bar{G}_s(t)\right). \end{aligned} \tag{3.1}$$

The above definition implies that the ROC of an increasing transform $T \circ s$ of the score s is the same as that of the score s. Since $T \circ s$ induces the same order as s, the ROC only evaluates the induced order on $\text{supp}(F)$ of the score s, and not its value.

We formalize the order over a random sample $\mathcal{D}_n = \{X_i\}_{i=1}^n \overset{i.i.d.}{\sim} P$ with a permutation $\sigma \in \mathfrak{S}_n$, as the order induced by s on $\mathcal{D}_n$, with the highest score on top and ties broken arbitrarily.

Specifically, the permutation σ maps each $i \in \{1, \dots, n\}$ to the position of $s(X_i)$ in the list of decreasing $s(X_j)$'s, which writes:

$$s(X_{\sigma^{-1}(1)}) \geqslant \cdots \geqslant s(X_{\sigma^{-1}(n)}),$$

The permutation σ^{-1} is the inverse of σ, hence $\sigma^{-1}(i)$ is the element in rank i for the ranking σ. Rather than a direct evaluation of the score s, the empirical version of the ROC is an evaluation of the permutation σ.

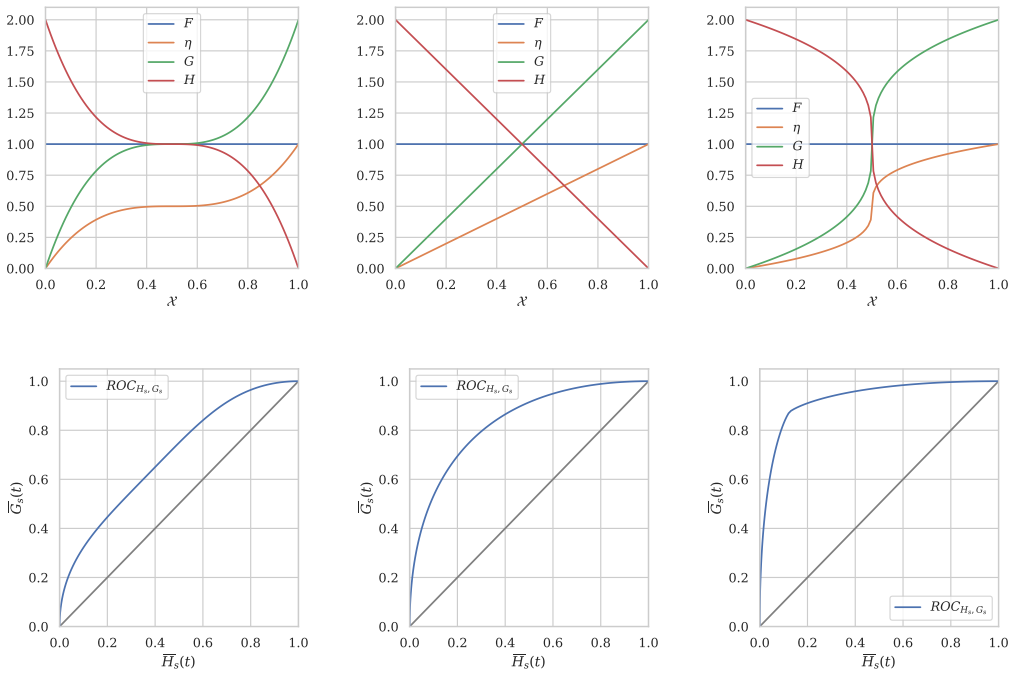
Consider an independent copy (X', Y') of the random pair (X, Y). If the score perfectly separates the positives and the negatives, *i.e.* $s(X) > s(X')|Y = +1, Y' = -1$ *a.s.*, then the ROC curve is included in the plot of a step function, defined as $\{(t_1, t_2) \mid t_1 = 0 \text{ or } t_2 = 1\}$. If the score does not separates the positives from the negatives at all, *i.e.* $s(X)|Y = -1$ and $s(X)|Y = +1$ have the same distribution, then the ROC curve is included in the plot of the identity function, defined as $\{(t, t) \mid t \in [0, 1]\}$.

As defined in Eq. (3.1), the ROC curve is not continuous in general. One approach to solve that limitation is to consider the ROC curve as the evaluation of the score s, where the equality $s(x) = s(x')$ between the score of two elements induces a random order between those two elements. That extension connects two consecutive disconnected dots of the ROC curve with a straight line.

Remark 3.1. *With the (F, η) defined in Example 2.1 from Chapter 2. Introduce the noise coefficient parameter $a \in [0, 1]$. Then, with $s : x \mapsto x$, we have $\bar{H}_s(t) = 2(1-t) - \bar{G}_s(t)$, and:*

$$\bar{G}_s(t) = 1 - t + \frac{a}{2}\left[1 - |1 - 2t|^{\frac{1}{a}}\right].$$

A graphical representation of the ROC *curves is featured in Fig. 3.1.*



(d) $a = 1/4$ (e) $a = 1/2$ (f) $a = 3/4$

Figure 3.1: Illustration of the ROC curve of the score function $s : x \in [0, 1] \mapsto x$ for three distributions with three different levels of separability of the distributions $s(X)|Y = -1$ and $s(X)|Y = +1$. Here, we denote by H (resp. G) the distribution $s(X)|Y = -1$ (resp. $s(X)|Y = +1$). Here, the distributions F, H, G are all absolutely continuous, and are thus represented by their Radon-Nykodim derivatives *w.r.t.* the Lebesgue measure.

Another parametrization of the ROC curve is as the plot of the function:

$$\begin{aligned}\mathrm{ROC}_{H_s,G_s} : [0,1] &\to [0,1],\\ \alpha &\mapsto \bar{G}_s \circ \bar{H}_s^{-1}(\alpha),\end{aligned}$$

where the generalized inverse of a decreasing function f writes $f^{-1} : t \mapsto \sup\{\lambda \mid f(\lambda) > t\}$. Introducing the definition of the generalized inverse of an increasing function f as $f^{-1}(t) = \inf\{\lambda \mid f(\lambda) \geqslant t\}$, we have:

$$\bar{H}_s^{-1}(\alpha) = \sup\{\lambda \mid 1-\alpha > H_s(\lambda)\} = \inf\{\lambda \mid H_s(\lambda) \geqslant 1-\alpha\} = H_s^{-1}(1-\alpha).$$

The quantity $\bar{H}_s^{-1}(\alpha)$ is a $(1-\alpha)$-quantile of the random variable $s(X)|Y=-1$, since a property of the quantile function, specifically Lemma 21.1 of van der Vaart (2000), implies:

$$\begin{aligned}\mathbb{P}\{s(X) \leqslant \bar{H}_s^{-1}(\alpha) \mid Y=-1\} &= \mathbb{P}\{s(X) \leqslant H_s^{-1}(1-\alpha) \mid Y=-1\}\\ &= H_s \circ H_s^{-1}(1-\alpha) \geqslant 1-\alpha.\end{aligned}$$

Hence, we introduce the notation $Q(Z,\alpha)$ to denote the quantile of order $1-\alpha$ of any random variable Z conditioned on the event $Y=-1$, and we have $Q(s(X),\alpha) = \bar{H}_s^{-1}(\alpha)$.

A score function s_1 is considered better than another score function s_2 whenever its ROC is higher, *i.e.* when the ROC of s_1 uniformly dominates that of s_2. Introducing the notation $\mathrm{ROC}(s,\alpha) := \mathrm{ROC}_{H_s,G_s}(\alpha)$ for any score function s and any $\alpha \in [0,1]$, it is verified when $\mathrm{ROC}(s_1,\alpha) \geqslant \mathrm{ROC}(s_2,\alpha)$ for any $\alpha \in [0,1]$.

Proposition 3.3 in Section 3.2.2 below proves the existence of optimal score functions for uniform dominance of ROC's, and shows that η is an optimal score function. Notice that the optimal element for the ranking problem involves the posterior probability η, as in the binary classification setting described in Section 2.1. Introducing the notations $H^* := H_\eta$ and $G^* := G_\eta$, we write the ROC curve of η as $\mathrm{ROC}^* := \mathrm{ROC}_{H_\eta,G_\eta}$, and similarly the $(1-\alpha)$-quantile of $\eta(X)|Y=-1$ as $Q^*(\alpha) := Q(\eta(X),\alpha) = \bar{H}_\eta^{-1}(\alpha)$. The invariance property under strictly increasing transforms of the ROC implies that any score $s \in \mathcal{S}^*$, with $\mathcal{S}^* = \{T \circ \eta \mid T : [0,1] \to \mathbb{R} \text{ is increasing }\}$, satisfies $\mathrm{ROC}_{H_s,G_s} = \mathrm{ROC}^*$.

The order defined by the uniform dominance between two ROC curves is not total, since there exists pairs of score functions that can not be compared. Indeed, the ROC a score function s_1 can strictly dominate that of another score function s_2 for all values α of a subset A of $[0,1]$ and the ROC of s_2 strictly dominate that of s_1 on another subset of A. Additionally, the estimation of the ROC curve from empirical data gives a random function. Results on the estimation of the ROC curve often involve the theory of empirical processes, as do for example the strong convergence and strong approximations theorems for the ROC curve of Hsieh and Turnbull (1996). The work of Bertail et al. (2008) draws on that analysis to derive a bootstrap procedure to derive tight confidence bands for the ROC curve.

Due to these two reasons, *i.e.* the absence of total order on the ROC curve and its functional nature, practitioners often consider summaries of the ROC curve instead of the full functional criterion. The most popular one is the *Area under the* ROC *Curve* (AUC), defined below.

Definition 3.2 (Area under the ROC curve (AUC))**.** *(Clémençon et al., 2008, Proposition B.2). The Area under the* ROC *Curve (*AUC*) of a score function s writes:*

$$\mathrm{AUC}_{H_s,G_s} := \int_0^1 \mathrm{ROC}_{H_s,G_s}(\alpha)\, d\alpha.$$

The AUC *is equal to the proportion of correctly ranked pairs of independent elements. Formally, with (X,Y) and (X',Y') i.i.d. copies of the distribution P, we have:*

$$\mathrm{AUC}_{H_s,G_s} = \mathbb{P}\{s(X) \geqslant s(X') \mid Y=+1, Y'=-1\}.$$

We write $\mathrm{AUC}(s) = \mathrm{AUC}_{H_s,G_s}$ *for any score s, and* $\mathrm{AUC}^* := \mathrm{AUC}_{H_\eta,G_\eta}$.

The AUC encourages viewing the ranking problem as a binary classification problem over $\mathcal{X} \times \mathcal{X}$ on the pairs (X,X') with different label, *i.e.* that satisfies $Y \neq Y'$. In that framework, the

binary objective is $(Y - Y')/2$). The ubiquity of the AUC implies that it is often considered as the standard performance measure for the bipartite ranking problem. Most of the papers about bipartite ranking focus on maximizing that quantity. A theoretical analysis of the AUC optimization procedure is provided in Clémençon et al. (2008). Many algorithm deal with the bipartite ranking problem by optimizing the AUC, such as RankSVM (Joachims, 2002), RankBoost (Freund et al., 2003) and RankNet (Burges et al., 2005).

The estimation of the AUC involves a special type of statistics, different in nature from the standard averages studied in Chapter 2. Specifically, studying the empirical AUC uses estimators called *U-statistics* and presented in Chapter 4, which in their simplest form are averages on all possible pairs of elements from a random sample. Essential results for those statistics are provided in Chapter 4. Formally, introduce a sequence of n *i.i.d.* pairs with the same distribution as (X, Y), *i.e.* $\mathcal{D}_n = \{(X_i, Y_i)\}_{i=1}^n \overset{i.i.d.}{\sim} P$, a natural estimator of the AUC writes:

$$\widehat{\mathrm{AUC}}_{H_s,G_s} := \frac{1}{n_+ n_-} \sum_{Y_i=+1} \sum_{Y_j=-1} \mathbb{I}\{s(X_i) > s(X_j)\}, \tag{3.2}$$

with $n_+ := \sum_{i=1}^n \mathbb{I}\{Y_i = +1\}$ and $n_- := n - n+$.

While the AUC is a simple summary of the ROC curve, a drawback of the AUC in practical problems is that it considers all inversions to be equivalent. For example, an inversion that puts only one negative instance above the next positive instance that is higher on the list, will contribute as much to decreasing the AUC if the positive instance is at the top or if in the middle of the list, see Eq. (3.2). However, for many practical problems, having an accurate ranking is way more important at the beginning of the list than at the bottom. Therefore, a corpus of literature addresses the problem of *ranking the best instances*, which introduces many criteria that consider errors at the top as more critical than errors at the bottom.

Those criteria include popular accuracy measures from the information retrieval community, such as the *Discounted Gain Criterion* (DCG) or the *Mean Reciprocal Rank* (MRR) (Menon and Williamson, 2016, section 9.2), as well as propositions from the statistical machine learning community, such as the *the p-norm push* (Rudin, 2006). In Clémençon and Vayatis (2007), ranking the best is seen through the lens of a combination of two problems: finding the best instances and ordering that subset of instances well.

In this section, we first discuss the task of finding the optimal score function for maximizing the true positive rate under an upper-bound $\alpha \in [0, 1]$ on the false positive rate. We refer to this task as *pointwise* ROC *optimization* pROC at level α. Then, we present theoretical guarantees for pROC, that were proven in Theorem 10 and Theorem 12 of Clémençon and Vayatis (2010). Those guarantees are modeled after those presented in Chapter 2. Finally, we recall a few results on the algorithm TreeRank, a flexible approach for solving the ranking problem by partitioning the input space. We refer to Clémençon and Vayatis (2009) for an extensive analysis of TreeRank.

3.2.2 Pointwise ROC Optimization (pROC)

Given a proposed family of score functions $\mathcal{S}$, we define pointwise ROC optimization at level $\alpha \in [0, 1]$ as the task of finding the score $s \in \mathcal{S}$ and threshold $t \in \mathbb{R}$, such that the test $s(x) > t$ has false positive rate constrained by α, and the highest true positive rate $\bar{G}_s(t)$ as possible. Formally, it writes:

$$\max_{(s,t) \in \mathcal{S} \times \mathbb{R}} \bar{G}_s(t) \quad \text{s.t.} \quad \bar{H}_s(t) \leqslant \alpha. \tag{3.3}$$

A solution of Eq. (3.3) is written (s_α, t_α). It satisfies $\mathrm{ROC}(s_\alpha, \alpha) \geqslant \mathrm{ROC}(s, \alpha)$ for any $s \in \mathcal{S}$ and depends on the parameter α.

While the formulation of Eq. (3.3) involves a scoring function s and threshold t, the quantities $\bar{G}_s(t)$ and $\bar{H}_s(t)$ are an evaluation of the classifier $g : x \mapsto 2 \cdot \mathbb{I}\{s(x) > t\} - 1$. Hence, Eq. (3.3) was first introduced as *Neyman-Pearson classification* due to its relation with the standard hypothesis testing framework, as explained below right before Proposition 3.3. We refer to Scott and Nowak (2005); Scott (2007); Rigollet and Tong (2011) for a detailed account of Neyman-Pearson

classification. Introduce the family $\mathcal{R}$ of super-level sets of $\mathcal{S}$, *i.e.* $\mathcal{R} := \{x \in \mathcal{X} \mid s(x) > t\}_{(s,t)\in\mathcal{S}\times\mathbb{R}}$, then Eq. (3.3) is equivalent to:

$$\max_{R\in\mathcal{R}} G(R) \quad \text{s.t.} \quad H(R) \leqslant \alpha. \tag{3.4}$$

A solution of Eq. (3.4) is $R_\alpha := \{x \in \mathcal{X} | s_\alpha(x) > t_\alpha\}$.

Eq. (3.3) is also very similar to *minimum volume set* (MV-set) estimation, an approach in anomaly detection. MV-set estimation consists in finding a set of minimum volume, as measured by the Lebesgue measure on the Euclidean space $\mathcal{X}$, in which a random variable $X \in \mathcal{X}$ falls with probability greater than $1 - \alpha$. The Lebesgue measure λ is a generalization of the notion of volume to general Euclidean spaces, defined formally in Shorack (2000) (Chapter 5, Example 1.1). Formally, with F the distribution of the *r.v.* X and for the proposed family of sets $\mathcal{A} \subset \mathcal{P}(\mathcal{X})$, the minimum volume set problem writes:

$$\min_{A\in\mathcal{A}} \lambda(A) \qquad \text{such that} \qquad F(A) \geqslant 1 - \alpha. \tag{3.5}$$

The main difference between Eq. (3.3) and Eq. (3.5) is that the objective $\lambda(A)$ of Eq. (3.5) does not have to be replaced by an empirical quantity. Indeed, the Lesbegue measure $\lambda(A)$ is known, unlike the constraint $F(A)$ or the quantities $G(R)$ and $H(R)$ in Eq. (3.4), which all have to be estimated from data. We refer to Polonik (1995, 1997) and Scott and Nowak (2006) for more details on the minimum volume set problem. The similarities between Eq. (3.4) and Eq. (3.5) have led authors to consider the *mass volume curve*, the counterpart of the ROC curve in the context of anomaly detection. We refer to Clémençon and Jakubowicz (2013) and Clémençon and Thomas (2018) for more details on the mass volume curve.

The problem presented in Eq. (3.3) can be considered in the standard hypothesis testing framework, presented in Wasserman (2010) (Section 11). In that light, consider the hypothesis $\mathcal{H}_0 : Y = -1$ against the alternative $\mathcal{H}_1 : Y = +1$. The true positive rate $\bar{G}_s(t)$ then corresponds to the power of the test $s(X) \leqslant t$, while the false positive rate $\bar{H}_s(t)$ corresponds to its type I error. Solving Eq. (3.3) for the proposed class of all measurable functions corresponds to finding the *most powerful test* for rejecting $\mathcal{H}_0$ at the *level of significance* α, *i.e.* the test with the highest power and a type I error below or equal to α. An application of the fundamental lemma of Neyman-Pearson (Lehmann and Romano, 2005, Theorem 3.2.1) implies the expression of an optimal solution of Eq. (3.3) in the space of all measurable functions. We present that solution in the proposition below, along an expression of a notion of *pointwise excess risk* at level α for the ROC curve.

Proposition 3.3 (Optimal score function)**.**
(Clémençon and Vayatis, 2009, Proposition 6) (Clémençon and Vayatis, 2010, Proposition 5). For any measurable scoring function s, we have:

$$\forall \alpha \in (0, 1), \qquad \mathrm{ROC}^*(\alpha) \geqslant \mathrm{ROC}(s, \alpha). \tag{3.6}$$

Introduce the notations:

$$R^*_\alpha := \{x \in \mathcal{X} \mid \eta(x) > Q^*(\alpha)\},$$
$$R_{s,\alpha} := \{x \in \mathcal{X} \mid s(x) > Q(s(X), \alpha)\}.$$

Consider α s.t. $Q^*(\alpha) < 1$, and $s \in \mathcal{S}$ such that the c.d.f. H_s (resp. H_η) is continuous at $Q(s(X), \alpha)$ (resp. at $Q^*(\alpha)$). Then we have:

$$\mathrm{ROC}^*(\alpha) - \mathrm{ROC}(s, \alpha) = \frac{\mathbb{E}\left[|\eta(X) - Q^*(\alpha)| \cdot \mathbb{I}\{R^*_\alpha \Delta R_{s,\alpha}\}\right]}{p(1 - Q^*(\alpha))}. \tag{3.7}$$

The proof of Eq. (3.6) can be found in Clémençon and Vayatis (2009) (Proposition 6) and is a consequence of the Neyman-Pearson fundamental lemma, proven in Lehmann and Romano (2005) (Section 3.2). The hypothesis of continuity serves to exclude random tests, introduced in Lehmann and Romano (2005) (Section 3.2). One can extend those results by considering a test of the hypothesis that rejects $\mathcal{H}_0$ with some probability.

Consider the decomposition of a score function s as an integral of its super-level sets. Formally:

$$s(x) = \int_0^1 \mathbb{I}\{s(x) > t\}\, dt. \tag{3.8}$$

Clémençon and Vayatis (2010) proposed to find a good solution of the ranking problem by combining the solutions of several pointwise ROC optimization problems, as those are super-level sets of the optimal score function η. Eq. (3.8) suggests a simple combination strategy.

The optimal solution for binary classification, *i.e.* the Bayes classifier $g^* : x \mapsto 2 \cdot \mathbb{I}\{\eta(x) > 1/2\} - 1$, consists in thresholding the Bayes score η by a threshold $t = 1/2$ to separate positive and negative instances. Similarly, the proof of Proposition 3.3 implies that the optimal solution of the pointwise ROC optimization problem at level α Eq. (3.3) also consists in thresholding the Bayes score function, as the optimal rejection region of the corresponding test is R^*_α. In that case, the value of the threshold is unknown, as $t_\alpha = Q^*(\alpha)$ depends on the distribution of the random variable $\eta(X) \mid Y = -1$.

Introduce the weighted classification cost for binary classification, as:

$$R_c(g) := c \cdot \mathbb{P}\{g(X) = -1, Y = +1\} + (1 - c) \cdot \mathbb{P}\{g(X) = +1, Y = -1\}. \tag{3.9}$$

Eq. (3.9) can be seen as a weighted sum of the false negative rate $G_s(t)$ and false positive rate $\bar{H}_s(t)$ for the classifier defined by $g : x \mapsto s(x) > t$. The proposition below shows that $g^*_c : x \mapsto 2 \cdot \mathbb{I}\{\eta(x) > 1 - c\} - 1$ is a minimizer of $R_c(g)$ in the class of all measurable functions. It is proven in a similar way as the optimality of the Bayes classifier for binary classification, presented in Chapter 2. Its proof is thus omitted.

Proposition 3.4 (Optimal classifier for weighted classification)**.**
For any $c \in [0, 1]$, we have that $R_c(g) \geqslant R_c(g^*_c)$ for any measurable classification function $g : X \to \{-1, +1\}$.

The optimal solution of weighted classification also writes as a thresholding operation on the Bayes score, with a threshold $t_c := 1 - c$. The optimal solution of pointwise ROC optimization Eq. (3.3) at level α is then the same as the optimal solution of Eq. (3.9) for an unknown cost asymmetry parameter $c = 1 - Q^*(\alpha)$. Hence, a common strategy for optimal ranking is to combine many solutions of the problem of classification with asymmetric costs, as done for example in Bach et al. (2006).

3.2.3 Generalization Guarantees for pROC

In this section, we derive guarantees for empirical solutions of the pointwise ROC optimization problem. We first derive an uniform learning bound, followed by a fast learning bound under a noise assumption on the data distribution. The uniform learning bound is found in Scott and Nowak (2005) (Proposition 1) for Neyman-Pearson classification and is a direct consequence of Theorem 10 of Clémençon and Vayatis (2010), while the fast learning bound is Theorem 12 in Clémençon and Vayatis (2010).

Using the same *i.i.d.* sample of n observations $\mathcal{D}_n$ as for the estimator $\widehat{\mathrm{AUC}}_{H_s,G_s}$ in Eq. (3.2), we introduce an empirical approximation of the empirical distribution of the negatives and positives as $\widehat{H}$ and $\widehat{G}$, respectively. Formally,

$$\widehat{H} := \frac{1}{n_-} \sum_{Y_i=-1} \delta_{X_i} \quad \text{and} \quad \widehat{G} := \frac{1}{n_+} \sum_{Y_i=+1} \delta_{X_i},$$

where δ_x is the Dirac distribution in x, *i.e.* $\delta_x(A) = 1$ if $x \in A$ and $\delta_x(A) = 0$ otherwise.

To define an empirical ROC curve, we first introduce an estimator $\widehat{H}_s$ of the distribution H_s of $s(X)|Y = -1$ and an estimator $\widehat{G}_s$ of the distribution G_s of $s(X)|Y = +1$, such that:

$$\widehat{H}_s(t) := \widehat{H}(s(X) \leqslant t) = \frac{1}{n_-} \sum_{Y_i=-1} \mathbb{I}\{s(X_i) \leqslant t\},$$

$$\widehat{G}_s(t) := \widehat{G}(s(X) \leqslant t) = \frac{1}{n_+} \sum_{Y_i=+1} \mathbb{I}\{s(X_i) \leqslant t\}.$$

Then, we can define an estimator of the ROC curve as $\widehat{\mathrm{ROC}}_{H_s,G_s}$, where:

$$\begin{aligned}\widehat{\mathrm{ROC}}_{H_s,G_s} : [0,1] &\to [0,1],\\ \alpha &\mapsto 1 - \widehat{H}_s \circ \widehat{G}_s(1-\alpha).\end{aligned}$$

Additionally, we introduce an empirical version of the pointwise ROC optimization problem introduced in Eq. (3.4), defined as:

$$\max_{R \in \mathcal{R}} \widehat{G}(R) \quad \text{s.t.} \quad \widehat{H}(R) \leqslant \alpha + \phi(n,\delta), \tag{3.10}$$

and a solution to Eq. (3.10) is written as $\widehat{R}_\alpha$. Note the presence of a tolerance term $\phi(n,\delta) \geqslant 0$ in the constraint, whose function is to tolerate the variations of $\widehat{H}(R)$ around $H(R)$. Precisely, the decomposition $\widehat{H}(R) = H(R) + (\widehat{H}(R) - H(R))$ implies that very often, very good solutions of Eq. (3.4) will not satisfy the constraint of Eq. (3.10) if $\phi(n,\delta) = 0$, depending on the generated sample $\mathcal{D}_n$. Rigollet and Tong (2011) take a conservative approach and investigate Eq. (3.10) with a negative tolerance term $\phi(n,\delta)$, so that it enforces an empirical solution $\widehat{R}_\alpha$ that satisfies the constraint $H(\widehat{R}_\alpha) \leqslant \alpha$.

Under a complexity assumption on the proposed family $\mathcal{S}$, Clémençon and Vayatis (2010) prove learning bounds for the pointwise ROC optimization problem, as presented in Proposition 3.5 below. In Proposition 3.5, the complexity assumption is that $\mathcal{S}$ is a VC-major class of functions, which means that the super-level sets $\mathcal{R}$ of the score functions in $\mathcal{S}$ form a VC-class of sets, see Definition 2.14 and Definition 2.30 in Chapter 2.

Proposition 3.5. *(Consequence of Theorem 10 in Clémençon and Vayatis (2010)). See Proposition 1 of Scott and Nowak (2005) for the same result in Neyman-Pearson classification.) Let $\alpha \in (0,1)$. Assume that:*

- *$\mathcal{R}$ is a VC-class of functions with VC-dimension V,*
- *$\epsilon = \min(p, 1-p) > 0$,*
- *H_η is continuous at $Q^*(\alpha)$ and $R^*_\alpha \in \mathcal{R}$.*

For all $(\delta, n) \in (0,1) \times \mathbb{N}^*$, set:

$$\phi(n,\delta) = \frac{1}{\epsilon}\left(\sqrt{\frac{2\log(2/\delta)}{n}} + \sqrt{\frac{8\log(2) + 8V\log(1+2n)}{n}}\right).$$

Then, for all $(\delta, n) \in (0,1) \times \mathbb{N}^*$, we simultaneously have w.p. $\geqslant 1-\delta$:

$$H(\widehat{R}_\alpha) \leqslant \alpha + 2\phi(n,\delta/2) \quad \textit{and} \quad G(\widehat{R}_\alpha) \geqslant G(R^*_\alpha) - 2\phi(n,\delta/2). \tag{3.11}$$

The proof of Proposition 3.5 is slightly different than that of Clémençon and Vayatis (2010) (Theorem 10), as the quantities $\widehat{H}$ and $\widehat{G}$ satisfy:

$$\text{for any } R \in \mathcal{R}, \quad \widehat{H}(R) = \frac{\sum_{i=1}^n \mathbb{I}\{Y_i = -1, X_i \in R\}}{\sum_{i=1}^n \mathbb{I}\{Y_i = -1\}} \quad \text{and} \quad \widehat{G}(R) = \frac{\sum_{i=1}^n \mathbb{I}\{Y_i = +1, X_i \in R\}}{\sum_{i=1}^n \mathbb{I}\{Y_i = -1\}},$$

thus are ratios of averages, while Clémençon and Vayatis (2010) (Theorem 10) assumed n_- and n_+ to be deterministic. However, the core argument remains the same. It relies on the fact that PAC-style bounds for the supremum of empirical processes $\widehat{G}(R) - G(R)$ and $\widehat{H}(R) - H(R)$ over $\mathcal{R}$ imply the bilateral control of errors presented in Eq. (3.11).

Proposition 3.5 recovers the standard learning bound in $O(n^{-1/2})$ presented in Section 2.3 for binary classification, for both the constraint and the objective, Using the same tools as in Section 2.4, Clémençon and Vayatis (2010) (Section 5.2) prove faster speeds of convergence for the pointwise ROC optimization problem under a noise assumption on the data given below. The noise assumption is very similar to the Mammen-Tsybakov assumption for binary classification, but concerns the distribution of $\eta(x)$ around $Q^*(\alpha)$ instead of $1/2$, see Assumption 2.24.

Assumption 3.6. *(Clémençon and Vayatis, 2010, noise assumption)*
Let $\alpha \in (0,1)$. There exists $B > 0$ and $a \in [0,1]$, s.t.:

$$\mathbb{P}\{|\eta(X) - Q^*(\alpha)| \leqslant t\} \leqslant B \cdot t^{\frac{a}{1-a}}.$$

Clémençon and Vayatis (2010) (Remark 14) explains that Assumption 3.6 is satisfied with $a = 1/2$ and $B = 2 \cdot \sup_t f^*(t)$ as soon as $\eta(X)$ has a bounded density f^*. Indeed, with F^* the *c.d.f.* of $\eta(X)$, the finite increments theorem gives:

$$\mathbb{P}\{|\eta(X) - Q^*(\alpha)| \leqslant t\} = F^*(Q^*(\alpha) + t) - F^*(Q^*(\alpha) - t) \leqslant Bt.$$

As in Section 2.4.2, the noise assumption (Assumption 3.6) implies a control on the variance of a notion of excess loss, presented in the lemma below. The implication is proven in Clémençon and Vayatis (2010), in a similar manner as Proposition 2.26 in Chapter 2, using equivalent formulations of the Mammen-Tsybakov assumption that can be found in Bousquet et al. (2003) (Definition 7).

Lemma 3.7. *(Clémençon and Vayatis, 2010, Lemma 11)*
Suppose that H_η is continuous at $Q^*(\alpha)$ and that Assumption 3.6 is fulfilled. Set for all $R \in \mathcal{R}$,

$$s_\alpha^2(R) := Var\left[\mathbb{I}\{Y = +1\}\left(\mathbb{I}\{X \in R_\alpha^*\} - \mathbb{I}\{X \in R\}\right)\right].$$

Then we have:

$$\forall R \in \mathcal{R}, \quad s_\alpha^2(R) \leqslant c\left[p(1 - Q^*(\alpha))(G(R_\alpha^*) - G(R)) + Q^*(\alpha)(1-p)(H(R) - \alpha)\right]^a.$$

Similarly to Section 2.4.3, the control on the variance can be used to derive fast bounds on a finite family of functions with Bernstein's inequality, as proven in Clémençon and Vayatis (2010).

Proposition 3.8. *(Clémençon and Vayatis, 2010, Theorem 12)*
Assume that the assumptions of Proposition 3.5 are satisfied and that Assumption 3.6 is fulfilled for some $\alpha \in (0,1)$ with noise parameter $a \in [0,1]$. Then, for all $\delta > 0$, there exists $C := C(\mathcal{R}, \delta, \alpha, p, Q^*(\alpha))$ and $n_0 := n_0(\mathcal{R}, \delta, \alpha, p, Q^*(\alpha))$, such that $\forall n \geqslant n_0$, we simultaneously have $w.p \geqslant 1 - \delta$:

$$H(\widehat{R}_\alpha) \leqslant \alpha + 2\phi(n, \delta/2) \quad \textit{and} \quad \mathrm{ROC}^*(\alpha) - G(\widehat{R}_\alpha) \leqslant Cn^{-\frac{2+a}{4}}.$$

Proposition 3.8 above proves bounds of the order $O(n^{-(2+a)/4})$ for the true positive rate under bounds in $O(n^{-1/2})$ for the false positive rate. It is slower than the usual fast learning rate proven in Section 2.4, due to the bilateral control of the two types of errors.

3.2.4 TreeRank

TreeRank is a recursive algorithm that gives a solution of the scoring problem based on the adaptive approximation of the ROC curve of the optimal score function. Like other decision tree-based algorithms, it recursively partitions the input space into cells. However, the output of TreeRank is a piecewise constant scoring function, and the inverse image of its values corresponds to leaves of the associated tree. In its simplest form, the algorithm generates a full binary tree of depth D, which gives a partition of the space $\mathcal{X}$ in 2^D cells $C_{D,k}$, indexed by $k \in \{0, \ldots, 2^D - 1\}$. TreeRank is a *greedy algorithm*, since the splitting procedure is designed to optimize the AUC at each step.

The algorithm is described in Clémençon and Vayatis (2009) and Clémençon et al. (2011). We refer the reader to Clémençon et al. (2011) (Section 4) for approaches to prune the full ranking tree, and to Clémençon et al. (2013) for a description of an *ensemble method* for aggregating TreeRank-based score functions. The aggregate is named *Ranking Forest*, as a reference of the notion of *Random Forest* introduced in Breiman (2001). The aggregation of the random trees is based on the idea of aggregating the order induced by each of the trees, see Section 3.3 below for a short introduction to ranking aggregation, but in practice often consists of simply averaging the scores. In this section, we present theoretical results for the TreeRank procedure that can

be found in Clémençon and Vayatis (2009), as the thesis extends those results to a variant of TreeRank adapted to the problem of similarity learning.

In the case of a full binary tree $\mathcal{T}$ of depth D, the score function writes as:

$$s_{\mathcal{T}}(x) := \sum_{k=0}^{2^D-1} (2^D - k) \cdot \mathbb{I}\{x \in C_{D,k}\},$$

and the score $s_{\mathcal{T}}$ induces a total order on the cells $\{C_{D,k}\}_{k=0}^{2^D-1}$. At the beginning of the formation of the ranking tree, a cell $C_{0,0}$ contains all of the input space, hence all instances of $\mathcal{X}$ have the same score. To form a tree of depth $D = 1$, $C_{0,0}$ is split in two cells $C_{1,0}$ and $C_{1,1}$. In general, we introduce the notation corresponding to the cell $C_{d,k}$ of each node, such that:

$$\forall d, k \in \{1, \dots, D\} \times \{0, \dots, 2^d - 1\} \text{ with } k = 2m,\ m \in \mathbb{N}, \qquad C_{d,k} \cup C_{d,k+1} = C_{d-1,m}.$$

The parametrization of the nodes by the tuple (d, k) is best understood by considering the problem of visiting all the nodes of a tree, see Sedgewick and Wayne (2011) (Chapter 29). Going through the nodes (d, k) by increasing d and then by increasing k implements a depth first search (DFS). On the other hand, going through the nodes (d, k) by increasing k, then by increasing d implements a breadth first search (BFS). For example, for a tree of depth 2, it gives:

$$\begin{aligned} \text{(DFS)}: &\quad (0,0) \prec (1,0) \prec (2,0) \prec (2,1) \prec (1,1) \prec \dots \\ \text{(BFS)}: &\quad (0,0) \prec (1,0) \prec (1,1) \prec (2,0) \prec (2,1) \prec \dots \end{aligned}$$

where $(d, k) \prec (d', k')$ means that the node (d, k) is visited first.

As shown by the definition of the ROC curve, see Eq. (3.1), the ROC curve of a piecewise constant score function is a set of points on the ROC plane. A common response in the analysis of piecewise constant scores, is to consider its ROC to be the broken line that connects those points, which corresponds to considering an element to be above another with probability 1/2 when their scores are equal, as precised below Eq. (3.1).

We introduce the alternative notations for the proportion of positives (resp. proportion of negatives) contained by a set C as $\beta(C)$ (resp. $\alpha(C)$), *i.e.* for any $C \subset \mathcal{X}$, $\beta(C) := G(C)$ and $\alpha(C) := H(C)$. Then, Clémençon and Vayatis (2009) (Remark 13) provides an expression for the AUC of the piecewise constant score function s_τ as:

$$\mathrm{AUC}(s_\tau) = \frac{1}{2} \sum_{k=1}^{2^D-1} \left(\alpha(C_{D,k}) + \alpha(C_{D,k-1})\right) \beta(R_{D,k-1}), \tag{3.12}$$

where $R_{d,j} = \cup_{k=0}^{j} C_{d,k}$ for all $d \geqslant 0$, $j \in \{0, \dots, 2^d - 1\}$.

The ROC of the output s_D of TreeRank is thus a piecewise linear function, whose ROC ($\mathrm{ROC}(s_D, \cdot)$) is designed to approach that of a piecewise linear approximation of the optimal ROC curve ROC*. Clémençon and Vayatis (2009) introduce a score function s_D^*, whose ROC ($\mathrm{ROC}(s_D^*, \cdot)$) is the piecewise linear approximation of ROC*. Hence, deriving theoretical guarantees for TreeRank consists in two steps. First, under regularity assumptions on the optimal ROC curve, we quantify how the piecewise linear interpolation $\alpha \mapsto \mathrm{ROC}(s_D^*, \alpha)$ approximates the optimal ROC curve ROC*. Second, we show that the ROC of s_D can recover a good approximation of the interpolation $\alpha \mapsto \mathrm{ROC}(s_D^*, \alpha)$ learned from data, under an assumption on the complexity of the proposed family $\mathcal{C}$ used to split the cells. The algorithm for splitting the cells is referred to as LeafRank and its nature dictates the complexity of $\mathcal{C}$. The distance between the ROC curves of s_D, s_D^* and $s^* \in \mathcal{S}^*$ is measured with the following quantities, for any two score functions s_1, s_2:

$$\begin{aligned} d_1(s_1, s_2) &= \int_0^1 |\mathrm{ROC}(s_1, \alpha) - \mathrm{ROC}(s_2, \alpha)|\, d\alpha, \\ d_\infty(s_1, s_2) &= \sup_{\alpha \in (0,1)} |\mathrm{ROC}(s_1, \alpha) - \mathrm{ROC}(s_2, \alpha)|. \end{aligned}$$

Proposition 3.3 implies that for any score function s, we have $d_1(s^*, s) = \mathrm{AUC}(s^*) - \mathrm{AUC}(s)$.

In general, the approximation error of the piecewise linear interpolation of a function f that is twice differentiable on $[a, b]$ is controlled using its second derivative, as shown in Davis (1975) (Theorem 3.1.1) and reminded in the proposition below. This property is a consequence of the generalized Rolle's theorem (Davis, 1975, Theorem 1.6.3).

Proposition 3.9. *Assume that $f : \mathbb{R} \to \mathbb{R}$ is twice differentiable on $[a, b]$. Formally, for any x such that $a \leqslant x \leqslant b$, we have:*

$$\left| f(x) - \left(f(a) + \frac{f(b) - f(a)}{b - a}(x - a) \right) \right| \leqslant \frac{(b - a)^2}{8} \max_{a<u<b} \left| f''(u) \right|.$$

By the definition of the optimal ROC curve, points of ROC^* correspond to error rates of super-level sets of the posterior probability η. Therefore, the candidate splitting class for the interpolation procedure writes as a super level-set on the value $\eta(x)$:

$$\mathcal{C}^* = \{\{x \in \mathcal{X} : \eta(x) > t\} : t \in [0, 1]\},$$

as expected from Proposition 3.3. The interpolation procedure seeks to find a set $C_t \in \mathcal{C}^*$ that satisfies $C_t = \{x \in \mathcal{X} : \eta(x) > t\}$, with $t \in [0, 1]$, such that adding the point $(\alpha(C_t), \beta(C_t))$ between the two points $(\alpha^*_{d,k}, \beta^*_{d,k})$ and $(\alpha^*_{d,k+1}, \beta^*_{d,k+1})$ of ROC^* maximizes the AUC increment $\Lambda^*_{d,k}(C_t)/2$, with:

$$\begin{aligned} \Lambda^*_{d,k}(C_t) &:= (\alpha^*_{d,k+1} - \alpha^*_{d,k}) \cdot \beta(C_t) - (\beta^*_{d,k+1} - \beta^*_{d,k}) \cdot \alpha(C_t), \\ &= (\alpha^*_{d,k+1} - \alpha^*_{d,k}) \cdot \bar{G}^*(t) - (\beta^*_{d,k+1} - \beta^*_{d,k}) \cdot \bar{H}^*(t). \end{aligned} \tag{3.13}$$

The quantity $\Lambda^*_{d,k}(C_t)$ is the AUC increment of the split, since the magnitude of the cross-product between the vectors $(\alpha(C_t), \beta(C_t), 0)$ and $(\alpha^*_{0,1} - \alpha^*_{0,0}, \beta^*_{0,1} - \beta^*_{0,0}, 0)$ is equal to the area of the parallelogram that the vectors span. Maximizing $\Lambda^*_{d,k}(C_t)$ in t yields the following equality, if both $\bar{H}^*$ and $\bar{G}^*$ are differentiable and the derivative of $\bar{H}^*$ is nonzero:

$$\frac{\bar{G}^{*'}(t)}{\bar{H}^{*'}(t)} = \frac{\beta^*_{d,k+1} - \beta^*_{d,k}}{\alpha^*_{d,k+1} - \alpha^*_{d,k}}, \tag{3.14}$$

Eq. (3.14) is an equality between the derivative of ROC^* and the slope between the points $(\alpha^*_{d,k}, \beta^*_{d,k})$ and $(\alpha^*_{d,k+1}, \beta^*_{d,k+1})$ of ROC^*.

Both Eq. (3.14) and the approximation result presented in Proposition 3.9 justify the derivation of an expression for the first and second derivative of the ROC curve, provided in Proposition 3.11. Proposition 3.11 requires the following technical assumptions.

Assumption 3.10. *(Clémençon and Vayatis, 2009, Assumptions A1 and A2)*
The three following assumptions hold true:

1. *The distributions G and H are* equivalent*, i.e. each of them is absolutely continuous w.r.t. the other (Shorack, 2000, Definition 1.3, Chapter 4).*
2. *The likelihood ratio $dG/dH(X)$ is bounded. This property is equivalent from Bayes' theorem to $\eta(X) < 1$ a.s..*
3. *The distribution of $\eta(X)$ is absolutely continuous w.r.t. the Lebesgue measure.*

Proposition 3.11 (Derivatives of ROC^*)**.** *(Clémençon and Vayatis, 2009, Proposition 6)*
Assume that Assumption 3.10 is satisfied, and that there exists $c > 0$ such that $H^{*'}(u) \geqslant c$ for any $u \in supp(H^{*'})$, where $supp(H^{*'})$ is the support of $H^{*'}$. Then, the optimal ROC *curve* ROC^* *is twice differentiable on $[0, 1]$ with bounded derivatives, and we have: $\forall \alpha \in [0, 1]$,*

$$\begin{aligned} \frac{d}{d\alpha}\mathrm{ROC}^*(\alpha) &= \frac{1 - p}{p} \cdot \frac{Q^*(\alpha)}{1 - Q^*(\alpha)}, \\ \frac{d^2}{d\alpha^2}\mathrm{ROC}^*(\alpha) &= \frac{1 - p}{p} \cdot \frac{Q^{*'}(\alpha)}{(1 - Q^*(\alpha))^2}, \end{aligned}$$

where $Q^{*'}(\alpha) = -1/H^{*'}(Q^*(\alpha))$. Note that $Q^*(\alpha) \leqslant Q^*(0) = \|\eta(X)\|_\infty < 1$, which implies the boundedness of the derivatives.

The procedure to define s_D^* is summarized in Fig. 3.3. The expression of $\Delta_{d+1,2k}^*$ in Fig. 3.3 is justified by combining Eq. (3.13) with the following expression of the likelihood ratio of $\eta(X) \mid Y = -1$ and $\eta(X) \mid Y = +1$:

$$\frac{dG^*}{dH^*}(u) = \frac{1-p}{p} \cdot \frac{u}{1-u},$$

found in the proof of Proposition 3.11, and maximizing $\Lambda_{d,k}^*(C_t)$ in t. The definition of s_d^* gives, combined with Proposition 3.9 and Lemma 18 of Clémençon and Vayatis (2009), a control for both of the distances d_1 and d_∞ between the ROC of the piecewise constant s_d^* and the ROC of an optimal score function, as shown in Proposition 3.12 below.

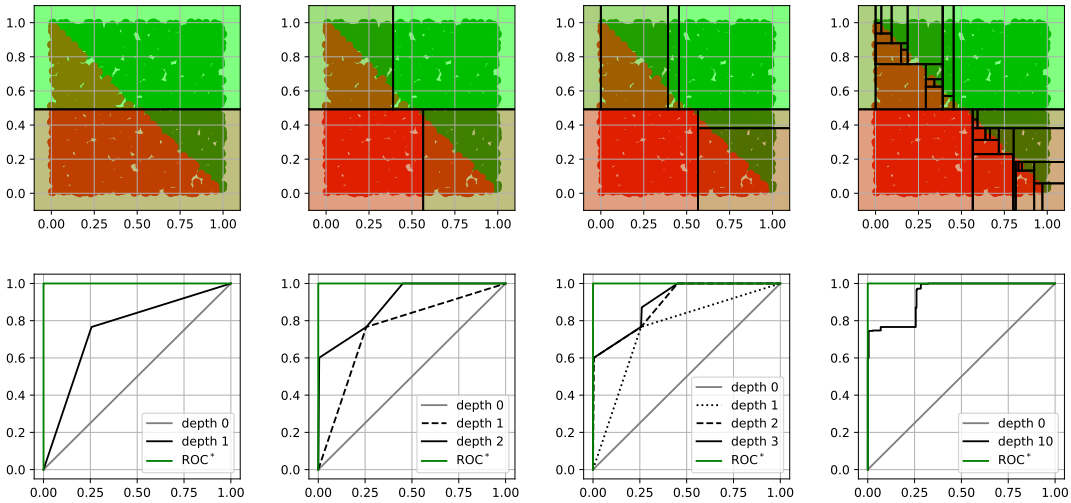


Figure 3.2: Iterations of the LeafRank algorithm for depths 1, 2, 3 and 10. The top figures represent the split of the space, where greener areas correspond to higher scores. The bottom figures correspond to the *ROC* curves associated with the score functions.

Proposition 3.12. *(Clémençon and Vayatis, 2009, Proposition 10)*
Assume that the same assumptions as Proposition 3.11 hold. Then, the sequence of piecewise constant scoring functions $(s_d^*)_{d\geqslant 1}$, satisfies: for any $d \geqslant 1$,

$$\begin{aligned} \mathrm{AUC}^* - \mathrm{AUC}(s_d) = d_1(s^*, s_d^*) &\leqslant C \cdot 2^{-2d}, \\ d_\infty(s^*, s_d^*) &\leqslant C \cdot 2^{-2d}. \end{aligned}$$

Now, we present guarantees for the distance between the ROC's of the output of the TreeRank algorithm ($\mathrm{ROC}(s_D, \cdot)$) and the piecewise linear approximation of ROC* ($\mathrm{ROC}(s_D^*, \dot{})$). As TreeRank's scoring rule is learned from empirical data, we introduce the empirical proportion of negatives (resp. positives) as $\widehat{\alpha}$ (resp. $\widehat{\beta}$) on the set C as:

$$\widehat{\alpha}(C) := \frac{1}{n_-} \sum_{i=1}^{n} \mathbb{I}\{X_i \in C, Y_i = -1\} \qquad \text{and} \qquad \widehat{\beta}(C) := \frac{1}{n_+} \sum_{i=1}^{n} \mathbb{I}\{X_i \in C, Y_i = +1\}.$$

TreeRank relies on a proposed family of sets $\mathcal{C}$, that has finite VC-dimension. The TreeRank procedure is described in Fig. 3.4, and follows the same ideas as Fig. 3.3 but without knowledge of the data distributions and optimal ROC curve ROC*. See Fig. 3.2 for a visual representation of the construction of a ranking tree. The following lemma defines a recurrence hypothesis that implies Theorem 3.15. Combined with Proposition 3.12, Theorem 3.15 implies guarantees for the TreeRank algorithm.

Example 3.13. *Consider that the random variable X is uniformly distributed on $\mathcal{X} = [0,1]^2$ and that $\eta(x) := \mathbb{I}\{x_1 + x_2 > 1\}$. Then, the data is separable and the optimal* ROC *(ROC*) is the unit step. In Fig. 3.2, we show the* ROC *curve obtained by* TreeRank *with coordinate splits*

INTERPOLATION OF ROC*

(Clémençon and Vayatis, 2009, Section III-D)

Initialization ($d = 0$ and $d = 1$). For the extremal points, we set

$$\forall d \in \mathbb{N}, \qquad \alpha^*_{d,0} = \beta^*_{d,0} = \alpha(\varnothing) = 0 \quad \text{and} \quad \alpha^*_{d,2^d} = \beta^*_{d,2^d} = \alpha(\mathcal{C}_{0,0}) = 1.$$

From d to $d+1$, for $d \geqslant 1$.
We are given the collection of points $\{(\alpha^*_{d,k}, \beta^*_{d,k})\}_{k=0,\dots,2^d-1}$. On each interval $(\alpha^*_{d,k}, \alpha^*_{d,k+1})$, we find a point on ROC* that maximizes the incremental AUC quantity $\Lambda^*_{d,k}$, given by:

$$\alpha^*_{d+1,2k+1} = \bar{H}^*(\Delta^*_{d+1,2k+1}), \quad \text{and} \quad \beta^*_{d+1,2k+1} = \bar{G}^*(\Delta^*_{d+1,2k+1}),$$

where:

$$\Delta^*_{d+1,2k+1} = \frac{p(\beta^*_{d,k+1} - \beta^*_{d,k})}{(1-p)\left(\alpha^*_{d,k+1} - \alpha^*_{d,k}\right) + p\left(\beta^*_{d,k+1} - \beta^*_{d,k}\right)}.$$

Moreover, the previous cut-off point is renamed:

$$\alpha^*_{d+1,2k} = \alpha^*_{d,k} \quad \text{and} \quad \beta^*_{d+1,2k} = \beta^*_{d,k} \quad \text{and also} \quad \Delta^*_{d+1,2k} = \Delta^*_{d,k}.$$

Output.
For each level d, the resulting partition is given by the class of sets:

$$C^*_{d,k} = \{x \in \mathcal{X} : \Delta^*_{d,k+1} < \eta(x) \leqslant \Delta^*_{d,k}\},$$

for all $k = 0, \dots, 2^d - 1$ with the convention that $\Delta^*_{d,0} = 0$ and $\Delta_{d,2^d} = 1$ for all $d \geqslant 0$.

Figure 3.3: Description of the ROC* interpolation procedure.

THE TREERANK ALGORITHM

Input. Maximal depth $D \geqslant 1$ of the scoring tree, class $\mathcal{A}$ of measurable subsets of $\mathcal{X}$, training dataset $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$.

1. (INITIALIZATION.) Set $\mathcal{C}_{0,0} = \mathcal{X}$, $\alpha_{d,0} = \beta_{d,0} = 0$ and $\alpha_{d,2^d} = \beta_{d,2^d} = 1$ for all $d \geqslant 0$.
2. (ITERATIONS.) For $d = 0, \ldots, D-1$ and $k = 0, \ldots, 2^d - 1$:
   (a) (OPTIMIZATION STEP.) Set the entropic measure:
   $$\widehat{\Lambda}_{d,k+1}(\mathcal{C}) = (\alpha_{d,k+1} - \alpha_{d,k})\widehat{\alpha}(\mathcal{C}) - (\beta_{d,k+1} - \beta_{d,k})\widehat{\beta}(\mathcal{C}).$$
   Find the best subset $\mathcal{C}_{d+1,2k}$ of the cell $\mathcal{C}_{d,k}$ in the AUC sense:
   $$\mathcal{C}_{d+1,2k} = \operatorname*{arg\,max}_{\mathcal{C} \in \mathcal{A},\ \mathcal{C} \subset \mathcal{C}_{d,k}} \widehat{\Lambda}_{d,k+1}(\mathcal{C}) \ . \tag{3.15}$$
   Then, set $\mathcal{C}_{d+1,2k+1} = \mathcal{C}_{d,k} \backslash \mathcal{C}_{d+1,2k}$.
   (b) (UPDATE.) Set
   $$\alpha_{d+1,2k+1} = \alpha_{d,k} + \widehat{\alpha}(\mathcal{C}_{d+1,2k}), \quad \alpha_{d+1,2k+2} = \alpha_{d,k+1},$$
   $$\beta_{d+1,2k+1} = \beta_{d,k} + \widehat{\beta}(\mathcal{C}_{d+1,2k}), \quad \beta_{d+1,2k+2} = \beta_{d,k+1} \ .$$
3. (OUTPUT.) After D iterations, get the piecewise constant score function:
   $$s_D(x) = \sum_{k=0}^{2^D - 1} (2^D - k)\ \mathbb{I}\{(x) \in \mathcal{C}_{D,k}\}. \tag{3.16}$$

Figure 3.4: Description of the TREERANK procedure.

as LEAFRANK *for differents values of the depth D. As* TREERANK *is a greedy algorithm, it can not recover from an initial bad split in that case. Our illustrations shows the necessity for the* LEAFRANK *algorithm is adapted to type of data considered.*

Lemma 3.14. *(Clémençon and Vayatis, 2009, Lemma 19)*
Assume that the assumptions of Proposition 3.12 hold true and that the class $\mathcal{C}$ of subset candidates contains all level sets $\{R^*_\alpha\}_{\alpha \in [0,1]}$ and is of VC-dimension V and is intersection stable, i.e., $\forall C, C' \in \mathcal{C}^2 : C \cap C' \in \mathcal{C}$. Then, there exists positive constants κ_1, κ_2, c_1 and c_2 such that: for all $\delta > 0$, we have w.p. $\geqslant 1 - \delta$, that $\forall d \in \mathbb{N}^*, \forall n \in \mathbb{N}^*$,

$$|\mathrm{AUC}(s_d) - \mathrm{AUC}(\widehat{s}_d)| \leqslant \kappa_1^{d-1} B(d, n, \delta),$$

and $\forall k \in \{0, \ldots, 2^{d-1} - 1\}$:

$$\left|\alpha(C^*_{d,2k}) - \alpha(C_{d,2k})\right| + \left|\beta(C^*_{d,2k}) - \beta(C_{d,2k})\right| \leqslant \kappa_2^d B(d+1, n, \delta),$$

where $\forall (d, n, \delta) \in \mathbb{N} \times \mathbb{N} \times]0, 1[$:

$$B(d, n, \delta) = \left(\frac{c_1^2 V}{n}\right)^{\frac{1}{2d}} + \left(\frac{c_2^2 \log(1/\delta)}{n}\right)^{\frac{1}{2d}}.$$

Lemma 3.14 implies the following theorem.

Theorem 3.15. *(Clémençon and Vayatis, 2009, Theorem 15)*
Assume that the assumptions of Lemma 3.14 hold. For all $\delta > 0$, there exists a constant c_0 and universal constants c_1, c_2 such that w.p. $\geqslant 1 - \delta$, we have for all $D \geqslant 1$, $n \in \mathbb{N}$,

$$d_1(\widehat{s}_D, s_D) \leqslant c_0^D \left(\left(\frac{c_1^2 V}{n}\right)^{\frac{1}{2D}} + \left(\frac{c_2^2 \log(1/\delta)}{n}\right)^{\frac{1}{2D}}\right),$$

$$d_\infty(\widehat{s}_D, s_D) \leqslant c_0^D \left(\left(\frac{c_1^2 V}{n}\right)^{\frac{1}{2(D+1)}} + \left(\frac{c_2^2 \log(1/\delta)}{n}\right)^{\frac{1}{2(D+1)}}\right).$$

Combining Theorem 3.15 and Proposition 3.12 gives guarantees on the ability of TreeRank to recover the optimal ROC curve.

Corollary 3.16. *(Clémençon and Vayatis, 2009, Corollary 16)*
Assume that the assumptions of Lemma 3.14 hold. Choose $D = D_n$, so that $D_n \sim \sqrt{\log(n)}$ as $n \to \infty$. Then, for all $\delta > 0$, there exists a constant κ such that, w.p. $\geqslant 1 - \delta$: for all $n \in \mathbb{N}$,

$$d_i(\widehat{s}_{D_n}, s^*) \leqslant \exp(-\kappa\sqrt{\log(n)}), \qquad i \in \{1, \infty\}.$$

3.3 Ranking aggregation

3.3.1 Introduction

As pointed out in Korba (2018) (Part 1), ranking data is less considered in the machine learning literature than other types of data because the space of rankings is not a vector space. Hence, many approaches in machine learning cannot directly apply to ranking data. Introduce the space of all permutations of $\{1, \dots, K\}$ as $\mathfrak{S}_K$. The basic notion of mean or median of a sample of n elements $\mathcal{D}_n = \{\sigma_i\}_{i=1}^n \in \mathfrak{S}_K$ in the space of permutations does not have a clear definition, but it does on vector spaces. The mean or the median of a sample in a vector space gives a sensible summary of that sample, and *ranking aggregation* can be defined as the problem of finding a sensible summary in $\mathfrak{S}_K$ for the sample $\mathcal{D}_n$.

Approaches to ranking aggregation can be separated in two groups: those that introduce a distance on the space of rankings and seek a ranking σ^* that minimizes an expected distance to elements of the sample $\mathcal{D}_n$, and those that model explicitly the generation of the σ_i's and derive a notion of true ranking from the estimated parameters of the model.

The natural average $\bar{X}$ of a sample in a inner product space is the minimizer of the average distance between the points of the sample and $\bar{X}$, which justifies the distance-based approach. Precisely, introduce a random sample $\mathcal{Q}_n = \{X_i\}_{i=1}^n$ in the space $\mathcal{X}$ with a scalar product $\langle \cdot, \cdot \rangle$ and an associated norm $\|\cdot\|$. Then, the average $\bar{X}$ of the random sample $\mathcal{Q}_n$ is the solution of the following problem:

$$\min_{x \in \mathcal{X}} E(x) \quad \text{with} \quad E(x) := \sum_{i=1}^n \|X_i - x\|^2.$$

Indeed, simple calculus implies that $E(x) - E(\bar{X}) = n\left\|\bar{X} - x\right\|^2 \geqslant 0$ with $\bar{X} = (1/n)\sum_{i=1}^n X_i$. The most typical example of a distance-based approach to ranking aggregation is the *Kemeny ranking aggregation* (Kemeny, 1959). Formally, it writes on the sample $\mathcal{D}_n$, as:

$$\min_{\sigma \in \mathfrak{S}_K} \sum_{i=1}^n d_\tau(\sigma, \sigma_i), \tag{3.17}$$

where d_τ is the *Kendall τ distance* between two permutations, defined for any two $\sigma, \sigma' \in \mathfrak{S}_{\mathfrak{K}}$ as the number of pairwise disagreements over $\{1, \dots, K\}$. Formally:

$$\begin{aligned} d_\tau(\sigma, \sigma') &:= \sum_{k<l} \mathbb{I}\{(\sigma(k) - \sigma(l))(\sigma'(k) - \sigma'(l)) < 0\}, \\ &= \sum_{k<l} \mathbb{I}\{\sigma \circ \sigma'^{-1}(k) > \sigma \circ \sigma'^{-1}(l)\}. \end{aligned}$$

Since $d_\tau(\sigma, \sigma')$ depends only on the permutation $\sigma \circ \sigma'^{-1}$, authors often consider a function $d_\tau : \mathfrak{S}_K \to \mathbb{R}$ defined as the distance between σ and the identity permutation, with:

$$d_\tau(\sigma) := \sum_{k<l} \mathbb{I}\{\sigma(k) > \sigma(l)\}.$$

A minimizer σ^* of Eq. (3.17) is called a *Kemeny consensus*. The Kemeny consensus for an infinite sample, *i.e.* the solution of $\min_{\sigma \in \mathfrak{S}_K} \mathbb{E}[d_\tau(\sigma, \Sigma)]$, is referred to as *Kemeny median* in

Korba (2018) (Chapter 1). While the Kemeny ranking aggregation procedure satisfies good properties, computing the Kemeny consensus is a NP-hard problem (Cormen et al., 2009, Section 34). We refer to Korba (2018) (Chapter 2 and 3) for an overview of methods that give a good approximation of the Kemeny consensus σ^*. Now, we focus on the notion of aggregation derived from probabilistic models for ranking data, as those are more related to the thesis.

3.3.2 Probabilistic Rankings Models

Probabilistic models for ranking data model the distribution of a random permutation $\Sigma \in \mathfrak{S}_K$ by real-valued parameters. In this context, we consider the observations to aggregate $\mathcal{D}_n = \{\sigma_i\}_{i=1}^n \subset \mathfrak{S}_K$ as *i.i.d.* realizations of the random variable Σ. The most simple distribution considers the *r.v.* Σ to be a slightly modified permutation of a reference permutation σ^*, with the strength of the modifications proportional to a dispersion parameter λ. It is the *Mallows model* presented below.

Definition 3.17 (The Mallows model)**.** *(Korba, 2018, Section 2.2.1)*
The Mallows model is parameterized by a reference ranking σ^* and a dispersion parameter $\lambda \in \mathbb{R}_+$. Let $\sigma \in \mathfrak{S}_K$, then:

$$\mathbb{P}\{\Sigma = \sigma\} = \frac{1}{Z} \exp\left(-\lambda \cdot d_\tau(\sigma, \sigma^*)\right),$$

where $Z = \sum_{\sigma \in \mathfrak{S}_K} \exp\left(-\lambda \cdot d_\tau(\sigma, \sigma^*)\right)$.

Now, we provide a simple technique to sample from the Mallows model, referred to in Mesaoudi-Paul et al. (2018) (Section 2.1). Notice that:

$$d_\tau(\sigma, \sigma^*) = \sum_{1 \leqslant k < l \leqslant K} \mathbb{I}\{\sigma \circ \sigma^{*-1}(k) > \sigma \circ \sigma^{*-1}(l)\},$$

which implies that the Kendall τ distance $d_\tau(\sigma, \sigma^*)$ can be seen as the number of wrongly ranked-elements when resorting the reference ranking σ^* with σ. Precisely, for $l \in \{2, K\}$, introduce the position of the l-th element in the reordering of $\{\sigma^{*-1}(1), \ldots, \sigma^{*-1}(l-1)\}$ by σ as the following value:

$$V_l(\sigma \mid \sigma^*) := \sum_{1 \leqslant k < l} \mathbb{I}\{\sigma \circ \sigma^{*-1}(k) > \sigma \circ \sigma^{*-1}(l)\}.$$

The function $j \mapsto V_l(\sigma \mid \sigma^*)$ is a well-known notion in combinatorics defined as the *Lehmer code* of the permutation $\sigma \circ \sigma^{*-1}$. Then, the distribution of Σ writes as:

$$\mathbb{P}\{\Sigma = \sigma\} = \frac{1}{Z} \prod_{l=2}^{K} \exp\left(-\lambda V_l(\sigma \mid \sigma^*)\right). \tag{3.18}$$

Eq. (3.18) implies that generating from the Mallows model can be done by constructing a random reordering of the reference ranking σ^*. Precisely, consider an empty list and add to this list the element $\sigma^{*-1}(1)$. With probability $\exp(-\lambda)$, insert the second element $\sigma^{*-1}(2)$ before that first element, *i.e.* in position $V_2(\sigma \mid \sigma^*)$, with positions starting at 0. Then, insert sequentially all of the remaining elements $\sigma^{*-1}(l)$ at position $V_l(\sigma \mid \sigma^*) \in \{0, \ldots, l-1\}$ for all $l \in \{3, \ldots, K\}$, where the Lehmer code is generated with, for any $r \in \{0, \ldots, l-1\}$:

$$\mathbb{P}\{V_l(\sigma \mid \sigma^*) = r \mid V_{l-1} \ldots, V_1\} \propto e^{-\lambda r},$$

and $\propto$ denotes proportionality. This approach for generating from a Mallows model is called the *Repeated Insertion Model* (RIM) and was introduced by Doignon et al. (2004). We refer to that last publication for details.

Another popular model is the *Thurstone model*, which simply generates K independent random variables in $\mathbb{R}$ with different distributions, and outputs the natural order Σ on $\mathbb{R}$ of those K variables. Hence, sampling from the Thurstone model is straightforward.

Definition 3.18 (The Thurstone model)**.** *(Korba, 2018, Section 2.2.1)*
Given the independent random variables $X_1, X_2, \dots, X_K$ with different real-valued distributions $X_k \sim P_k$ for $k \in \{1, \dots, K\}$, the probability to observe a ranking σ in the Thurstone model is defined as:

$$\mathbb{P}\{\Sigma = \sigma\} = \mathbb{P}\{X_{\sigma^{-1}(1)} < X_{\sigma^{-1}(2)} < \cdots < X_{\sigma^{-1}(K)}\}.$$

Finally, the Plackett-Luce model is a generalization to full rankings of the pairwise comparison model of Bradley-Terry. That last model assumes that Σ follows the property below, for a preference vector $(w_1, \dots, w_K)$:

$$\mathbb{P}\{\Sigma(k) > \Sigma(l)\} = \frac{w_k}{w_k + w_l}.$$

Definition 3.19 is thus very often referred to as the *Bradley-Terry-Luce-Plackett (BTLP) model*. It was introduced in Bradley and Terry (1952), Luce (1959) and Plackett (1975) and is presented below.

Definition 3.19 (The Plackett-Luce model)**.** *(Korba, 2018, Section 2.2.1)*
Given a preference vector $w = (w_1, \dots, w_K)$, the Plackett-Luce model states that:

$$\mathbb{P}\{\Sigma = \sigma\} = \prod_{k=1}^{K-1} \frac{w_{\sigma^{-1}(k)}}{\sum_{l=k}^{K} w_{\sigma^{-1}(l)}}. \tag{3.19}$$

As expected from the expression of Eq. (3.19), a natural way to sample from the Plackett-Luce model is to sample sequentially the $\Sigma^{-1}(k)$ for each $k \in \{1, \dots, K\}$, *i.e.* the elements at each rank k for the permutation Σ. First, note that $\mathbb{P}\{\Sigma^{-1}(1) = l\} = w_l/(\sum_{j=1}^{K} w_j)$, which means that $\Sigma^{-1}(1)$ follows a multinomial distribution of size 1 with support $S_1 := \{1, \dots, K\}$ and parameters $\{w_l/(\sum_{j=1}^{K} w_j)\}_{l=1}^{K}$. Let $k \in \{2, \dots, K\}$ and denote by S_k the remaining elements, *i.e.* $S_k := S_1 \backslash \{\Sigma^{-1}(1), \dots, \Sigma^{-1}(k-1)\}$, then, for any $l \in S_k$:

$$\mathbb{P}\{\Sigma^{-1}(k) = l \mid \Sigma^{-1}(1), \dots, \Sigma^{-1}(k-1)\} = \frac{w_l}{\sum_{j \in S_k} w_j}.$$

The limitations of the parametric models presented have led authors to consider non-parametric models for rankings, which we do not detail here. We refer to Korba (2018) (Section 2.2.2) for more details. Under the assumption below on a probabilistic model on rankings, the Kemeny consensus writes as a function of the parameters of the model.

Assumption 3.20 (Strict Stochastic Transitivity (SST))**.** *(Korba, 2018, Definition 1.2)*
For any $(k, l) \in \{1, \dots, K\}^2$, introduce the pairwise probability as $p_{k,l} = \mathbb{P}\{\Sigma(k) < \Sigma(l)\}$.
The Strong Stochastic Transitivity (SST) assumption states that: for any $k, l, m \in \{1, \dots, K\}$,

$$p_{k,l} > 1/2 \qquad \text{and} \qquad p_{l,m} > 1/2 \qquad \Rightarrow \qquad p_{k,m} > 1/2.$$

Under Assumption 3.20, the *Kemeny median* is equal to:

$$\sigma^*(k) = 1 + \sum_{l \neq k} \mathbb{I}\{p_{k,l} < 1/2\} \quad \text{for any } k \in \{1, \dots, K\},$$

see Korba (2018) (Proposition 1.3). Assume that the SST assumption holds true. For a model, we can explicit the pairwise probabilities $p_{k,l}$'s involved in the expression of the Kemeny median as functions of its parameters. Therefore, we can derive a sensible solution of the ranking aggregation problem directly from the estimation of the parameters.

3.4 Connections to Present Work

The theory of bipartite ranking presented in this chapter is extended in Part II to our similarity ranking problem (Chapter 5). That theory also helps to grasp the origin of our practical

propositions for similarity ranking (Chapter 7). In Part III, Chapter 10 considers bipartite ranking under fairness constraints. Finally, Chapter 8 relies on probabilistic models for interpreting classification data as partial information about rankings on labels.

In Part II, Chapter 5 frames similarity learning as scoring pairs of features in the product space $\mathcal{X} \times \mathcal{X}$ from the most similar to the least similar, and names this problem similarity ranking. This point of view is natural, as many applications evaluate the performance of similarity functions with metrics based on induced rankings, such as the ROC curve. In this context, Chapter 5 derives guarantees for the pointwise ROC optimization problem and the TreeRank algorithm in the case of similarity ranking. Precisely, it extends the results of Proposition 3.5, Proposition 3.8 and Lemma 3.14, which involves results on U-statistics (Chapter 4).

In Part III, Chapter 10 discusses learning fair scoring functions, with constraints derived from ROC-based criteria. As such, it leverages the discussion on bipartite ranking of this chapter, combined with a decomposition of the ROC (Hsieh and Turnbull, 1996, Theorem 2.1) not presented in this chapter. Finally, Chapter 8 considers the problem of providing a list of the possible labels for an instance, ordered by the likelihood of the label. For that matter, it models the relationship between random labels and an ordering of the labels. Hence, Chapter 8 leverages probabilistic models for rankings (Section 3.3.2), and combines their properties with fast generalization bounds in binary classification (Chapter 2)

Chapter 4

U-statistics

Summary: This chapter is a short introduction to U-statistics, which in their simplest form are averages of the evaluation of a real-valued function on all $n(n-1)/2$ pairs that can be formed with a sample of size n. It provides the necessary tools for for our theoretical results on similarity ranking (Chapter 5) and distributed U-statistics (Chapter 6). To a lesser extent, it is involved in the analysis of pairwise functionals in our work on fair ranking (Chapter 10). The complexity of dealing with U-statistics arises from the dependence of the terms involved in the sum. To extend the results for empirical processes to U-statistics, many statistical tools were introduced. In this chapter, we first detail the derivation of finite-sample guarantees on the supremum of the deviations of U-statistics, following a similar analysis as finite-sample bounds for binary classification (Chapter 2). That analysis also implies analytical expressions for the variance of U-statistics. The main drawback of standard U-statistics is their computational complexity as the sample size n grows. Authors have thus proposed to consider incomplete U-statistics, which average a random sample of B pairs selected by sampling with replacement. We detail in this chapter the guarantees of Clémençon et al. (2016) for incomplete U-statistics, as well as their expression of the variance. Finally, we detail the involvement throughout the thesis of the theory of U-statistics presented here. We refer to Lee (1990) and de la Pena and Giné (1999) for a more detailed presentation of U-statistics.

4.1 Introduction

Most of the statistical machine learning literature focuses on averages of *i.i.d.* random variables, since usual risks can be estimated with that type of estimators, for example in binary classification (Chapter 2) or in bipartite ranking (Chapter 3). For other problems, estimators of the risk can not — or should not — be introduced as averages of *i.i.d.* variables. For example, in the case of metric learning (Bellet et al. (2015a)), empirical evaluations of the risk rely on evaluations of the distance between a pair of independent copies of a random variable. If a realization of the same random variable is involved in two pairs, the terms averaged in an estimator of the risk are not independent, an the whole analysis of Chapter 2 and Chapter 3 do not apply. Computing the average of a carefully selected set of pairs may restore the independence between the elements of the sum. However, the limitations over the possible number of selected independent pairs leads to an inaccurate estimator.

Averages over all pairs of independent random variables are well-known in the literature and can be considered as the simplest form of a broad family of statistics referred to as *U-statistics*. U-statistics are minimum variance unbiased estimators (MVUE) of their expected value (Lee, 1990), and were first introduced by Hoeffding (1948). Hoeffding (1948) proposed decompositions for U-statistics that imply generalizations of the usual finite-sample learning bounds of Chapter 2 to U-statistics, as well as analytical expressions of their variance.

The main drawback of U-statistics is their computational cost as the number of samples n grows, since the number of terms that they average is quadratic in n. A natural approach to dealing with the computational cost of a full U-statistic are incomplete U-statistics, which average a fixed number of pairs, selected at random in the set of all possible pairs. They were introduced in Blom (1976) and we refer to Clémençon et al. (2016) for a detailed account of their statistical accuracy. In this chapter, we focus on one-sample U-statistics of degree 2 — *i.e.* U-statistics formed with only one *i.i.d.* sample and pairs of elements of that sample — and give a more general presentation in Section 6.2.1 of Chapter 6. We refer to Lee (1990) and de la Pena and Giné (1999) for a more details on U-statistics.

Section 4.2 introduces formally U-statistics, and gives a few statistical properties that motivate their study. Section 4.3 presents the theoretical tools for the extension of the learning bounds presented in Chapter 2 to the case of U-statistics. Additionally, those same tools imply expressions of their variance. In Section 4.4, we provide similar results for incomplete U-statistics. Section 4.5 details the implications of the results of this chapter on the original work of the thesis.

4.2 Preliminaries

Introduce a random variable $X \sim P$ in an Euclidean space $\mathcal{X}$ as well as an independent copy X' of X. Consider the estimation of a value that depends of P, denoted by $\theta(P) \in \mathbb{R}$, such that:

$$\theta(P) = \mathbb{E}[h(X, X')], \tag{4.1}$$

where h is a measurable function $\mathcal{X} \times \mathcal{X} \to \mathbb{R}$, called a *kernel*. The value $\theta(P)$ is called a *population characteristic* of P in Hoeffding (1948), and the function θ is called a *functional* of P. Examples of quantities that write as Eq. (4.1) include:

- the variance of a distribution: $\mathcal{X} = \mathbb{R}$ and $h(X, X') = (X - X')^2/2$,
- the Gini mean difference (Hoeffding, 1948, page 297): $\mathcal{X} = \mathbb{R}$ and $h(X, X') = |X - X'|$,
- Kendall's τ: $\mathcal{X} = [0, 1] \times [0, 1]$ and $h((X_1, X_2), (X_1', X_2')) = \mathbb{I}\{(X_1 - X_1')(X_2 - X_2') > 0\}$,

as well as the estimators of the AUC involved in Chapter 3 and all of the statistics considered in the similarity ranking setting presented in Part II.

A simple example of U-statistic is the natural estimator of the population characteristic $\theta(P)$. Formally, consider a sample $\mathcal{D}_n := \{X_i\}_{i=1}^n$ composed of n *i.i.d.* copies of X. The U-statistic $U_n(h)$ constitutes an unbiased estimator of $U(h) := \mathbb{E}[U_n(h)] = \theta(P)$, with:

$$U_n(h) := \frac{2}{n(n-1)} \sum_{i<j} h(X_i, X_j). \tag{4.2}$$

In the rest of the chapter, we assume without loss of generality that the kernel h is symmetric, since we can define the symmetric kernel h' as:

$$h'(x, x') := \frac{1}{2}(h(x, x') + h(x', x)),$$

such that $U_n(h')$ is a U-statistic with symmetric kernel, and also an unbiased estimator of $\theta(P)$. The estimator $U_n(h)$ is said to be symmetric, which means that computing it on a permutation of the sample $\mathcal{D}_n$ yields the same value. This property is true irrelevant of the distribution P. The following property shows the unicity of a symmetric unbiased estimator in a large class of distributions $\mathcal{P}$ over $\mathcal{X}$ is unique.

Theorem 4.1. *(Lee, 1990, Theorem 2, Section 1.1, Chapter 1).*
Consider the class of distributions $\mathcal{P}$ of all distributions with finite support in $\mathcal{X}$. Then, $U_n(h)$ is the unique symmetric unbiased estimator of $\theta(P)$ for any $P \in \mathcal{P}$.

The main complexity when studying $U_n(h)$ is the dependence of the terms involved in the summation in Eq. (4.2). For example, the summation involves the two dependent terms $h(X_1, X_2)$

and $h(X_1, X_3)$. Hence, all of the results for standard empirical means introduced in Chapter 2 do not extend to U-statistics. For that reason, other estimators of $\theta(P)$ could be considered, such as the average $V_n(h)$ of $\lfloor n/2 \rfloor$ independent terms:

$$V_n(h) := \frac{1}{\lfloor n/2 \rfloor} \sum_{i=1}^{\lfloor n/2 \rfloor} h\left(X_i, X_{i+\lfloor n/2 \rfloor}\right). \tag{4.3}$$

Using $V_n(h)$ instead of $U_n(h)$ requires less computational power as we average $O(n)$ terms instead of $O(n^2)$ for $U_n(h)$. Additionally, $V_n(h)$ is covered by the analysis introduced in Chapter 2. However, the estimator $V_n(h)$ is less precise than $U_n(h)$ since it averages way less terms. In that regard, the importance of U-statistics is formally justified by Theorem 4.2 below.

Theorem 4.2. *(Lee, 1990, Theorem 3, Section 1.1, Chapter 1)*
The statistic $U_n(h)$ is a minimum variance unbiased estimator (MVUE) of the parameter $\theta(P)$.

4.3 Properties of Simple U-Statistics

As presented in Eq. (4.3), estimators of $\theta(F)$ that are simple empirical averages exist, but have higher variance than U-statistics. The *first Hoeffding decomposition* (Proposition 4.3 below) shows that U-statistics can be written as the mean of $n!$ dependent simple empirical averages. It is proven simply by rearranging of the terms of the sum. Lemma 4.4 exploits that decomposition, to provide a first approach for extending the results of Chapter 2 on standard averages to U-statistics.

Proposition 4.3 (First Hoeffding decomposition)**.**
(Hoeffding, 1963, Equation 5.5) or (Clémençon et al., 2016, Equation 15)
The U-statistic $U_n(h)$ writes as the mean of $n!$ dependent standard averages. Those averages correspond to evaluations of $V_n(h)$ (Eq. (4.3)) on a permuted sample $\mathcal{D}_n$. Precisely, we have:

$$U_n(h) = \frac{1}{n!} \sum_{\sigma \in \mathfrak{S}_n} \frac{1}{\lfloor n/2 \rfloor} \sum_{i=1}^{\lfloor n/2 \rfloor} h\left(X_{\sigma(i)}, X_{\sigma(i+\lfloor n/2 \rfloor)}\right).$$

Lemma 4.4. *(Clémençon et al., 2008, Lemma A.1)*
Let $\{h_t\}_{t \in T}$ be a family of symmetric measurable functions indexed by $t \in T$, with T some set. Assume that ψ is a convex non-decreasing function. Then, we have:

$$\mathbb{E}\left[\psi\left(\sup_{t \in T} \frac{2}{n(n-1)} \sum_{i<j} h_t(X_i, X_j)\right)\right] \leqslant \mathbb{E}\left[\psi\left(\sup_{t \in T} \frac{1}{\lfloor n/2 \rfloor} \sum_{i=1}^{\lfloor n/2 \rfloor} h_t(X_i, X_{\lfloor n/2 \rfloor + i})\right)\right],$$

under the assumption that the suprema are measurable and the expected values exist.

The main implication of Lemma 4.4 is that the results of Section 2.3.1 hold for U-statistics, a consequence of the Chernoff bound (Proposition 2.4-Chapter 2) that is presented below. The drawback of that analysis is that it gives the same bounds for both $U_n(h)$ and the inaccurate estimator $V_n(h)$ (Eq. (4.3)). Corollary 4.5 below is an extension of Proposition 2.19 (in Chapter 2) to U-statistics. Its proof is a simple combination of Lemma 4.4, Proposition 2.19 and the inequalities $(n-1)/2 \leqslant \lfloor n/2 \rfloor \leqslant n/2$.

Corollary 4.5. *(Clémençon et al., 2016, Proposition 2)*
Let $\mathcal{H}$ be a collection of symmetric kernels, that are indicators of a VC-class of sets of VC-dimension V. Then, for all $\delta \in (0,1)$ and $n \in \mathbb{N}$ such that $n \geqslant 2$, we have: w.p. $\geqslant 1-\delta$,

$$\sup_{h \in \mathcal{H}} |U_n(h) - U(h)| \leqslant \sqrt{\frac{\log(1/\delta)}{n-1}} + \sqrt{\frac{16\log(2) + 16V\log(1+n)}{n-1}}.$$

A more refined version of this inequality can be proven using a chaining argument, see Proposition 2.19 (Chapter 2). It states that, for all $\delta \in (0,1)$ and $n \in \mathbb{N}$ such that $n \geqslant 2$, we have: w.p. $\geqslant 1-\delta$, w.p $\geqslant 1-\delta$,

$$\sup_{h \in \mathcal{H}} |U_n(h) - U(h)| \leqslant \sqrt{\frac{\log(1/\delta)}{n-1}} + 2C\sqrt{\frac{2V}{n}},$$

where $C > 0$ is an universal constant.

As in Chapter 2, tighter bounds are obtained by considering the variance of U-statistics. To compute that variance, we use a linearization technique presented below, called *the second Hoeffding decomposition* (Proposition 4.6). It writes the estimator $U_n(h)$ as the sum of two decorrelated terms, and implies directly a simple expression for the variance of $U_n(h)$. Precisely, it decomposes $U_n(h)$ as its Hájek projection — its projection on standard averages *w.r.t.* the L_2 norm — plus a remainder, that satisfies a "degenerate" property presented in Definition 4.8 below. The second Hoeffding decomposition is proven simply by rearranging of the terms of the sum.

Proposition 4.6 (Second Hoeffding decomposition)**.**
(Hoeffding, 1948, Equation 5.18) or (Lee, 1990, Theorem 3, Section 1.3, Chapter 1)
The U-statistic $U_n(h)$ can be written as the sum between an empirical process and a degenerate U-statistic. Formally:

$$U_n(h) - U(h) = 2T_n(h) + W_n(h),$$

where:

$$T_n(h) := \frac{1}{n}\sum_{i=1}^{n} h_0(X_i) \quad \textit{and} \quad W_n(h) := \frac{2}{n(n-1)}\sum_{i<j} h_1(X_i, X_j),$$

with:

$$h_0(x) = \mathbb{E}[h(x, X')] - U(h) \quad \textit{and} \quad h_1(x, x') := h(x, x') - U(h) - h_0(x) - h_0(x').$$

Corollary 4.7. *Since: $\forall i, j, k, l \in \{1, \ldots, n\}$, with $i \neq j, k \neq l$ and $(i, j) \neq (k, l)$,*

$$\begin{aligned} &\textit{Cov}(h_1(X_i, X_j), h_1(X_k, X_l)) = 0, \qquad \textit{Cov}(h_0(X_i), h_0(X_k)) = 0, \\ &\textit{and} \quad \textit{Cov}(h_0(X_i), h_1(X_k, X_l)) = 0, \end{aligned}$$

we have that:

$$\textit{Var}(U_n(h)) = \frac{4\sigma_0^2}{n} + \frac{2\sigma_1^2}{n(n-1)}.$$

The U-statistic $W_n(h)$ in Proposition 4.6 is a special type of U-statistic called a *degenerate U-statistic*. Its variance is low compared to $T_n(h)$, as implied by Corollary 4.7.

Definition 4.8 (Degenerate U-statistic)**.**
(Hoeffding, 1948, Equation 5.11) or (van der Vaart, 2000, Section 12.3)
Assume that for any $x \in \mathcal{X}$, we have $\mathbb{E}[h(x, X)] = 0$. Or equivalently, that $h_0 = 0$. Then, $U_n(h)$ is said to be a degenerate U-statistic — or stationary of order 1 — and it satisfies:

$$\textit{Var}(U_n(h)) = O(n^{-2}).$$

Clémençon et al. (2008) presented fast learning bounds for empirical risk minimization with U-statistics. For that matter, they introduced a noise assumption (as in Section 2.4) to derive bounds for the term $T_n(h)$. They deal with the term $W_n(h)$ with specific properties of degenerate U-statistics.

For $T_n(h)$, Clémençon et al. (2008) give a noise condition on the variance of $T_n(h)$ directly. However, in Section 2.4.2 of Chapter 2, a condition on the distribution implies a condition on some variance. By Jensen's inequality, we have $\mathrm{Var}(h_0(X)) \leqslant \mathrm{Var}(h(X, X'))$. It shows that any term in the Hájek projection $T_n(h)$ has lower variance than any term of the quantity $V_n(h)$ in Eq. (4.3) (van der Vaart, 2000, Chapter 11). As a result of the variance reduction property of the Hájek projection, fast learning bounds hold under weaker assumptions on the data distribution. This property explains the noise condition of Clémençon et al. (2008). A straightforward application of the analysis used to derive fast learning speeds in Chapter 2 achieves to bound $T_n(h)$.

For $W_n(h)$, we first present fast learning bounds for a finite family of functions, which follow from an union bound of results on single functions (Proposition 4.9). Then, we provide a result for more general classes of functions (Proposition 4.10).

Proposition 4.9. *Consequence of (de la Pena and Giné, 1999, Theorem 4.1.13) Assume that h satisfies $\|h\|_\infty \leqslant \mathcal{M}_h$. Then, for any $(n, \delta) \in \mathbb{N}^* \times (0, 1)$: w.p $\geqslant 1 - \delta$,*

$$|W_n(h)| \leqslant \frac{4c\mathcal{M}_h \log(4/\delta)}{n}.$$

where $c > 0$ is an universal constant.

Proof. We refer to de la Pena and Giné (1999) (Theorem 4.1.13) for the full proof of Proposition 4.9, but detail the application of the result. With the kernel h_1, it gives:

$$\mathbb{P}\{|W_n(h)| > t\} \leqslant 4 \exp\left(-\frac{nt}{4c\mathcal{M}_h}\right),$$

since $|h_1| \leqslant 4\mathcal{M}_h$. Proposition 4.9 follows from inverting the bound. □

Proposition 4.10. *(Arcones and Giné, 1994, Theorem 3.2) Assume that the family of kernels $\mathcal{H}$ is of VC-dimension V, and that any $h \in \mathcal{H}$ satisfies $\|h\|_\infty \leqslant \mathcal{M}_\mathcal{H}$. Then, for any $(n, \delta) \in \mathbb{N} \times (0, 1)$ such that $n \geqslant 2$: w.p $\geqslant 1 - \delta$,*

$$\sup_{h \in \mathcal{H}} |W_n(h)| \leqslant \frac{8\mathcal{M}_\mathcal{H} \log(c/\delta)}{c'(n-1)},$$

where c, c' are constants that depend of V.

Proof. We refer to (Arcones and Giné, 1994, Theorem 3.2) for the full proof of Proposition 4.10, but detail the application of the result. Arcones and Giné (1994) refers to degenerate U-statistics as P-canonical U-statistics, see Equation 2.11 therein. On the other hand, the VC-dimension implies an upper bound on the covering number of a family of functions, as shown in Györfi (2002) (Theorem 1.17). Applying the result to the family of h_1's derived from the elements $h \in \mathcal{H}$, gives, since $\|h_1\|_\infty \leqslant 4\mathcal{M}_\mathcal{H}$:

$$\mathbb{P}\left\{\frac{n-1}{8\mathcal{M}_\mathcal{H}} \cdot \sup_{h \in \mathcal{H}} |W_n(h)| \geqslant t\right\} \leqslant ce^{-c't},$$

where c and c' are positive constants that depend on V. Proposition 4.10 follows from inverting the bound. □

To derive fast learning bounds for U-statistics, it suffices to: 1) repeat the analysis for standard averages (Chapter 2) on $T_n(h)$, while taking into account the specificities regarding the noise condition, 2) combine those with concentration bounds for degenerate U-statistics

4.4 Properties of Incomplete U-Statistics

The main drawback of the complete U-statistic $U_n(h)$ is its computational cost for large n, since the number of terms to average is $n(n-1)/2$ (Eq. (4.2)). Thus, a popular idea is to consider the *incomplete U-statistic* $U_B(h)$, which averages the value of h over B pairs (X_i, X_j) selected at random in the set of all pairs (Section 4.3 of Lee (1990) or Clémençon et al. (2016)). Formally, we have:

$$U_B(h) := \frac{1}{B} \sum_{(i,j) \in \mathcal{D}_B} h(X_i, X_j),$$

where $\mathcal{D}_B$ is a set of cardinality B built by sampling with replacement in the set of all possible pairs $\Lambda := \{(i, j) : i < j\}$ of size $\#\Lambda = n(n-1)/2$. Another approach to reduce the computational cost of U-statistics is to consider estimating $U(h)$ with the full U-statistic $U_m(h)$ on a smaller sample of size m. Replacing $U_n(h)$ by $U_m(h)$ in Corollary 4.5 gives a bound of the order $O(m^{-1/2})$. The following result gives an expression of the variance of the incomplete U-statistic $U_B(h)$, which tends to that of $U_n(h)$ when B tends to infinity, as expected.

Theorem 4.11. *(Clémençon et al., 2016, Equation 21)*
For any $n > 1$ and $B \in \mathbb{N}^*$, the variance of the incomplete U-statistic $U_B(h)$ can be decomposed as:

$$Var(U_B(h)) = \left(1 - \frac{1}{B}\right) Var(U_n(h)) + \frac{1}{B} Var(h(X_1, X_2)).$$

Besides the expression of the variance, Clémençon et al. (2016) (Theorem 6) derived an upper-bound on the deviation of the incomplete U-statistic $U_B(h)$ from the complete one $U_n(h)$.

Theorem 4.12. *(Clémençon et al., 2016, Theorem 6)*
Let $\mathcal{H}$ be a collection of kernels, that satisfies the assumptions of Corollary 4.5, then, for any $n > 1$ and $B \in \mathbb{N}^*$, w.p. $\geqslant 1 - \delta$,

$$\sup_{h \in \mathcal{H}} |U_B(h) - U_n(h)| \leqslant \sqrt{2 \frac{V \log(1 + n(n-1)/2) + \log(2/\delta)}{B}}.$$

The combination of Corollary 4.5 and Theorem 4.12 implies a simple learning bound for incomplete U-statistics. A striking fact, underlined in Clémençon et al. (2016) (Section 3.1), is that it suffices to select $B = O(n)$ pairs to recover the standard learning bound in $O(n^{-1/2})$ presented in Chapter 2. For the same number of evaluated pairs $B = m(m-1)/2 = O(n)$, computing the complete U-statistic $U_m(h)$ on a sample of m pairs, gives a significantly slower bound in $O(n^{-1/4})$. While the observation of Clémençon et al. (2016) gives significant guarantees for large n, in practical cases (finite n) usual learning bounds are quite loose and do not give information about the relative accuracy of an incomplete U-statistics with respect to a complete U-statistic.

4.5 Connections to Present Work

Theoretical results on U-statistics are involved in Chapter 5 and Chapter 6 of Part II, as well as in Chapter 10 of Part III. Precisely, Chapter 5 extends the proofs for bipartite ranking seen in Chapter 3 to the case of similarity ranking, which involves pairwise learning. Chapter 6 introduces estimators of U-statistics in a distributed environment. Finally, Chapter 10 discusses fairness for bipartite ranking, and relies on the empirical AUC as a performance measure.

In Part II, Chapter 5 leverages standard learning bounds for U-statistics (Corollary 4.5) to derive uniform generalization bounds for similarity ranking, defined as the problem of learning a similarity on the product space $\mathcal{X} \times \mathcal{X}$ with a pairwise bipartite ranking objective. The second Hoeffding decomposition (Proposition 4.6), combined with bounds on degenerate U-statistics (Proposition 4.9 or Proposition 4.10), implies faster but data-dependent guarantees. Additionally, our convergence bound for incomplete U-statistics (Theorem 4.12) implies theoretical results for scalable sampling-based methods for similarity ranking. Those results are not a direct application of this chapter, since the pointwise ROC optimization problem presented in Chapter 5 is a constrained optimization. In Chapter 6, the implications on the variance of U-statistics (Theorem 4.11) of the second Hoeffding decomposition (Proposition 4.6) are essential to compute the variance of our distributed estimators for U-statistics.

In Part III, Chapter 10 involves standard learning bounds for U-statistics (Corollary 4.5) to derive generalization guarantees for maximizing the Area Under the ROC Curve (AUC), a pairwise functional criterion for ranking.

Part II

Similarity Ranking

Chapter 5

Similarity Ranking Theory

Summary: The performance of biometric systems — and that of many machine learning techniques — depends on the choice of an appropriate similarity or distance measure on the input space. Similarity learning (or metric learning) aims at building such a measure from training data so that observations with the same (resp. different) label are as close (resp. far) as possible. In this chapter, similarity learning is investigated from the perspective of pairwise bipartite ranking, where the goal is to rank the elements of a database by decreasing order of the probability that they share the same label with some query data point, based on the similarity scores. We study this novel perspective on similarity learning, that we call similarity ranking, through a rigorous probabilistic framework. Our results are an extension of the analysis of bipartite ranking provided in Chapter 3, but the pairwise nature of similarity ranking involves results on U-statistics (Chapter 4). We provide an extensive study of the generalization of pointwise ROC optimization for similarity ranking, completed by an empirical illustration of the fast learning rates that we prove. Then, we proceed to extend the theoretical guarantees of the TreeRank algorithm to similarity ranking with the theory of U-statistics.

5.1 Introduction

Similarity (or distance) functions play a key role in many machine learning algorithms for problems ranging from classification (e.g., k-nearest neighbors) and clustering (e.g., k-means) to dimensionality reduction (van der Maaten and Hinton, 2008) and ranking (Chechik et al., 2010). They are also essential to biometric identification algorithms. The success of such methods is heavily dependent on the relevance of the similarity function to the task and dataset of interest. This has motivated the research in similarity and distance metric learning (Bellet et al., 2015a), a line of work which consists in automatically learning a similarity function from data. This training data often comes in the form of pairwise similarity judgments derived from labels, such as positive (resp. negative) pairs composed of two instances with same (resp. different) label. Most existing algorithms can then be framed as unconstrained optimization problems where the objective is to minimize some average loss function over the set of similarity judgments, see for instance (Goldberger et al., 2004; Weinberger and Saul, 2009; Bellet et al., 2015a). Some generalization bounds for this class of methods have been derived, accounting for the specific dependence structure found in the training similarity judgments (Jin et al., 2009; Bellet and Habrard, 2015; Cao et al., 2016; Mason et al., 2017; Verma and Branson, 2015). We refer to Kulis (2012) and Bellet et al. (2015a) for detailed surveys on similarity and metric learning.

In this chapter, we study similarity learning from the perspective of *pairwise bipartite ranking*, where the goal is to rank the elements of a database by decreasing order of the probability that they share the same label with some query data point. This problem is motivated by many concrete applications: for instance, biometric identification aims to check the claimed identity of an individual by matching their biometric information (e.g. a photo taken at an airport) with a large reference database of authorized people (e.g. of passport photos) (Jain et al., 2011). Given

a similarity function and a threshold, the database elements are ranked in decreasing order of similarity score with the query, and the matching elements are those whose score is above the threshold. In this context, performance criteria are related to the ROC curve associated with the similarity function, *i.e.* the relation between the false positive rate and the true positive rate. Previous approaches have empirically tried to optimize the Area under the ROC curve (AUC) of the similarity function McFee and Lanckriet (2010); Huo et al. (2018), without establishing any generalization guarantees. The AUC is a global summary of the ROC curve which penalizes pairwise ranking errors regardless of the positions in the list. More local versions of the AUC (e.g., focusing on the top of the list) are difficult to optimize in practice and lead to complex nonconvex formulations (Clémençon and Vayatis, 2007; Huo et al., 2018).

In this chapter, we focus on a specific performance criterion, namely *pointwise* ROC *optimization*, which aims at maximizing the true positive rate under a fixed false positive rate. This objective, formulated as a constrained optimization problem, naturally expresses the operational constraints present in many practical scenarios. For instance, in biometric applications such as the one outlined above, the verification system is typically set to keep the proportion of people falsely considered a match below a predefined acceptable threshold (see e.g., Jain et al., 2000, 2004).

Specifically, we derive statistical guarantees for the approach of solving the constrained optimization problem corresponding to the empirical version of our theoretical objective, based on a dataset of n labeled data points. As the empirical quantities involved are not *i.i.d.* averages but rather in the form of U-statistics Lee (1990), our results rely on concentration bounds developed for U-processes Clémençon et al. (2008). We first derive a learning rate of order $O(1/\sqrt{n})$ which holds without any assumption on the data distribution. We then show that one can obtain faster rates under a low-noise assumption on the data distribution, which has the form of a margin criterion involving the conditional quantile.

The chapter is organized as follows. Section 5.2 introduces the proposed probabilistic framework for similarity learning, draws connections to existing approaches and introduces the pointwise ROC optimization problem. In Section 5.3, we derive universal and fast learning rates for the minimizer of the empirical version of our problem, with Section 5.3.3 illustrating the different generalization speeds on a simple toy example. Finally, Section 5.4 presents the extension of the well-known TreeRank algorithm for learning score functions to the case of learning similarity functions as well as an extension of its theoretical guarantees based on the theory of U-statistics.

5.2 Similarity Ranking

In this section, we formulate the supervised similarity learning problem from the perspective of pairwise bipartite ranking, and highlight connections with some popular metric and similarity learning algorithms of the literature.

5.2.1 Similarity Learning as Pairwise Ranking

We consider the (multi-class) classification setting. The random variable Y denotes the output label with values in the discrete set $\{1, \ldots, K\}$ with $K \geqslant 1$, and X is the input random variable, taking its values in a feature space $\mathcal{X} \subset \mathbb{R}^d$ with $d \geqslant 1$ and modeling some information hopefully useful to predict Y. We denote by F the marginal distribution of X and by $\eta(x) = (\eta_1(x), \ldots, \eta_K(x))$ the posterior probability, where $\eta_k(x) = \mathbb{P}\{Y = k \mid X = x\}$ for $x \in \mathcal{X}$ and $k \in \{1, \ldots, K\}$. The distribution of the random pair (X, Y) is entirely characterized by $P = (F, \eta)$. The probability of occurrence of an observation with label $k \in \{1, \ldots, K\}$ is assumed to be strictly positive and denoted by $p_k = \mathbb{P}\{Y = k\}$, and the conditional distribution of X given $Y = k$ is denoted by F_k. Equipped with these notations, we have $F = \sum_{k=1}^{K} p_k F_k$.

Optimal similarity measures. The objective of *similarity learning* can be informally formulated as follows: the goal is to learn, from a training sample $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ composed of $n \geqslant 1$ independent copies of (X, Y), a (measurable) similarity measure $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ such that given two independent pairs (X, Y) and (X', Y') drawn from P, the larger the similarity $s(X, X')$ between two observations, the more likely they are to share the same label.

The set of all similarity measures is denoted by $\mathcal{S}$. The class $\mathcal{S}^*$ of optimal similarity rules naturally corresponds to the set of strictly increasing transforms T of the pairwise posterior probability $\eta(x,x') = \mathbb{P}\{Y = Y' \mid (X,X') = (x,x')\}$, where (X',Y') denotes an independent copy of (X,Y):

$$\{T \circ \eta \,|\, T : Im(\eta) \to \mathbb{R}_+ \text{ borel, strictly increasing}\},$$

and where $Im(\eta)$ denotes the support of $\eta(X,X')$'s distribution. With the notations previously introduced, we have $\eta(x,x') = \sum_{k=1}^K \eta_k(x)\eta_k(x')$ for all $(x,x') \in \mathcal{X}^2$. A similarity rule $s^* \in \mathcal{S}^*$ defines the optimal preorder[1] $\preceq^*$ on the product space $\mathcal{X} \times \mathcal{X}$: for any $(x_1,x_2,x_3,x_4) \in \mathcal{X}^4$, x_1 and x_2 are more similar to each other than x_3 and x_4 if and only if (*i.f.f.*) $\eta(x_1,x_2) \geqslant \eta(x_3,x_4)$, and one writes $(x_3,x_4) \preceq^* (x_1,x_2)$ in this case. For any $x \in \mathcal{X}$, s^* also defines a preorder $\preceq_x^*$ on the input space $\mathcal{X}$, permitting to rank optimally all possible observations by increasing degree of similarity to x: for all $(x_1,x_2) \in \mathcal{X}^2$, x_1 is more similar to x than x_2 (one writes $x_2 \preceq_x^* x_1$) *i.f.f.* $(x,x_2) \preceq^* (x,x_1)$, meaning that $\eta(x,x_2) \leqslant \eta(x,x_1)$. We point out that, despite its simplicity, this framework covers a wide variety of applications, such as the biometric identification problem mentioned earlier in the introduction.

Similarity learning as pairwise bipartite ranking. In view of the objective formulated above, similarity learning can be seen as a *bipartite ranking* problem on the product space $\mathcal{X} \times \mathcal{X}$ where, given two independent realizations (X,Y) and (X',Y') of P, the input *r.v.* is the pair (X,X') and the binary label is $Z = 2\mathbb{I}\{Y = Y'\} - 1$. See Chapter 3 for a statistical learning view of bipartite ranking.

ROC analysis is the gold standard to evaluate the performance of a similarity measure s in this context, *i.e.* to measure how close the preorder induced by s is to $\preceq^*$. The ROC curve of s is the PP-plot $t \in \mathbb{R}_+ \mapsto (\bar{H}_s(t), \bar{G}_s(t))$, where, for all $t \geqslant 0$,

$$H_s(t) = \mathbb{P}\{s(X,X') \leqslant t \mid Z = -1\} \quad \text{and} \quad G_s(t) = \mathbb{P}\{s(X,X') \leqslant t \mid Z = +1\},$$

where possible jumps are connected by line segments and $\bar{F}$ denotes the survival function of any distribution F over $\mathbb{R}$, *i.e.* $\bar{F} = 1 - F$. Hence, it can be viewed as the graph of a continuous function $\alpha \in (0,1) \mapsto \mathrm{ROC}_s(\alpha)$, where $\mathrm{ROC}_s(\alpha) = \bar{G}_s \circ H_s^{-1}(\alpha)$ at any point $\alpha \in (0,1)$ such that $\bar{H}_s \circ \bar{H}_s^{-1}(\alpha) = \alpha$. The curve ROC_s reflects the ability of s to discriminate between pairs with same labels and pairs with different labels: the stochastically smaller than H_s the distribution G_s is, the higher the associated ROC curve. Note that it corresponds to the type I error vs power plot of the statistical test $\mathbb{I}\{s(X,X') > t\}$ when the null hypothesis stipulates that X and X' have different marginal distribution (*i.e.* $Y \neq Y'$). A similarity measure s_1 is said to be more accurate than another similarity s_2 when $\mathrm{ROC}_{s_2}(\alpha) \leqslant \mathrm{ROC}_{s_1}(\alpha)$ for any $\alpha \in (0,1)$.

Pointwise ROC **optimization.** In many applications, one is interested in finding a similarity function which optimizes the ROC curve at a particular point $\alpha \in (0,1)$. The superlevel sets of similarity functions in $\mathcal{S}^*$ define the solutions of pointwise ROC optimization problems in this context. In the above framework, it indeed follows from Neyman Pearson's lemma that the test statistic of type I error less than α with maximum power is the indicator function of the set $R_\alpha^* = \{(x,x') \in \mathcal{X}^2 : \eta(x,x') \geqslant Q_\alpha^*\}$, where Q_α^* is the conditional quantile of the *r.v.* $\eta(X,X')$ given $Z = -1$ at level $1 - \alpha$. Restricting our attention to similarity functions bounded by 1, this corresponds to the unique solution of the following problem:

$$\max_{s:\mathcal{X}^2 \to [0,1],\ \text{borel}} R^+(s) \quad \text{subject to} \quad R^-(s) \leqslant \alpha, \tag{5.1}$$

where $R^+(s) = \mathbb{E}[s(X,X') \mid Z = +1]$ is referred to as *positive risk* and $R^-(s) = \mathbb{E}[s(X,X') \mid Z = -1]$ as the *negative risk*.

5.2.2 Connection to Metric Learning

We point out that the similarity learning framework described above can be equivalently described in terms of learning a dissimilarity measure (or pseudo distance metric) $D : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$. In

[1] A preorder on a set $\mathcal{X}$ is any reflexive and transitive binary relationship on $\mathcal{X}$. A preorder is an order if, in addition, it is antisymmetrical.

this case, the pointwise ROC optimization problem (5.1) translates into:

$$\min_{D:\mathcal{X}^2\to[0,1]} \mathbb{E}\left[D(X,X') \mid Z=+1\right] \quad \text{subject to} \quad \mathbb{E}\left[D(X,X') \mid Z=-1\right] \geqslant 1-\alpha. \tag{5.2}$$

A large variety of practical similarity and distance metric learning algorithms have been proposed in the literature, all revolving around the same idea that a good similarity function should output large scores for pairs of points in the same class, and small scores for pairs with different label. They differ from one another by the class of metric/similarity functions considered, and by the kind of objective function they optimize (see Bellet et al., 2015a, for a comprehensive review). In any case, ROC curves are commonly used to evaluate metric learning algorithms when the number of classes is large (see for instance Guillaumin et al. (2009); Köstinger et al. (2012); Shen et al. (2012), which makes our framework very relevant in practice. Several popular algorithms optimize an empirical version of Problems (5.1)-(5.2), often in their unconstrained version as in Liu et al. (2010) and Xie and Xing (2014). We argue here in favor of the constrained version as the parameter α has a direct correspondence with the point $\mathrm{ROC}_S(\alpha)$ of the ROC curve, unlike the unconstrained case presented below.

A remarkable fact is that the superlevel set R^*_α of the pairwise posterior probability $\eta(x,x')$ is the measurable subset R of $\mathcal{X}^2$ that minimizes the cost-sensitive classification risk:

$$p(1-Q^*_\alpha)\mathbb{P}\left\{(X,X') \notin R \mid Z=+1\right\} + (1-p)Q^*_\alpha \mathbb{P}\left\{(X,X') \in R \mid Z=-1\right\}, \tag{5.3}$$

where $p = \mathbb{P}\{Z=+1\} = \sum_{k=1}^K p_k^2$. Hence, the solution of Eq. (5.1) corresponds to the solution of the minimization of Eq. (5.3). Notice however that the asymmetry factor, namely the quantile Q^*_α, is unknown in practice, just like the *r.v.* $\eta(X,X')$. For this reason, one typically considers the problem of maximizing:

$$R^+(s) - \lambda R^-(s), \tag{5.4}$$

for different values of the constant $\lambda > 0$. The performance in terms of ROC curve can only be analyzed *a posteriori*, and the value λ thus needs to be tuned empirically by model selection techniques. We focus here on the constrained version as the parameter α has a direct correspondence with the point $\mathrm{ROC}_s(\alpha)$ of the ROC curve, unlike the unconstrained version.

Interestingly, our framework sheds light on MMC, the seminal metric learning algorithm of Xing et al. (2002) originally designed for clustering with side information. MMC solves the empirical version of (5.2) with α fixed to 0. This is because MMC optimizes over a class of distance functions with unbounded values, hence modifying α does not change the solution (up to a scaling factor). We note that by choosing a bounded family of distance functions, one can use the same formulation to optimize the pointwise ROC curve.

5.3 Statistical Guarantees for Generalization

Pointwise ROC optimization problems have been investigated from a statistical learning perspective by Scott and Nowak (2005) and Clémençon and Vayatis (2010) in the context of binary classification, which we presented in Chapter 3. The major difference with the present framework lies in the pairwise nature of the quantities appearing in Problem (5.1) and, consequently, in the complexity of its empirical version.

In this section, we present the extension of those results to the problem of learning similarities. First, we deal with the derivation of uniform learning rates, *i.e.* learning rates that do not depend on the data distribution, which relies on the extension of usual concentration inequalities for U-statistics. Second, we present data-dependent fast rates, which requires less restrictive assumptions on the data distribution than in bipartite ranking, which stems from the variance-reducing property of the Hájek projection of a U-statistic.

5.3.1 Uniform Rates for Pointwise ROC Optimization

Natural statistical estimates for the positive risk $R^+(s)$ and the negative risk $R^-(s)$ (5.1) computed on the training sample $\mathcal{D}_n = \{(X_1, Y_1),\ \ldots,\ (X_n, Y_n)\}$ are given by:

$$\widehat{R}_n^+(s) = \frac{1}{n_+} \sum_{1 \leqslant i < j \leqslant n} s(X_i, X_j) \cdot \mathbb{I}\{Y_i = Y_j\}, \tag{5.5}$$

$$\widehat{R}_n^-(s) = \frac{1}{n_-} \sum_{1 \leqslant i < j \leqslant n} s(X_i, X_j) \cdot \mathbb{I}\{Y_i \neq Y_j\}, \tag{5.6}$$

where $n_+ = \sum_{1 \leqslant i < j \leqslant n} \mathbb{I}\{Y_i = Y_j\} = n(n-1)/2 - n_-$. It is important to note that these quantities are not *i.i.d.* averages, since several pairs involve each *i.i.d.* sample. This breaks the analysis presented in Chapter 3. We can however observe that:

$$U_n^+(s) := \frac{2n_+}{n(n-1)} \cdot \widehat{R}_n^+(s) \quad \text{and} \quad U_n^-(s) := \frac{2n_-}{n(n-1)} \cdot \widehat{R}_n^-(s) \tag{5.7}$$

are U-statistics of degree two — averages of a function of the pairs of an *i.i.d.* sample (see Chapter 4) — with respective symmetric kernels:

$$h_+((x,y),(x',y')) = s(x,x') \cdot \mathbb{I}\{y = y'\} \quad \text{and} \quad h_-((x,y),(x',y')) = s(x,x') \cdot \mathbb{I}\{y \neq y'\}.$$

We will therefore be able to use existing representation tricks to derive concentration bounds for U-processes (collections of U-statistics indexed by classes of kernel functions), under appropriate complexity conditions, see *e.g.* Dudley (1999).

We thus investigate the generalization ability of solutions obtained by solving the empirical version of Problem (5.1), where we also restrict the domain to a subset $\mathcal{S}$ of similarity functions bounded by 1, and we assume $\mathcal{S}$ has controlled complexity (*e.g.* finite VC dimension). Finally, we replace the target level α by $\alpha + \phi$, where $\phi \geqslant 0$ is some tolerance parameter that should be of the same order as the maximal deviation $\sup_{s \in \mathcal{S}} |\widehat{R}_n^-(s) - R^-(s)|$. This leads to the following empirical problem:

$$\max_{s \in \mathcal{S}} \widehat{R}_n^+(s) \quad \text{subject to} \quad \widehat{R}_n^-(s) \leqslant \alpha + \phi. \tag{5.8}$$

We are now ready to state our universal learning rate, describing the generalization capacity of solutions of the constrained optimization program (5.8) under specific conditions for the class $\mathcal{S}_0$ of similarity functions and a suitable choice of the tolerance parameter Φ. This result can be established by combining Corollary 4.5 in Chapter 4, an extension of usual concentrations inequalities for empirical processes to U-statistics, with the derivations of Clémençon and Vayatis (2010, Theorem 10 therein).

Theorem 5.1. *Suppose that the proposed family $\mathcal{S}$ is a VC-major class of functions of finite VC-dimension V, and that $s(x,x') \leqslant 1$ for all $s \in \mathcal{S}$ and any $(x,x') \in \mathcal{X}^2$. Assume also that there exists a constant $\kappa \in (0,1)$ such that $\kappa \leqslant \sum_{k=1} p_k^2 \leqslant 1 - \kappa$. For all $\delta \in (0,1)$ and $n > 1$, set*

$$\phi(\delta, n) = 2C\kappa^{-1}\sqrt{\frac{V}{n}} + 2\kappa^{-1}(1 + \kappa^{-1})\sqrt{\frac{\log(3/\delta)}{n-1}},$$

where C is a positive universal constant, and consider a solution $\widehat{s}_n$ of the constrained minimization problem (5.8) with $\phi = \phi(n, \delta/2)$. Then, for any $\delta \in (0,1)$, we have simultaneously, $\forall n \geqslant 1 + 4\kappa^{-2}\log(3/\delta)$, w.p. $\geqslant 1 - \delta$:

$$R^+(\widehat{s}_n) \geqslant \mathrm{ROC}_{s*}(\alpha) - \phi(n, \delta/2) - \Big\{\mathrm{ROC}_{s*}(\alpha) - \sup_{s \in \mathcal{S}:\ R^-(s) \leqslant \alpha} R^+(s)\Big\}, \tag{5.9}$$

and

$$R^-(\widehat{s}_n) \leqslant \alpha + \phi(n, \delta/2). \tag{5.10}$$

Proof. As presented in Chapter 3, precisely Proposition 3.5, the bilateral control of each risk R^+, R^- implies guarantees for the constrained problem. Set $p = \sum_{k=1}^{K} p_k^2$ and observe that we have: $\forall n > 1$,

$$\sup_{s\in\mathcal{S}}\left|\widehat{R}_n^+(s) - R^+(s)\right| \leqslant p^{-1}\left|\frac{2n_+}{n(n-1)} - p\right| + p^{-1}\sup_{s\in\mathcal{S}}\left|U_n^+(s) - \mathbb{E}[U_n^+(s)]\right|.$$

A generalization of Hoeffding's inequality for U-statistics (Serfling, 1980, Section 5.6, Theorem A), presented as Lemma 4.4 in Chapter 4, gives that, *w.p.* $\geqslant 1-\delta$,

$$\left|\frac{2n_+}{n(n-1)} - p\right| < \sqrt{\frac{\log(2/\delta)}{n-1}}.$$

Combining this proposition with Corollary 4.5 in Chapter 4, gives that: $\forall n > 1$, *w.p.* $\geqslant 1-\delta$,

$$\sup_{s\in\mathcal{S}}\left|\widehat{R}_n^+(s) - R^+(s)\right| \leqslant \frac{2C}{p}\sqrt{\frac{V}{n}} + \frac{3}{p}\sqrt{\frac{\log(3/\delta)}{n-1}}.$$

Similarly, we obtain, with $q = 1-p$:

$$\sup_{s\in\mathcal{S}}\left|\widehat{R}_n^-(s) - R^-(s)\right| \leqslant q^{-1}\left|\frac{2n_-}{n(n-1)} - q\right| + q^{-1}\sup_{s\in\mathcal{S}}\left|U_n^-(s) - \mathbb{E}[U_n^-(s)]\right|,$$

which gives with the exact same reasoning that: $\forall n > 1$, *w.p.* $\geqslant 1-\delta$,

$$\sup_{s\in\mathcal{S}}\left|\widehat{R}_n^-(s) - R^-(s)\right| \leqslant \frac{2C}{q}\sqrt{\frac{V}{n}} + \frac{3}{q}\sqrt{\frac{\log(3/\delta)}{n-1}}.$$

which gives the right order of convergence. The proof is then finished by following the proof of Theorem 10 in Clémençon and Vayatis (2010). □

The last term on the right hand side of (5.9) should be interpreted as the bias of the statistical learning problem (5.8), which depends on the richness of class $\mathcal{S}$. This term vanishes when $\mathbb{I}\{(x,x') \in R_\alpha^*\}$ belongs to $\mathcal{S}$. Choosing a class yielding a similarity rule of highest true positive rate with large probability can be tackled by means of classical model selection techniques, based on resampling methods or complexity penalization.

Except for the minor condition stipulating that the probability of occurrence of "positive pairs" $\sum_{k=1}^{K} p_k^2$ stays bounded away from 0 and 1, the generalization bound stated in Theorem 5.1 holds whatever the probability distribution of (X, Y). Beyond such universal results, we investigate situations where rates faster than $O(1/\sqrt{n})$ can be achieved by solutions of (5.8), which are presented in the section below.

5.3.2 Fast Rates for Pointwise ROC Optimization

Such fast rates results exist for binary classification under the so-called Mammen-Tsybakov noise condition, as presented in Chapter 2. By means of a variant of the Bernstein inequality for U-statistics, we can establish fast rate bounds under the following condition on the data distribution.

Assumption 5.2. *Let $\alpha \in (0,1)$. There exist $a \in [0,1]$ and a constant $c > 0$ such that, almost surely,*

$$\mathbb{E}_{X'}\left[\left|\eta(X,X') - Q_\alpha^*\right|^{-a}\right] \leqslant c.$$

This noise condition is similar to that introduced by Mammen and Tsybakov (1995) for the binary classification framework, except that the threshold $1/2$ is replaced here by the conditional quantile Q_α^*. It characterizes "nice" distributions for the problem of ROC optimization at point α: it essentially ensures that the pairwise posterior probability is bounded away from Q_α^* with high probability.

The condition expressed in Assumption 5.2 is weaker than that for standard bipartite ranking, *i.e.* Assumption 3.6, which is possible thanks to the variance reducing property of Hájek projections. For example, the noise condition is automatically fulfilled for any $a \in (0,1)$ when, for almost every point x with respect to the measure induced by X, $\eta(x, X')$ has an absolutely continuous distribution and bounded density. Intuitively, this assumption means that the problem of ranking elements modeled by X according to their similarity with an element $x \in A$ is somewhat easy (almost-surely). The implication is proven precily in the result below, with a proof that follows the same arguments as Clémençon et al. (2008) (Corollary 8).

Proposition 5.3. *Assume that there exist $A \subset \mathcal{X}, \mathbb{P}(X \in A) = 1$, such that for all $x \in A$, the random variable $\eta(x, X)$ has an absolutely continuous distribution on $[0,1]$ and its density is bounded by B. Then: for any $\epsilon > 0$,*

$$\mathbb{E}_{X'}\left[\left|\eta(X, X') - Q^*_\alpha\right|^{-1+\epsilon}\right] \leqslant \frac{2B}{\epsilon} \quad \textit{almost surely,}$$

which implies that the fast rate of convergence of Theorem 5.5 applies for any $a \in (0,1)$.

Proof. Let $x \in A$ and h_x be the density of $\eta(x, X)$, with $h_x \leqslant B$. Hence, for any $a \in (0,1)$,

$$\begin{aligned}
\mathbb{E}_{X'}\left[\left|\eta(x, X) - Q^*_\alpha\right|^{-a}\right] &= \int_0^1 |z - Q^*_\alpha|^{-a} h_x(z) dz, \\
&\leqslant B\left(\int_0^{Q^*_\alpha} (Q^*_\alpha - z)^{-a} dz + \int_{Q^*_\alpha}^1 (z - Q^*_\alpha)^{-a} dz\right), \\
&\leqslant B\left(\frac{Q^{*1-a}_\alpha}{1-a} + \frac{(1-Q^*_\alpha)^{1-a}}{1-a}\right), \\
&\leqslant \frac{2B}{1-a}.
\end{aligned}$$

□

The fast bounds result is based on the second Hoeffding decomposition (see Proposition 4.6 in Chapter 4) of the U-statistic U^+_n (see Eq. (5.7)), with an independent analysis of the sum of *i.i.d.* terms and the degenerate U-statistic (see Definition 4.8 in Chapter 4) remainder. Denote by $s^*(x, x') = \mathbb{I}\{(x, x') \in R^*_\alpha\}$ the optimal similarity function. We assume for simplicity that it belongs to $\mathcal{S}$, but the result can be extended using the local analysis arguments presented in Boucheron et al. (2005) (Section 5.3.5). For any $s \in \mathcal{S}$, the statistic:

$$\Delta_n(s) = \left(U^+_n(s) - \mathbb{E}\left[U^+_n(s)\right]\right) - \left(U^+_n(s^*) - \mathbb{E}\left[U^+_n(s^*)\right]\right), \tag{5.11}$$

is a U-statistic based on $\mathcal{D}_n$ with kernel Q_s given by:

$$Q_s((x,y),(x',y')) = \mathbb{I}\left\{y = y'\right\}\left(s(x,x') - s^*(x,x')\right) - \mathbb{E}\left[\mathbb{I}\left\{Y = Y'\right\}\left(s(X,X') - s^*(X,X')\right)\right].$$

The second Hoeffding decomposition of U-statistics, presented in Chapter 4, leads to the following decomposition of the U-statistic $\Delta_n(s)$:

$$\Delta_n(s) = 2T_n(s) + W_n(s), \tag{5.12}$$

where:

$$T_n(s) = \frac{1}{n}\sum_{i=1}^n q_s(X_i, Y_i) \quad \text{and} \quad W_n(s) = \frac{2}{n(n-1)} \sum_{1 \leqslant i < j \leqslant n} \widehat{q}_s\left((X_i, Y_i), (X_j, Y_j)\right),$$

with:

$$\begin{aligned}
q_s(x,y) &= \mathbb{E}\left[Q_s\left((X,Y),(x,y)\right)\right], \\
\widehat{q}_s\left((x,y),(x',y')\right) &= Q_s\left((x,y),(x',y')\right) - q_s(x,y) - q_s(x',y').
\end{aligned}$$

While in the binary classification or bipartite ranking settings, the noise condition implied a bound on the variance of an empirical process, our weak similarity ranking noise condition bounds the variance of the Hájek projection of a U-statistic, as seen in the lemma below.

Lemma 5.4. *Suppose that the assumptions of Theorem 5.1 are satisfied and that Assumption 5.2 holds true with parameter $a \in [0,1]$. Then, for any $s \in \mathcal{S}$, we have:*

$$Var(q_s(X,Y)) \leqslant c\left[(1-Q^*_\alpha)p(R^+(s^*)-R^+(s)) + (1-p)Q^*_\alpha(R^-(s)-R^-(s^*))\right]^a. \tag{5.13}$$

Proof. For any $s \in \mathcal{S}$ and $u \in [0,1]$, we introduce $\mathcal{R}_{s,u} := \{(x,x') \mid s(x,x') \geqslant u\}$. Then, s writes as an integral of indicators of its level sets (van der Vaart and Wellner, 1996, Lemma 2.6.13):

$$s(X,X') = \int_0^1 \mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\}\, du. \tag{5.14}$$

Additionally, we have $s^*(X,X') = \mathbb{I}\{(X,X') \in R^*_\alpha\}$. The definition of q_s implies that:

$$\mathrm{Var}\left(q_s(X,Y)\right) = \mathrm{Var}\left[\left(\mathbb{E}_{X',Y'}\left[\mathbb{I}\{Y=Y'\}(s(X,X')-s^*(X,X'))\right]\right)^2\right].$$

Therefore, the fact that $\mathrm{Var}(Z) \leqslant \mathbb{E}[Z^2]$ gives:

$$\mathrm{Var}\left(q_s(X,Y)\right) \leqslant \mathbb{E}_{X,Y}\left[\left(\mathbb{E}_{X',Y'}\left[\mathbb{I}\{Y=Y'\}(s(X,X')-s^*(X,X'))\right]\right)^2\right]. \tag{5.15}$$

Injecting Eq. (5.14) into Eq. (5.15) gives, from Fubini's theorem:

$$\mathrm{Var}\left(q_s(X,Y)\right) \leqslant \mathbb{E}_{X,Y}\left[\left(\int_0^1 \mathbb{E}_{X',Y'}\left[\mathbb{I}\{Y=Y'\}\left(\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\} - \mathbb{I}\{(X,X') \in R^*_\alpha\}\right)\right] du\right)^2\right].$$

Jensen's inequality applied to the integral over u, followed by Fubini's theorem gives:

$$\mathrm{Var}\left(q_s(X,Y)\right) \leqslant \int_0^1 \mathbb{E}_{X,Y}\left[\left(\mathbb{E}_{X',Y'}\left[\mathbb{I}\{Y=Y'\}\left(\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\} - \mathbb{I}\{(X,X') \in R^*_\alpha\}\right)\right]\right)^2\right] du.$$

We introduce γ (we hide dependencies for readability reasons) as:

$$\begin{aligned}\gamma :&= \mathbb{I}\{Y=Y'\}\left(\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\} - \mathbb{I}\{(X,X') \in R^*_\alpha\}\right)\\ &= \mathbb{I}\{(X,X') \in \mathcal{R}_{s,u} \Delta R^*_\alpha\}\left(1 - 2\mathbb{I}\{(X,X') \in R^*_\alpha\}\right)\mathbb{I}\{Y=Y'\},\end{aligned}$$

then $|\gamma| \leqslant \mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\Delta R^*_\alpha\}$ and the bound implies that:

$$\mathrm{Var}\left(q_s(X,Y)\right) \leqslant \int_0^1 \mathbb{E}_X\left[\left(\mathbb{E}_{X'}\left[\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\Delta R^*_\alpha\}\right]\right)^2\right] du. \tag{5.16}$$

Cauchy-Schwartz's inequality gives:

$$\begin{aligned}&\mathbb{E}_{X'}\left[\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\Delta R^*_\alpha\}\right]\\ &\quad\leqslant \left(\mathbb{E}_{X'}\left[\left|\eta(X,X') - Q^*_\alpha\right|^{-a}\right]\right)^{\frac{1}{2}} \times \left(\mathbb{E}_{X'}\left[\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\Delta R^*_\alpha\} \cdot \left|\eta(X,X') - Q^*_\alpha\right|^a\right]\right)^{\frac{1}{2}},\end{aligned}$$

and Assumption 5.2 implies:

$$\mathbb{E}_{X'}\left[\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\Delta R^*_\alpha\}\right] \leqslant \sqrt{c} \times \left(\mathbb{E}_{X'}\left[\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\Delta R^*_\alpha\}\left|\eta(X,X') - Q^*_\alpha\right|^a\right]\right)^{\frac{1}{2}}.$$

Plugging that last result into Eq. (5.16) gives:

$$\mathrm{Var}\left(q_s(X,Y)\right) \leqslant c \cdot \int_0^1 \mathbb{E}_X\left[\mathbb{E}_{X'}\left[\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\Delta R^*_\alpha\} \cdot \left|\eta(X,X') - Q^*_\alpha\right|^a\right]\right] du.$$

Since $a \in [0,1]$, the function $x \mapsto x^a$ is concave. Therefore, Jensen's inequality applied for integration with respect to X', X and u sequentially, gives:

$$\mathrm{Var}\left(q_s(X,Y)\right) \leqslant c\left(\int_0^1 \mathbb{E}\left[\mathbb{I}\{(X,X') \in \mathcal{R}_{s,u}\Delta R^*_\alpha\} \cdot \left|\eta(X,X') - Q^*_\alpha\right|\right] du\right)^a. \tag{5.17}$$

The right-hand-side of Eq. (5.17) is very similar to the expression of the excess risk in binary classification presented in Eq. (2.1). We now relate it to the right-hand side of Eq. (5.13):

$$\begin{aligned} R^+(s) - R^+(s^*) &= (1/p) \cdot \mathbb{E}\left[\mathbb{I}\{Y = Y'\}(s(X, X') - s^*(X, X'))\right], \\ &= (1/p) \cdot \int_0^1 \mathbb{E}\left[\eta(X, X')(\mathbb{I}\{(X, X') \in \mathcal{R}_{s,u}\} - \mathbb{I}\{(X, X') \in R^*_\alpha\})\right] du, \end{aligned}$$

by Fubini's theorem. Simple calculus gives:

$$R^+(s) - R^+(s^*) = (1/p) \cdot \int_0^1 \mathbb{E}\left[\left(1 - 2 \cdot \mathbb{I}\{(X, X') \in R^*_\alpha\}\right) \cdot \eta(X, X') \cdot \mathbb{I}\{(X, X') \in \mathcal{R}_{s,u} \Delta R^*_\alpha\}\right] du,$$

The definition of Q^*_α implies that:

$$\left(1 - 2 \cdot \mathbb{I}\{(X, X') \in R^*_\alpha\}\right)\left(\eta(X, X') - Q^*_\alpha\right) = -\left|\eta(X, X') - Q^*_\alpha\right|,$$

and we have:

$$\begin{aligned} R^+(s) - R^+(s^*) = &- (1/p) \cdot \int_0^1 \mathbb{E}\left[\left|\eta(X, X') - Q^*_\alpha\right| \cdot \mathbb{I}\{(X, X') \in \mathcal{R}_{s,u} \Delta R^*_\alpha\}\right] du \\ &+ (Q^*_\alpha/p) \cdot \int_0^1 \mathbb{E}\left[\mathbb{I}\{(X, X') \in R_{s,u}\} - \mathbb{I}\{(X, X') \in R^*_\alpha\}\right] du, \end{aligned} \tag{5.18}$$

since:

$$\mathbb{I}\{(X, X') \in \mathcal{R}_{s,u} \Delta R^*_\alpha\} \cdot (1 - 2 \cdot \mathbb{I}\{(X, X') \in R^*_\alpha\}) = \mathbb{I}\{(X, X') \in R_{s,u}\} - \mathbb{I}\{(X, X') \in R^*_\alpha\}.$$

Eq. (5.14) combined with Fubini's theorem gives:

$$\begin{aligned} \int_0^1 \mathbb{E}\left[\mathbb{I}\{(X, X') \in R_{s,u}\} - \mathbb{I}\{(X, X') \in R^*_\alpha\}\right] du &= \mathbb{E}[s(X, X')] - \mathbb{E}[s^*(X, X')], \\ &= p(R_+(s) - R_+(s^*)) + (1 - p)(R_-(s) - R_-(s^*)). \end{aligned} \tag{5.19}$$

Combining Eq. (5.19), Eq. (5.18) and Eq. (5.17) and rearranging the terms completes the proof. □

Using this property, we can derive fast learning rates.

Theorem 5.5. *Suppose that the assumptions of Theorem 5.1 and Lemma 5.4 are satisfied, that the class $\mathcal{S}$ is finite of size M, and that the optimal similarity rule $s^*_\alpha(x, x') = \mathbb{I}\{(x, x') \in R^*_\alpha\}$ belongs to $\mathcal{S}$. Fix $\delta > 0$. Then, there exists a constant C', depending on δ, κ, Q^*_α, a, c and M such that, for any $n > 1$, w.p. $\geqslant 1 - \delta$,*

$$\mathrm{ROC}_{s*}(\alpha) - R^+(\widehat{s}_n) \leqslant C' n^{-(2+a)/4} \quad \textit{and} \quad R^-(\widehat{s}_n) \leqslant \alpha + 2\phi(n, \delta/2).$$

Proof. The component $W_n(s)$ is a degenerate U-statistic (see Definition 4.8 in Chapter 4), meaning that, for all (x, y),

$$\mathbb{E}\left[\widehat{q}_s\left((x, y), (X, Y)\right)\right] = 0 \text{ almost-surely.}$$

The supremum of the family $\{W_n(s)\}_{s \in \mathcal{S}}$ is controlled using the results in Chapter 4, precisely Proposition 4.9 for finite families and Proposition 4.10 for more general classes of families. Precisely, from de la Pena and Giné (1999) (Theorem 4.1.13 therein), we have since $W_n(s)$ is a degenerate U-statistic, that: *w.p.* $\geqslant 1 - \delta$,

$$\sup_{s \in \mathcal{S}} |W_n(s)| \leqslant \frac{2C \log(4M/\delta)}{n}, \tag{5.20}$$

where C is an universal constant. This result shows that the second term in the decomposition (5.12) is uniformly negligible with respect to the first term $T_n(s)$.

We consider Lemma 5.4 as an analogue of Lemma 11 of Clémençon and Vayatis (2010) (in the preliminaries as Proposition 3.8 of Chapter 3) adapted to the case of similarity ranking. The fast

rate bound stated in Theorem 5.5 then follows from the application of Talagrand's inequality (or Bernstein's inequality when $\mathcal{S}$ is of finite cardinality), following the steps of Boucheron et al. (2005) (subsection 5.2) or Clémençon and Vayatis (2010) (Theorem 12).

Since $|Q_s| \leqslant 2$, Bernstein's inequality combined with the union bound gives that, if $\mathcal{S}$ is of cardinality M: *w.p.* $\geqslant 1-\delta$, for all $s \in \mathcal{S}$,

$$T_n(s) \leqslant \frac{4\log(M/\delta)}{3n} + \sqrt{\frac{2\mathrm{Var}(q_s(X,Y))\log(M/\delta))}{n}}. \tag{5.21}$$

Combining Eq. (5.21) and Eq. (5.20) give that: *w.p.* $\geqslant 1-\delta$, for all $s \in \mathcal{S}$

$$\Delta_n(s) \leqslant \frac{C'\log(5M/\delta)}{n} + \sqrt{\frac{2\mathrm{Var}(q_s(X,Y))\log(5M/\delta))}{n}}. \tag{5.22}$$

where $C' = 2C + 4/3$.

The proof of Theorem 5.1 may be adapted to the finite class setting. Formally, introducing the tolerance term:

$$\phi(n,\delta)' = 2\kappa^{-1}(1+\kappa^{-1})\sqrt{\frac{\log(2(M+1)/\delta)}{n-1}},$$

where M is the cardinal of the proposition class $\mathcal{S}$, we have simultaneously *w.p* $\geqslant 1-\delta$, that:

$$R^+(\widehat{s}_n) \geqslant \sup_{s\in\mathcal{S}:\ R^-(s)\leqslant\alpha} R^+(s) - \phi(n,\delta/2)' \quad \text{and} \quad R^-(\widehat{s}_n) \leqslant \alpha + \phi(n,\delta/2)', \tag{5.23}$$

$$\sup_{s\in\mathcal{S}} \left|R_n^+(s) - R^+(s)\right| \leqslant \phi(n,\delta/2)'. \tag{5.24}$$

Eq. (5.24) implies that s^* satisfies the constraint of the ERM problem Eq. (5.8), hence $R_n^+(\widehat{s}_n) - R_n^+(s^*) \geqslant 0$. It follows that :

$$\begin{aligned}\Delta_n(\widehat{s}_n) &= \frac{2n_+}{n(n-1)}\left(R_n^+(\widehat{s}_n) - R_n^+(s^*)\right) + p\left(R^+(s^*) - R^+(\widehat{s}_n)\right), \\ &\geqslant p\left(R^+(s^*) - R^+(\widehat{s}_n)\right). \end{aligned}\tag{5.25}$$

Let $\widehat{s}_n$ be a solution of Eq. (5.8) with $\phi(n,\delta'/2)'$, where $\delta' = 2(M+1)\delta/(9M+4)$. Introducing $K_{\delta,M} = (9M+4)/\delta$, we combine Lemma 5.4, Eq. (5.22), Eq. (5.23) and Eq. (5.25), to obtain that: *w.p.* $\geqslant 1-\delta$,

$$\begin{aligned} &p\left(R^+(s^*) - R^+(\widehat{s}_n)\right) \\ &\quad\leqslant \sqrt{\frac{2c\log K_{\delta,M}}{n}}\left([(1-Q_\alpha^*)p(R^+(s^*) - R(\widehat{s}_n))]^{a/2} + [Q_\alpha^*(1-p)\phi(n,\delta'/2)']^{a/2}\right) \\ &\qquad + \frac{C'\log K_{\delta,M}}{n}. \end{aligned}\tag{5.26}$$

The highest order term on the right-hand side is in $O(n^{-1/2}\phi(n,\delta'/2)'^{a/2})$ which is $O(n^{-(2+a)/4})$. Eq. (5.26) is a fixed point equation in $R^+(s^*) - R^+(\widehat{s}_n)$. Finding an upper bound on the solution of this fixed-point equation is done by invoking Lemma 7 of Cucker and Smale (2002), which can be found as Lemma 2.27 in Chapter 2. Applying Lemma 2.27 to Eq. (5.26) concludes the proof. □

Remark 5.6. *We state here the result for the case where $\mathcal{S}$ is of finite cardinality M. Proving this result for more general classes of functions $\mathcal{S}$ can be tackled by the localization argument expressed in Boucheron et al. (2005) (pages 341-346 therein).*

The proof is based on the same argument as that of Clémençon and Vayatis (2010) (Theorem 12), except that it involves a sharp control of the fluctuations of the U-statistic estimates of the true positive rate excess $\mathrm{ROC}_{s*}(\alpha) - R^+(s)$ over the class $\mathcal{S}$. The reduced variance property of U-statistics plays a crucial role in the analysis, which essentially relies on the Hoeffding decomposition (Hoeffding, 1948).

5.3.3 Illustration of the Fast Rates

In this section, we provide numerical evidence of the fast rates of Theorem 5.5, which shows that when Assumption 5.2 is verified, faster rates of generalization can be achieved. Showing the existence of fast rates experimentally requires us to design a problem for which η satisfies Assumption 5.2, which is not trivial due to the pairwise nature of the involved quantities. Such practical evidence of fast rates is rarely found in the literature.

We consider a simple scenario where we focus on pointwise ROC optimization at level $\alpha \in (0,1)$, and we have $\mathcal{X} = [0,1]$, $F = 1$ (X follows an uniform distribution over $[0,1]$), $K = 2$, $p_1 = p_2 = 1/2$ and for any $x \in [0,1/2]$, $F_1(x+1/2) = 2 - F_1(x)$ — *i.e.* F_1 symmetric in the point $(1/2, 1)$ — and:

$$F_1(x) = \begin{cases} 2C & \text{if} \quad x \in [0,m], \\ 1 - |1-2x|^{(1-a)/a} & \text{if} \quad x \in (m, 1/2], \end{cases}$$

where $C \in (0,1/2)$ and $m \in (0,1/2)$ satisfy:

$$C = \frac{1}{2} - \frac{\sqrt{1-2\alpha}}{4m} + \frac{a(1-2m)^{a^{-1}}}{4m}.$$

The complex expression of F_1 arises naturally when trying to design a suitable distribution. The variable m was fixed, and C is a solution of:

$$\int_{1-\eta(x,x')>\frac{1}{2}} (1 - \eta(x,x'))\, dx\, dx' = \frac{\alpha}{2}, \tag{5.27}$$

since we fixed $Q^*_\alpha = 1/2$. Eq. (5.27) is a simple quadratic equation since the posterior probability η expresses in F_1 as:

$$\eta(x,x') = \frac{1}{2} + \frac{1}{2}\left(F_1(x) - 1\right)\left(F_1(x') - 1\right). \tag{5.28}$$

For C to satisfy $0 < C < 1/2$, the variables (m, α, a) need to be restricted, as shown by Fig. 5.1. Precisely, we see that experimental parameters (m, α, a) are valid if their corresponding point is below the orange curve and above the dark blue curve. We see that excessively low values of m restrict severely the possible values of (α, a). The points should be under the red curve if possible, since then F_1 is increasing and assures that $\mathbb{P}\left(|\eta(X,X') - Q^*_\alpha| \leqslant t\right)$ is smooth on a larger neighborhood of 0. Empirical distributions of $|\eta(X,X') - Q^*_\alpha|$ are displayed in Fig. 5.2 for $\alpha = 0.26$, $m = 0.35$. Figure 5.4 shows example distributions of the data.

Finally, we have:

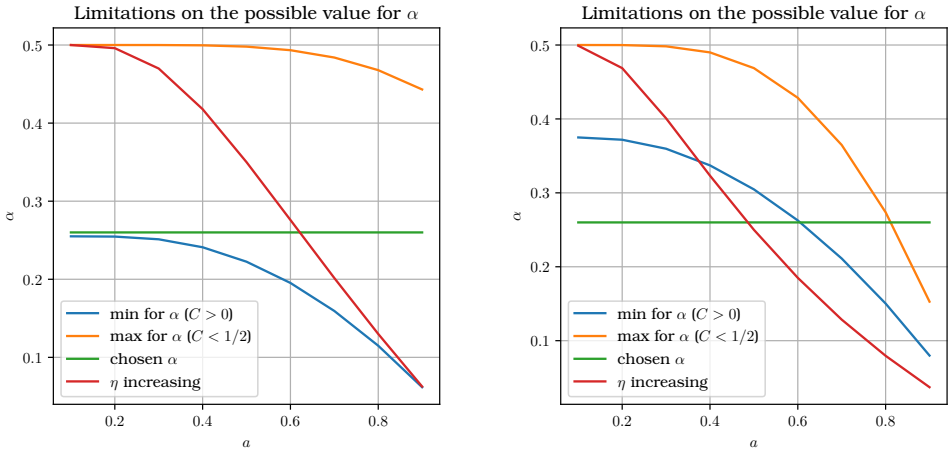
$$\begin{aligned} \mathbb{P}\left(\left|\eta(X,X') - Q^*_\alpha\right| \leqslant t\right) &= \mathbb{P}\left(\left|(F_1(x)-1)(F_1(x')-1)\right| \leqslant 2t\right), \\ &= \int \mathbb{I}\left\{|2x-1| \cdot |2x-1| \leqslant 2(4t)^{\frac{a}{1-a}}\right\}\, dx\, dx', \\ &= 2(4t)^{\frac{a}{1-a}}\left[1 - \log(2) - \frac{a}{1-a}\log(4t)\right]. \end{aligned} \tag{5.29}$$

When t is small enough, Eq. (5.29) gives $\mathbb{P}\left(|\eta(X,X') - Q^*_\alpha| \leqslant t\right)$ of the order $-t^{a/(1-a)}\log(t)$. Ignoring the log term, the result shows that our distribution satisfies the strong Mammen-Tsybakov assumption (Assumption 3.6 of Chapter 3) and not our weak assumption (Assumption 5.2). Due to the logarithm term in the noise condition, we expect the generalization speeds to be slightly worse than $O(n^{-(2+a)/4})$.

Now that the distribution of the data is set, we need to set the class of functions on which we optimize Eq. (5.8). For all $t \in [0,1]$, we define the proposed family $\mathcal{S}$ as

$$\left\{(x,x') \mapsto \mathbb{I}\{(x,x') \in S_t\} \,|\, 0 \leqslant t \leqslant 1\right\}$$

where $S_t = [0,t]^2 \cup [1-t,1]^2$ for any $t \in [0,1]$. The sets S_t are illustrated by Fig. 5.3.



$m = 0.35$ $m = 0.25$

Figure 5.1: Constraints on the value of α for several values of a and two values of m.

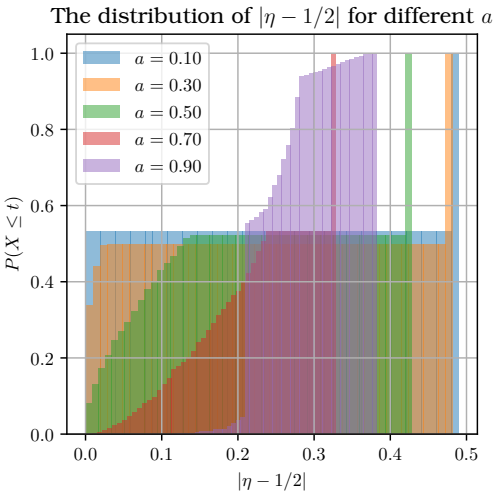


Figure 5.2: Mammen-Tsybakov distributions for values of a when $\alpha = 0.26$, $m = 0.35$.

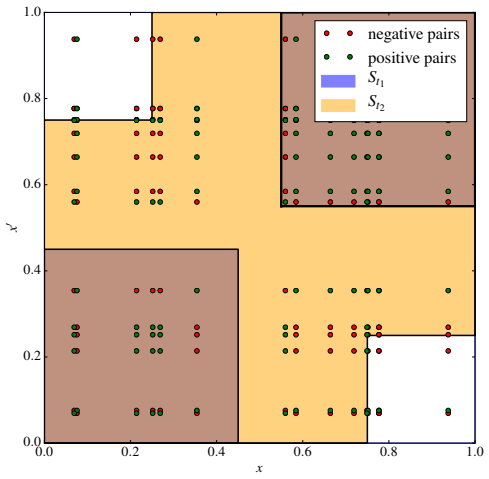


Figure 5.3: Proposal regions S_{t_1}, S_{t_2} for $0 < t_1 < 1/2 < t_2 < 1$. Note that $S_{t_1} \subset S_{t_2}$.

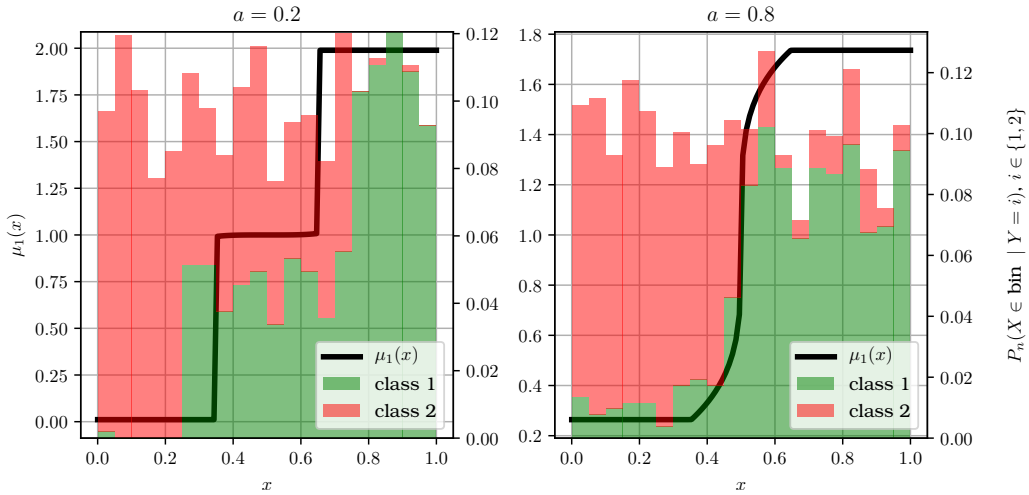


Figure 5.4: Example distributions and F_1's for $n = 1000$ and two values of a.

The risks $R^+(s)$, $R^-(s)$ of an element s of $\mathcal{S}$ that is an indicator of a set S_t can be expressed in closed form with the expression of η. Indeed:

$$R^+(s) = 2\int_{S_t} \eta(x,x')\,dx\,dx' = \lambda(S_t) + \int_{S_t} \left(F_1(x)-1\right)\left(F_1(x')-1\right)\,dx\,dx',$$

using Eq. (5.28), where λ is the usual area measure (*i.e.* Lebesgue measure over $[0,1]^2$). The right-hand side integral is easily developed since it is a sum of integrals over squares included in $[0,1]\times[0,1]$.

We now describe the processus of choosing an optimal empirical function $\widehat{s}_n$ for a set of observations (X_i, Y_i). For all pairs X_i, X_j, we derive the quantity $S_{i,j} = \min\left(\max\left(1-X_i, 1-X_j\right), \max\left(X_i, X_j\right)\right)$. Let $\sigma\left\{1,\ldots,n(n-1)/2\right\} \mapsto \{1,\ldots,n\}^2$ be the function that orders the quantities $S_{i,j}$ increasingly, i.e. $S_{\sigma(1)} \leqslant \cdots \leqslant S_{\sigma(n(n-1)/2)}$. Choosing an optimal empirical function $\widehat{s}_n$ in $\mathcal{S}$ requires to solve the pointwise ROC optimization problem for $(S_{i,j}, Z_{i,j})_{i<j}$ and proposed functions:

$$\left\{ x \mapsto \mathbb{I}\left\{ x \leqslant \frac{S_{\sigma(i)} + S_{\sigma(i+1)}}{2} \right\} \;\middle|\; 0 \leqslant i \leqslant \frac{n(n-1)}{2} \right\},$$

where $S_{\sigma(0)} = 0$ and $S_{\sigma((n(n-1)/2)+1)} = 1$ by convention. It can be solved in $O\left(n^2 \log n\right)$ time and we always neglect the tolerance parameter Φ, *i.e.* we set $\Phi = 0$.

For all $a \in \{1/10,\ldots,9/10\}$, we generate 512 data points and compute the generalization error $\mathrm{ROC}_{S*}(\alpha) - R^+(\widehat{s}_n)$ for the n first data points, where $n \in \{64, 128, 256, 512\}$, and repeat the operation 1000 times. We introduce $Q_{a,n}$ as the 90-quantile of the 1000 realizations of $\mathrm{ROC}_{S*}(\alpha) - R^+(\widehat{s}_n)$ for a given (n,a). The coefficients (C_a, D_a) of the regression $Q_{a,n} = D_a + C_a \times \log(n)$ are estimated. Fig. 5.6 shows the learning speeds C_a given the noise parameters a's. The estimation of the C_a's is illustrated by Fig. 5.5 and Fig. 5.6 summarizes the experiments for the case $\alpha = 0.26$, $m = 0.35$ and $a \in [0.1, 0.9]$. There is a clear downward trend when a increases, illustrating the fast rates in practice.

5.3.4 Solving Pointwise ROC Optimization

Though the formulation of pointwise ROC optimization is natural, this constrained optimization problem is very difficult to solve in practice, as discussed at length in Vogel et al. (2018). As far as we know, the only papers that tackle the problem directly are Scott and Nowak (2006) (Section 7.2), which is based on a fixed partition of the input space, or Scott and Nowak (2005) (Section VI-B), which is based on recursive partitioning of the input space. In Chapter 7, we propose a gradient-descent approach for learning to optimize for a specific point of the ROC curve.

These difficulties suggest the extension to the similarity ranking framework of the TreeRank approach for ROC optimization (Clémençon and Vayatis (2009) and Clémençon et al. (2011)), recalled below. Indeed, in the standard (non pairwise) statistical learning setup for bipartite ranking, whose probabilistic framework is the same as that of binary classification and stipulates that training data are *i.i.d.* labeled observations, this recursive technique builds (piecewise constant) scoring functions s, whose accuracy can be guaranteed in terms of sup norm, *i.e.* for which $D_\infty(s, s^*)$ can be controlled where:

$$D_p(s, s^*) = ||\mathrm{ROC}_s - \mathrm{ROC}^*||_p,$$

with $s^* \in \mathcal{S}^*$ and $p \in [1, +\infty]$. It is the essential purpose of the next section to prove that this remains true when the training observations are of the form $\{((X_i, X_j),\ Z_{i,j}) :\ 1 \leqslant i < j \leqslant n\}$, where $Z_{i,j} = 2\mathbb{I}\{Y_i = Y_j\} - 1$ for $1 \leqslant i < j \leqslant n$, and are thus far from being independent. Regarding the implementation of TreeRank for pairs, attention should be paid to the fact that the splitting rules for recursive partitioning of the space $\mathcal{X} \times \mathcal{X}$ must ensure that the decision functions produced by the algorithm fulfill the symmetric property.

5.4 The TreeRank Algorithm for Learning Similarities

Because they offer a visual model summary in the form of an easily interpretable binary tree graph, decision trees, see *e.g.* Breiman et al. (1984) or Quinlan (1986), remain very popular

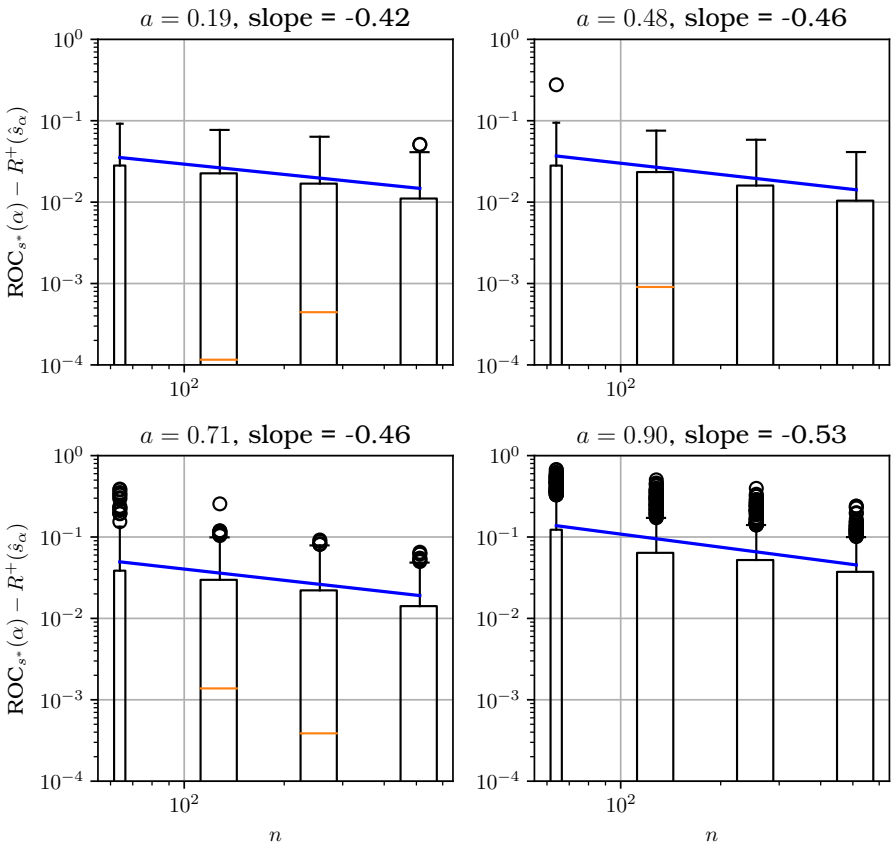


Figure 5.5: Boxplot of the 1000 regrets $\mathrm{ROC}_{S*}(\alpha) - R^+(\widehat{S}_n)$ for each n and several values of a. The line represents the regression on the 0.80-quantile.

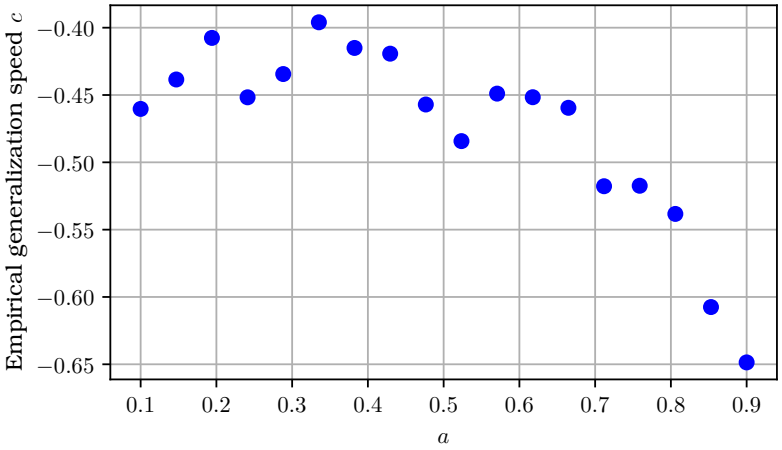


Figure 5.6: Generalization speed for different values of a.

among practitioners. In general, predictions are computed through a hierarchical combination of elementary rules comparing the value taken by a (quantitative) component of the input information (the *split variable*) to a certain threshold (the *split value*). In contrast to (supervised) learning problems such as classification/regression, which are of local nature, predictive rules for a global problem such as *similarity learning* cannot be described by a simple (tree-structured) partition of $\mathcal{X} \times \mathcal{X}$: the (symmetric) cells corresponding to the terminal leaves of the binary decision tree must be sorted in order to define a similarity function.

5.4.1 TreeRank on a Product Space

We define a *similarity tree* as a binary tree whose leaves all correspond to symmetric subsets $\mathcal{C}$ of the product space $\mathcal{X} \times \mathcal{X}$ (*i.e.* $\forall (x, x') \in \mathcal{X}^2$; $(x, x') \in \mathcal{C} \Leftrightarrow (x', x) \in \mathcal{C}$) and is equipped with a 'left-to-right' orientation, that defines a tree-structured collection of similarity functions. Incidentally, the symmetry property makes it a specific *ranking tree*, using the terminology introduced in Clémençon and Vayatis (2009). The root node of a tree $\mathcal{T}_J$ of depth $J \geqslant 0$ corresponds to the whole space $\mathcal{X} \times \mathcal{X}$: $\mathcal{C}_{0,0} = \mathcal{X}^2$, while each internal node (j, k) with $j < J$ and $k \in \{0, \ldots, 2^j - 1\}$ represents a subset $\mathcal{C}_{j,k} \subset \mathcal{X}^2$, whose left and right siblings respectively correspond to (symmetric) disjoint subsets $\mathcal{C}_{j+1,2k}$ and $\mathcal{C}_{j+1,2k+1}$ such that $\mathcal{C}_{j,k} = \mathcal{C}_{j+1,2k} \cup \mathcal{C}_{j+1,2k+1}$. Equipped with the left-to-right orientation, any subtree $\mathcal{T} \subset \mathcal{T}_J$ defines a preorder on $\mathcal{X}^2$: the degree of similarity being the same for all pairs (x, x') lying in the same terminal cell of $\mathcal{T}$. Figure 5.7 represents a fully grown tree of depth 3 with its associated scores.

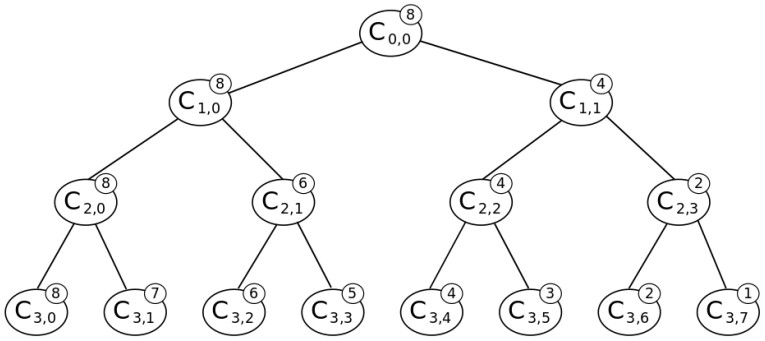


Figure 5.7: A piecewise constant similarity function described by an oriented binary subtree $\mathcal{T}$. For any pair $(x, x') \in \mathcal{X}^2$, the similarity score $s_{\mathcal{T}}(x, x')$ can be computed very fast in a top-down manner using the heap structure: starting from the initial value 2^J at the root node, at each internal node $\mathcal{C}_{j,k}$, the score remains unchanged if (x, x') moves down to the left sibling and one subtracts $2^{J-(j+1)}$ from it if (x, x') moves down to the right.

The similarity function related to the oriented tree $\mathcal{T}$ can be written as:

$$\forall (x, x') \in \mathcal{X}^2, \quad s_{\mathcal{T}}(x, x') = \sum_{\mathcal{C}_{j,k} \text{: terminal leaf of } \mathcal{T}} 2^J \left(1 - \frac{k}{2^j}\right) \cdot \mathbb{I}\{(x, x') \in \mathcal{C}_{j,k}\}. \tag{5.30}$$

Observe that its symmetry results from that of the $\mathcal{C}_{j,k}$'s. The ROC curve of the similarity function $s_{\mathcal{T}}(x, x')$ is the piecewise linear curve connecting the knots:

$$(0, 0) \text{ and } \left(\sum_{l=0}^{k} H(\mathcal{C}_{j,l}),\ \sum_{l=0}^{k} G(\mathcal{C}_{j,l})\right) \text{ for all terminal leaf } \mathcal{C}_{j,k} \text{ of } \mathcal{T},$$

denoting by H and G the conditional distribution of (X, X') given $Z = -1$ and $Z = +1$, respectively. Setting $p = \mathbb{P}\{Z = +1\} = \sum_k p_k^2$, we have $G = (1/p) \sum_k p_k^2 \cdot F_k \otimes F_k$ and $H = (1/(1-p)) \sum_{k \neq l} p_k p_l \cdot F_k \otimes F_l$. A statistical version can be computed by replacing the $H(\mathcal{C}_{j,l})$ or $G(\mathcal{C}_{j,l})$ by their empirical counterpart.

The TreeRank algorithm, a bipartite ranking technique optimizing the ROC curve in a recursive fashion, has been introduced in Clémençon and Vayatis (2009) and its properties have been investigated in Clémençon et al. (2011) at length. Its output consists of a tree-structured scoring rule (5.30) whose ROC curve can be viewed as a piecewise linear approximation of ROC* obtained by a Finite Element Method with implicit scheme and is proved to be nearly optimal in the D_1 sense under mild assumptions. The growing stage is performed as follows. At the root, one starts with a constant similarity function $s_1(x, x') = \mathbb{I}\{(x, x') \in \mathcal{C}_{0,0}\} = 1$ and after $m = 2^j + k$ iterations, $0 \leqslant k < 2^j$, the current similarity function is:

$$s_m(x, x') = \sum_{l=0}^{2k-1} (m - l) \cdot \mathbb{I}\{(x, x') \in \mathcal{C}_{j+1,l}\} + \sum_{l=k}^{2^j - 1} (m - k - l) \cdot \mathbb{I}\{(x, x') \in \mathcal{C}_{j,l}\}$$

and the cell $\mathcal{C}_{j,k}$ is split so as to form a refined version of the similarity function,

$$s_{m+1}(x,x') = \sum_{l=0}^{2k}(m-l)\cdot\mathbb{I}\{(x,x')\in\mathcal{C}_{j+1,l}\} + \sum_{l=k+1}^{2^j-1}(m-k-l)\cdot\mathbb{I}\{(x,x')\in\mathcal{C}_{j,l}\},$$

namely, with maximum (empirical) AUC. Therefore, it happens that this problem boils down to solve a cost-sensitive binary classification problem on the set $\mathcal{C}_{j,k}$, see subsection 3.3 in Clémençon et al. (2011). Indeed, one may write the AUC increment as

$$\mathrm{AUC}(s_{m+1}) - \mathrm{AUC}(s_m) = \frac{1}{2}H(\mathcal{C}_{j,k})G(\mathcal{C}_{j,k}) \times (1-\Lambda(\mathcal{C}_{j+1,2k} \mid \mathcal{C}_{j,k})), \tag{5.31}$$

where $\Lambda(\mathcal{C}_{j+1,2k} \mid \mathcal{C}_{j,k}) := G(\mathcal{C}_{j,k}\backslash\mathcal{C}_{j+1,2k})/G(\mathcal{C}_{j,k}) + H(\mathcal{C}_{j+1,2k})/H(\mathcal{C}_{j,k})$.

Setting $p = G(\mathcal{C}_{j,k})/(H(\mathcal{C}_{j,k}) + G(\mathcal{C}_{j,k}))$, the crucial point of the TreeRank approach is that the quantity $2p(1-p)\Lambda(\mathcal{C}_{j+1,2k} \mid \mathcal{C}_{j,k})$ can be interpreted as the cost-sensitive error of a classifier on $\mathcal{C}_{j,k}$ predicting positive label for any pair lying in $\mathcal{C}_{j+1,2k}$ and negative label fo all pairs in $\mathcal{C}_{j,k}\backslash\mathcal{C}_{j+1,2k}$ with cost p (respectively, $1-p$) assigned to the error consisting in predicting label $+1$ given $Z=-1$ (resp., label -1 given $Z=+1$), balancing thus the two types of error. Hence, at each iteration of the similarity tree growing stage, the TreeRank algorithm calls a *cost-sensitive* binary classification algorithm, termed LeafRank, in order to solve a statistical version of the problem above (replacing the theoretical probabilities involved by their empirical counterparts) and split $\mathcal{C}_{j,k}$ into $\mathcal{C}_{j+1,2k}$ and $\mathcal{C}_{j+1,2k+1}$. As described at length in Clémençon et al. (2011), one may use cost-sensitive versions of celebrated binary classification algorithms such as CART or SVM for instance as LeafRank procedure, the performance depending on their ability to capture the geometry of the level sets R_α^* of the posterior probability $\eta(x,x')$. As highlighted above, in order to apply the TreeRank approach to similarity learning, a crucial feature the LeafRank procedure implemented must have is the capacity to split a region in subsets that are both stable under the reflection $(x,x') \in \mathcal{X}^2 \mapsto (x',x)$. This point is discussed in the next section. Rate bounds for the TreeRank method in the sup norm sense are also established therein in the statistical framework of similarity learning, when the set of training examples $\{((X_i,X_j),\ Z_{i,j})\}_{i<j}$ is composed of non independent observations with binary labels, formed from the original multi-class classification dataset $\mathcal{D}_n$. From a statistical perspective, a learning algorithm can be derived from the recursive approximation procedure recalled in the previous section, simply by replacing the quantities $H(\mathcal{C})$ and $G(\mathcal{C})$, by their respective ($\sigma=-1$ and $\sigma=+1$) empirical counterparts based on the dataset $\mathcal{D}_n$:

$$\widehat{F}_{\sigma,n}(\mathcal{C}) = \frac{1}{n_\sigma}\sum_{i<j}\mathbb{I}\{(X_i,X_j)\in\mathcal{C},\ Z_{i,j}=\sigma 1\}, \tag{5.32}$$

with $n_\sigma = (2/(n(n-1)))\sum_{i<j}\mathbb{I}\{Z_{i,j}=\sigma 1\}$, $\mathcal{C}\subset\mathcal{X}\times\mathcal{X}$ any borelian. Observe incidentally that the quantities (5.32) are by no means *i.i.d.* averages, but take the form of ratios of U-statistics of degree two (*i.e.* averages over pairs of observations, *cf* Lee (1990)), see section 3 in Vogel et al. (2018). For this reason, a specific rate bound analysis (ignoring bias issues) guaranteeing the accuracy of the TreeRank approach in the similarity learning framework is carried out in the following subsections.

The symmetry property of the function (5.30) output by the learning algorithm is directly inherited from that of the candidate subsets $\mathcal{C}\in\mathcal{A}$ of the product space $\mathcal{X}\times\mathcal{X}$ among which the $\mathcal{C}_{d,k}$'s are selected. We new explain at length how to perform the optimization step Eq. (5.31) for similarity ranking ((3.15) for its bipartite ranking version) in practice in the similarity learning context. As recalled above, maximizing (5.31) boils down to finding the best classifier on $\mathcal{C}_{d,k}\subset\mathcal{X}^2$ of the form:

$$g_{\mathcal{C}|\mathcal{C}_{d,k}}(x,x') = \mathbb{I}\{(x,x')\in\mathcal{C}\} - \mathbb{I}\{(x,x')\in\mathcal{C}\backslash\mathcal{C}_{d,k}\} \ \text{ with } \mathcal{C}\subset\mathcal{C}_{d,k},\ \ \mathcal{C}\in\mathcal{A},$$

in the empirical AUC sense, that is to say minimizing a statistical version of the cost-sensitive classification error based on $\{((X_i,X_j),\ Z_{i,j}):\ \ 1\leqslant i<j\leqslant n,\ (X_i,X_j)\in\mathcal{C}_{d,k}\}$:

$$\Lambda(\mathcal{C} \mid \mathcal{C}_{d,k}) = \frac{\mathbb{P}\{g_{\mathcal{C}|\mathcal{C}_{d,k}}(X,X')=1 \mid Z=-1\}}{\mathbb{P}\{(X,X')\in\mathcal{C}_{d,k} \mid Z=-1\}} + \frac{\mathbb{P}\{g_{\mathcal{C}|\mathcal{C}_{d,k}}(X,X')=-1 \mid Z=1\}}{\mathbb{P}\{(X,X')\in\mathcal{C}_{d,k} \mid Z=1\}},$$

which is straightforwardly estimated from data in the same manner as in Chapter 3, see Section 3.2.4.

5.4.2 Learning a Symmetric Function

In Clémençon et al. (2011), it is highlighted that, in the standard ranking bipartite setup, any (cost-sensitive) classification algorithm (*e.g.* Neural Networks, CART, Random Forest, SVM, nearest neighbours) can be possibly used for splitting, whereas, in the present framework, classifiers are defined on product spaces and the symmetry issue must be addressed. For simplicity, assume that $\mathcal{X}$ is a subset of the space $\mathbb{R}^q$, $q \geqslant 1$, whose canonical basis is denoted by $(e_1, \ldots, e_q)$. Denote by $P_V(x, x')$ the orthogonal projection of any point (x, x') in $\mathbb{R}^q \times \mathbb{R}^q$ equipped with its usual Euclidean structure onto the subspace $V = Span((e_1, e_1), \ldots, (e_q, e_q))$. Let W be V's orthogonal complement in $\mathbb{R}^q \times \mathbb{R}^q$.

For any $(x, x') \in \mathcal{X}^2$, write $x = (x_1, \ldots, x_q)$ and $x' = (x'_1, \ldots, x'_q)$. Introduce $f(x, x') := (f_1(x, x'), \ldots, f_{2q}(x, x')) \in \mathbb{R}^{2q}$, which satisfies, for any $j \in \{1, \ldots, q\}$:

$$f_j(x, x') := (x_j + x'_j)/\sqrt{2} \quad \text{and} \quad f_{j+q}(x, x') = \operatorname{sgn}\left(x_{l(x,x')} - x'_{l(x,x')}\right)(x_j - x'_j)/\sqrt{2}, \tag{5.33}$$

where $l(x, x) := \arg\max_{l \in \{1,\ldots,q\}} |x_l - x'_l|$ and $\operatorname{sgn}(\cdot) : \mathbb{R} \to \{-1, 0, +1\}$ is the sign function. Observe that, by construction, $f(x, x') = f(x', x)$ for all $(x, x') \in \mathcal{X}^2$. The first q components of $f(x, x')$ are the coordinates of the projection $P_V(x, x')$ of (x, x') onto the subspace V in an orthonormal basis of V (say $\{(1/\sqrt{2})(e_1, e_1), \ldots, (1/\sqrt{2})(e_q, e_q)\}$ for instance) its last components are formed by a simple data-dependent transformation of the coordinates of the projection $P_W(x, x') = ((x_1 - x'_1)/\sqrt{2}, \ldots, (x_q - x'_q)/\sqrt{2})$ of (x, x') onto W expressed in a given orthonormal basis (say $\{(1/\sqrt{2})(e_1, -e_1), \ldots, (1/\sqrt{2})(e_q, -e_q)\}$ for instance). The following lemma guarantees useful properties of our symmetric transformation of the input space for the derivation of symmetric classifiers.

Lemma 5.7. *Introduce f defined in Eq.* (5.33)*. Let $s : \mathcal{X}^2 \to \mathbb{R}$. Then, s is symmetric if and only if there exists $k : \mathbb{R}^q \times \mathbb{R}^q_+ \to \mathbb{R}$ such that: $\forall (x, x') \in \mathcal{X}^2$, $s(x, x') = (k \circ f)(x, x')$.*

Proof. We first prove the implication. Assume that s is symmetric. Let $(x, x') \in \mathcal{X}^2$. Observe that:

$$x = \frac{\sqrt{2}}{2}\left[P_V(x, x') + P_W(x, x')\right] \quad \text{and} \quad x' = \frac{\sqrt{2}}{2}\left[P_V(x, x') - P_W(x, x')\right].$$

We denote by $f^{(1)}(x, x')$ (resp. $f^{(2)}(x, x')$) the first (resp. last) q components of $f(x, x')$. Denote by $\gamma(x, x') := \operatorname{sgn}(x_{l(x,x)} - x'_{l(x,x')})$, observe that $P_W(x, x') = \gamma(x, x') \cdot f^{(2)}(x, x')$. If $\gamma(x, x') = +1$, we have:

$$s(x, x') = s\left(\frac{\sqrt{2}}{2}\left[f^{(1)}(x, x') + f^{(2)}(x, x')\right], \frac{\sqrt{2}}{2}\left[f^{(1)}(x, x') - f^{(2)}(x, x')\right]\right), \tag{5.34}$$

If $\gamma(x, x') = -1$, the symmetry of s implies that:

$$s(x, x') = s(x', x) = s\left(\frac{\sqrt{2}}{2}\left[f^{(1)}(x, x') + f^{(2)}(x, x')\right], \frac{\sqrt{2}}{2}\left[f^{(1)}(x, x') - f^{(2)}(x, x')\right]\right).$$

Finally, if $\gamma(x, x') = 0$, then $x = x'$ and Eq. (5.34) is also true. We have proven from the symmetry of s implies that we can write $s(x, x')$ as Eq. (5.34), thus that for any $x, x' \in \mathcal{X}^2$, we have $s(x, x') = (k \circ f)(x, x')$.

We now prove the converse. Assume that there exist $k : \mathbb{R}^q \times \mathbb{R}^q_+ \to \mathbb{R}$ such that $\forall (x, x') \in \mathcal{X}^2$, $s(x, x') = (k \circ f)(x, x')$. However for any $(x, x') \in \mathcal{X}^2$, $f(x', x) = f(x, x')$, thus f is symmetric, and s is symmetric. □

In order to get splits that are symmetric *w.r.t.* the reflection $(x, x') \mapsto (x', x)$, we propose to build directly classifiers of the form $(k \circ f)(x, x')$. In practice, this splitting procedure referred to as Symmetric LeafRank and summarized below simply consists in using as input space $\mathbb{R}^q \times \mathbb{R}^q_+$ rather than $\mathbb{R}^{2q}$ and considering as training labeled observations the dataset $\{(f(X_i, X_j), Z_{i,j}) : 1 \leqslant i < j \leqslant n, (X_i, X_j) \in \mathcal{C}_{d,k}\}$ when running a cost-sensitive classification algorithm. Figure 5.8 represents a split produced by the LeafRank procedure.

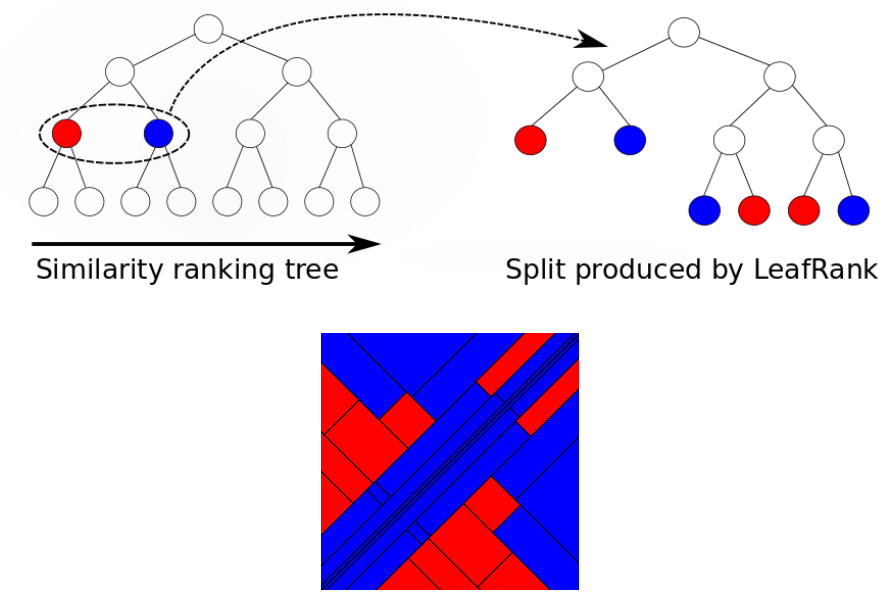


Figure 5.8: Symmetric split produced by the SYMMETRIC LEAFRANK procedure on a bounded two-dimensional space.

Just like in the original version of the TREERANK method, the growing stage can be followed by a pruning procedure, where children of a same parent node are recursively merged in order to produce a similarity subtree that maximizes an estimate of the AUC criterion, based on cross-validation usually, one may refer to section 4 in Clémençon et al. (2011) for further details. In addition, as in the standard bipartite ranking context, the RANKING FOREST approach (Clémençon et al., 2013), an *ensemble learning* technique based on TREERANK that combines aggregation and randomization, can be implemented to dramatically improve stability and accuracy of similarity tree models both at the same time, while preserving their advantages (*e.g.* scalability, interpretability).

SYMMETRIC LEAFRANK

- **Input.** Pairs $\{((X_i, X_j),\ Z_{i,j}) :\ 1 \leqslant i < j \leqslant n,\ (X_i, X_j) \in \mathcal{C}_{d,k}\}$ lying in the (symmetric) region to be split. Classification algorithm $\mathcal{A}$.
- **Cost.** Compute the number of positive pairs lying in the region $\mathcal{C}_{d,k}$

$$p = \frac{\sum_{1 \leqslant i < j \leqslant n} \mathbb{I}\{(X_i, X_j) \in \mathcal{C}_{d,k},\ Z_{i,j} = +1\}}{\sum_{1 \leqslant i < j \leqslant n} \mathbb{I}\{(X_i, X_j) \in \mathcal{C}_{d,k}\}}$$

- **Cost-sensitive classification.** Based on the labeled observations

$$\{(f(X_i, X_j),\ Z_{i,j}) :\ 1 \leqslant i < j \leqslant n,\ (X_i, X_j) \in \mathcal{C}_{d,k}\},$$

  run algorithm $\mathcal{A}$ with cost p for the false positive error and cost $1 - p$ for the false negative error to produce a (symmetric) classifier $g(x, x')$ on $\mathcal{C}_{d,k}$.
- **Output** Define the subregions:

$$\mathcal{C}_{d+1,2k} = \{(x, x') \in \mathcal{C}_{d,k} :\ g(x, x') = +1\} \text{ and } \mathcal{C}_{d+1,2k+1} = \mathcal{C}_{d,k} \backslash \mathcal{C}_{d+1,2k}.$$

5.4.3 Generalization Ability - Rate Bound Analysis

We now prove that the theoretical guarantees formulated in the ROC space equipped with the sup norm that have been established for the TREERANK algorithm in the standard bipartite

ranking setup in Clémençon and Vayatis (2009) remain valid in the similarity learning framework. The rate bound result stated below is the analogue of Corollary 1 in Clémençon and Vayatis (2009). The following technical assumptions are involved:

- the feature space $\mathcal{X}$ is bounded;
- $\alpha \mapsto \mathrm{ROC}^*(\alpha)$ is twice differentiable with a bounded first order derivative;
- the class $\mathcal{A}$ is intersection stable, *i.e.* $\forall (\mathcal{C},\ \mathcal{C}') \in \mathcal{A}^2$, $\mathcal{C} \cap \mathcal{C}' \in \mathcal{A}$;
- the class $\mathcal{A}$ has finite VC dimension $V < +\infty$;
- we have $\{(x, x') \in \mathcal{X}^2 :\ \eta(x, x') \geqslant q\} \in \mathcal{A}$ for any $q \in [0, 1]$.

Theorem 5.8. *Assume that the conditions above are fulfilled. Choose $D = D_n$ so that $D_n \sim \sqrt{\log n}$, as $n \to \infty$, and let s_{D_n} denote the output of the* Similarity TreeRank *algorithm. Then, for all $\delta > 0$, there exists a constant λ s.t., w.p. $\geqslant 1 - \delta$, we have for all $n \geqslant 2$: $D_\infty(s_{D_n}, s^*) \leqslant \exp(-\lambda\sqrt{\log n})$.*

Proof. The proof is based on usual control of U-statistics, as provided in Corollary 4.5 of Chapter 4 and introduced in Clémençon et al. (2016) (Proposition 2). This crucial result permits to control the deviation of the progressive outputs of the Similarity TreeRank algorithm and those of the nonlinear approximation scheme (based on the true quantities) investigated in Clémençon and Vayatis (2009). The proof can be thus derived by following line by line the argument of Corollary 1 in Clémençon and Vayatis (2009). □

This *universal* logarithmic rate bound may appear slow at first glance but attention should be paid to the fact that it directly results from the hierarchical structure of the partition induced by the tree construction and the *global* nature of the similarity learning problem. As pointed out in Clémençon and Vayatis (2009) (see Remark 14 therein), the same rate bound holds true for the deviation in sup norm between the empirical ROC curve $\widehat{\mathrm{ROC}}(s_{D_n}, .)$ output by the TreeRank algorithm and the optimal curve ROC^*.

5.5 Conclusion

We have introduced a rigorous probability framework to study similarity learning from the novel perspective of pairwise bipartite ranking and pointwise ROC optimization. We derived statistical guarantees for generalization that do not depend on the distribution, as well as stronger data-dependent guarantees that do. As far as we know, we have provided the first empirical illustration of fast generalization bounds using simulated data, which confirms our theoretical results. Finally, we presented an extension of the well-known TreeRank algorithm for bipartite ranking and extended its theoretical analysis to similarity ranking, using results on U-statistics.

Our study opens promising directions of future work. We are especially interested in extending our results to allow the rejection of queries from unseen classes (e.g., unknown identities) at test time (see for instance Bendale and Boult, 2015). This could be achieved by incorporating a loss function to encourage the score of all positive pairs to be above some fixed threshold, below which we would reject the query.

The similarity ranking setting presented in this chapter is a direct mathematical translation of the 1:1 biometric identification/verification problem. While our generalization bounds give security guarantees for that problem, they do not imply new practical approaches to 1:1 verification. The next chapters focus on that aspect. Precisely, Chapter 6 focuses on alleviating the scalability issues encountered when dealing with large-scale problems, using approximation methods of U-statistics combined with distributed learning. Additionally, Chapter 7 sketches practical gradient-descent approaches to similarity ranking.

Chapter 6

Distributed U-Statistics

Summary: Pairwise learning problems involves functionals on pairs of observations, as does the specific case of similarity ranking of Chapter 5, which is a direct formalization of the 1:1 biometric verification problem. These problems often involve summing a quadratic number of observations, which implies a prohibitive number of operations for any modern large scale application. Hence, they requires sensible approximations that do not sacrifice statistical accuracy. Most functionals in these problems are U-statistics, a well-studied type of statistic, for which a popular sampling-based approximation is incomplete U-statistics, presented in Section 4.4 of Chapter 4. The extension of those sampling approximations to similarity ranking (Chapter 5) is a direct consequence of Section 4.4, but it assumes that all of the data is accessible for the computation of the U-statistic. However, large-scale learning often requires data-parallelism or learns with small random subsamples of the data. For that reason, this chapter introduces techniques to estimate U-statistics in a distributed environment, and studies their compromise between statistical accuracy and computing time. We prove the benefits brought by our techniques in terms of variance reduction, and extend our results to design distributed gradient descent algorithms for tuplewise empirical risk minimization. Our analysis builds on top of the properties for U-statistics presented in Chapter 4. Our results are supported by numerical experiments in pairwise statistical estimation and learning on synthetic and real-world datasets.

6.1 Introduction

Statistical machine learning has seen dramatic development over the last decades, due partially to the availability of massive datasets, observed in particular in biometrics. For example, the largest public facial recognition dataset now contains 8.2 million images (Guo et al., 2016), and private datasets are much larger. Another contributing factor is the increasing need to perform predictive/inference/optimization tasks in a wide variety of domains. Those two elements have given a considerable boost to the field and led to successful applications. In parallel to those advances, there has been an ongoing technological progress in the architecture of data repositories and distributed systems, allowing to process ever larger (and possibly complex, high-dimensional) data sets gathered on distributed storage platforms.

This trend is illustrated by the development of many easy-to-use cluster computing frameworks for large-scale distributed data processing. These frameworks implement the data-parallel setting, in which data points are partitioned across different machines which operate on their partition in parallel. Some striking examples are Apache Spark Zaharia et al. (2010) and Petuum Xing et al. (2015), the latter being fully targeted to machine learning. The goal of such frameworks is to abstract away the network and communication aspects in order to ease the deployment of distributed algorithms on large computing clusters and on the cloud, at the cost of some restrictions in the types of operations and parallelism that can be efficiently achieved. However, these limitations as well as those arising from network latencies or the nature of certain memory-

intensive operations are often ignored or incorporated in a stylized manner in the mathematical description and analysis of statistical learning algorithms (see *e.g.* Balcan et al. (2012); III et al. (2012); Bellet et al. (2015b); Arjevani and Shamir (2015)). The implementation of statistical methods proved to be theoretically sound may thus be hardly feasible in a practical distributed system, and seemingly minor adjustments to scale-up these procedures can turn out to be disastrous in terms of statistical performance, see e.g. the discussion in Jordan (2013). This greatly restricts their practical interest in some applications and urges the statistics and machine learning communities to get involved with distributed computation more deeply (Bekkerman et al., 2011).

In this chapter, we propose to study these issues in the context of *tuplewise* estimation and learning problems, where the statistical quantities of interest are not basic sample means but come in the form of averages over all pairs (or more generally, d-tuples) of data points. Such data functionals are known as U-statistics (Lee, 1990; de la Pena and Giné, 1999), and many empirical quantities describing global properties of a probability distribution fall in this category (*e.g.* the sample variance, the Gini mean difference, Kendall's tau coefficient). U-statistics are also natural empirical risk measures in several learning problems such as ranking (Clémençon et al., 2008), metric learning (Vogel et al., 2018), cluster analysis (Clémençon, 2014) and risk assessment (Bertail and Tressou, 2006). In the similarity ranking problem of the preceding chapter (Chapter 5), all estimators are combinations of U-statistics. The behavior of these statistics is well-understood and a sound theory for empirical risk minimization based on U-statistics is now documented in the machine learning literature (Clémençon et al., 2008), but the computation of a U-statistic poses a serious scalability challenge as it involves a summation over an exploding number of pairs (or d-tuples) as the dataset grows in size. In the centralized (single machine) setting, this can be addressed by appropriate subsampling methods, which have been shown to achieve a nearly optimal balance between computational cost and statistical accuracy (Clémençon et al., 2016). Unfortunately, naive implementations in the case of a massive distributed dataset either greatly damage the accuracy or are inefficient due to a lot of network communication (or disk I/O). This is due to the fact that, unlike basic sample means, a U-statistic is not separable across the data partitions.

Our main contribution is to design and analyze distributed methods for statistical estimation and learning with U-statistics that guarantee a good trade-off between accuracy and scalability. Our approach incorporates an occasional data repartitioning step between parallel computing stages in order to circumvent the limitations induced by data partitioning over the cluster nodes. The number of repartitioning steps allows to trade-off between statistical accuracy and computational efficiency. To shed light on this phenomenon, we first study the setting of statistical estimation, precisely quantifying the variance of estimates corresponding to several strategies. Thanks to the use of Hoeffding's decomposition (Hoeffding, 1948), our analysis reveals the role played by each component of the variance in the effect of repartitioning. We then discuss the extension of these results to statistical learning and design efficient and scalable stochastic gradient descent algorithms for distributed empirical risk minimization. Finally, we carry out some numerical experiments on pairwise estimation and learning tasks on synthetic and real-world datasets to support our results from an empirical perspective.

The chapter is structured as follows. Section 6.2 reviews background on U-statistics and their use in statistical estimation and learning, and discuss the common practices in distributed data processing. Section 6.3 deals with statistical tuplewise estimation: we introduce our general approach for the distributed setting and derive (non-)asymptotic results describing its accuracy. Section 6.4 extends our approach to statistical tuplewise learning. We provide experiments supporting our results in Section 6.5, and we conclude in Section 6.6.

6.2 Background

In this section, we first review the definition and properties of U-statistics, and discuss some popular applications in statistical estimation and learning. We then discuss the recent randomized methods designed to scale up tuplewise statistical inference to large datasets stored on a single machine. Finally, we describe the main features of cluster computing frameworks.

6.2.1 U-Statistics: Definition and Applications

U-statistics are the natural generalization of *i.i.d.* sample means to tuples of points. We state the definition of U-statistics in their generalized form, where points can come from $K \geqslant 1$ independent samples. Note that we recover classic sample mean statistics in the case where $K = d_1 = 1$.

Definition 6.1. (Generalized U-statistic) *Let $K \geqslant 1$ and $(d_1, \ldots, d_K) \in \mathbb{N}^{*K}$. For each $k \in \{1, \ldots, K\}$, let $\mathbf{X}_{\{1, \ldots, n_k\}} = (X_1^{(k)}, \ldots, X_{n_k}^{(k)})$ be an independent sample of size $n_k \geqslant d_k$ composed of i.i.d. random variables with values in some measurable space $\mathcal{X}_k$ with distribution $F_k(dx)$. Let $h : \mathcal{X}_1^{d_1} \times \cdots \times \mathcal{X}_K^{d_K} \to \mathbb{R}$ be a measurable function, square integrable with respect to the probability distribution $F = F_1^{\otimes d_1} \otimes \cdots \otimes F_K^{\otimes d_K}$. Assume w.l.o.g. that $h(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(K)})$ is symmetric within each block of arguments $\mathbf{x}^{(k)}$ (valued in $\mathcal{X}_k^{d_k}$). The generalized (or K-sample) U-statistic of degrees $(d_1, \ldots, d_K)$ with kernel H is defined as*

$$U_{\mathbf{n}}(h) = \frac{1}{\prod_{k=1}^{K} \binom{n_k}{d_k}} \sum_{I_1} \cdots \sum_{I_K} h(\mathbf{X}_{I_1}^{(1)}, \mathbf{X}_{I_2}^{(2)}, \ldots, \mathbf{X}_{I_K}^{(K)}), \tag{6.1}$$

where $\sum_{I_k}$ denotes the sum over all $\binom{n_k}{d_k}$ subsets $\mathbf{X}_{I_k}^{(k)} = (X_{i_1}^{(k)}, \ldots, X_{i_{d_k}}^{(k)})$ related to a set I_k of d_k indexes $1 \leqslant i_1 < \ldots < i_{d_k} \leqslant n_k$ and $\mathbf{n} = (n_1, \ldots, n_K)$.

The U-statistic $U_{\mathbf{n}}(h)$ is known to have minimum variance among all unbiased estimators of the parameter $m(h) = \mathbb{E}\big[h(X_1^{(1)}, \ldots, X_{d_1}^{(1)}, \ldots, X_1^{(K)}, \ldots, X_{d_K}^{(K)})\big]$. The price to pay for this low variance is a complex dependence structure exhibited by the terms involved in the average (6.1), as each data point appears in multiple tuples. The (non) asymptotic behavior of U-statistics and U-processes (*i.e.*, collections of U-statistics indexed by classes of kernels) can be investigated by means of linearization techniques (Hoeffding, 1948) combined with decoupling methods (de la Pena and Giné, 1999), reducing somehow their analysis to that of basic *i.i.d.* averages or empirical processes. One may refer to Lee (1990) for an account of the asymptotic theory of U-statistics, and to van der Vaart (2000) (Chapter 12 therein) and de la Pena and Giné (1999) for nonasymptotic results.

U-statistics are commonly used as point estimators for inferring certain global properties of a probability distribution as well as in statistical hypothesis testing. Popular examples include the (debiased) *sample variance*, obtained by setting $K = 1$, $d_1 = 2$ and $h(x_1, x_2) = (x_1 - x_2)^2$, the *Gini mean difference*, where $K = 1$, $d_1 = 2$ and $h(x_1, x_2) = |x_1 - x_2|$, and *Kendall's tau rank correlation*, where $K = 2$, $d_1 = d_2 = 1$ and $h((x_1, y_1), (x_2, y_2)) = \mathbb{I}\{(x_1 - x_2) \cdot (y_1 - y_2) > 0\}$.

U-statistics also correspond to empirical risk measures in statistical learning problems such as clustering (Clémençon, 2014), metric learning (Vogel et al., 2018) and multipartite ranking (Clémençon and Robbiano, 2014). The generalization ability of minimizers of such criteria over a class $\mathcal{H}$ of kernels can be derived from probabilistic upper bounds for the maximal deviation of collections of centered U-statistics under appropriate complexity conditions on $\mathcal{H}$ (*e.g.*, finite VC dimension) (Clémençon et al., 2008; Clémençon et al., 2016).

Below, we describe the example of multipartite ranking used in our numerical experiments (Section 6.5). We refer to Clémençon et al. (2016) for details on more learning problems involving U-statistics.

Example 6.2 (Multipartite Ranking)**.** *Consider items described by a random vector of features $X \in \mathcal{X}$ with associated ordinal labels $Y \in \{1, \ldots, K\}$, where $K \geqslant 2$. The goal of multipartite ranking is to learn to rank items in the same preorder as that defined by the labels, based on a training set of labeled examples. Rankings are generally defined through a scoring function $s : \mathcal{X} \to \mathbb{R}$ transporting the natural order on the real line onto $\mathcal{X}$. Given K independent samples, the empirical ranking performance of $s(x)$ is evaluated by means of the empirical* VUS *(Volume Under the* ROC *Surface) criterion (Clémençon and Robbiano, 2014):*

$$\widehat{VUS}(s) = \frac{1}{\prod_{k=1}^{K} n_k} \sum_{i_1=1}^{n_1} \cdots \sum_{i_K=1}^{n_K} \mathbb{I}\{s(X_{i_1}^{(1)}) < \ldots < s(X_{i_K}^{(K)})\}, \tag{6.2}$$

which is a K-sample U-statistic of degree $(1, \ldots, 1)$ with kernel $h_s(x_1, \ldots, x_K) = \mathbb{I}\{s(x_1) < \ldots < s(x_K)\}$.

In Chapter 5, we analyzed the learning rates achieved by minimizing functionals akin to U-statistics. In the large-scale setting, solving this problem is computationally costly due to the very large number of training pairs. In particular, given a sample $\{(X_i, Y_i)\}_{i=1}^n \subset \mathcal{X} \times \{1, \dots, K\}$, averages over the positive pairs, *i.e.* of the same class, and the negatives pairs are sums over respectively $\sum_{k=1}^K n_k(n_k - 1)/2$ and $\sum_{k<l} n_k n_l$ pairs, with $n_k = \sum_{i=1}^n \mathbb{I}\{Y = k\}$ for any $k \in \{1, \dots, K\}$. In the setting where we have a large number of (rather balanced) classes, as in a biometric identification example where a class corresponds to an identity, the number of pairs is quadratic in the number of observations, which makes the approach unpractical. One can extend the theoretical guarantees of Chapter 5 to learning with sampling-based approximations with much lower computational cost, as detailed in Vogel et al. (2018) (Section 4), which is achieved using incomplete U-statistics.

6.2.2 Large-Scale Tuplewise Inference with Incomplete U-statistics

The cost related to the computation of the U-statistic (6.1) rapidly explodes as the sizes of the samples increase. Precisely, the number of terms involved in the summation is $\binom{n_1}{d_1} \times \cdots \times \binom{n_K}{d_K}$, which is of order $O(n^{d_1+\dots+d_K})$ when the n_k's are all asymptotically equivalent. Whereas computing U-statistics based on subsamples of smaller size would severely increase the variance of the estimation, the notion of *incomplete generalized U-statistic* (Blom, 1976) enables to significantly mitigate this computational problem while maintaining a good level of accuracy.

Definition 6.3. (Incomplete generalized U-statistic) *Let $B \geqslant 1$. The incomplete version of the U-statistic* (6.1) *based on B terms is defined by:*

$$\widetilde{U}_B(H) = \frac{1}{B} \sum_{I=(I_1, \dots, I_K) \in \mathcal{D}_B} h(\mathbf{X}_{I_1}^{(1)}, \dots, \mathbf{X}_{I_K}^{(K)}) \tag{6.3}$$

where $\mathcal{D}_B$ is a set of cardinality B built by sampling uniformly with replacement in the set Λ of vectors of tuples $((i_1^{(1)}, \dots, i_{d_1}^{(1)}), \dots, (i_1^{(K)}, \dots, i_{d_K}^{(K)}))$, where $1 \leqslant i_1^{(k)} < \dots < i_{d_k}^{(k)} \leqslant n_k$ and $1 \leqslant k \leqslant K$.

Note incidentally that the subsets of indices can be selected by means of other sampling schemes (Clémençon et al., 2016), but sampling with replacement is often preferred due to its simplicity. In practice, the parameter B should be picked much smaller than the total number of tuples to reduce the computational cost. Like (6.1), the quantity (6.3) is an unbiased estimator of $m(h)$ but its variance is naturally larger:

$$\mathrm{Var}(\widetilde{U}_B(h)) = \Big(1 - \frac{1}{B}\Big)\mathrm{Var}(U_{\mathbf{n}}(h)) + \frac{1}{B}\mathrm{Var}(h(X_1^{(1)}, \dots, X_{d_K}^{(K)})). \tag{6.4}$$

The recent work in Clémençon et al. (2016) has shown that the maximal deviations between (6.1) and (6.3) over a class of kernels $\mathcal{H}$ of controlled complexity decrease at a rate of order $O(1/\sqrt{B})$ as B increases. An important consequence of this result is that sampling $B = O(n)$ terms is sufficient to preserve the learning rate of order $\sqrt{\log n/n}$ of the minimizer of the complete risk (6.1), whose computation requires to average $O(n^{d_1+\dots+d_K})$ terms. In contrast, the distribution of a complete U-statistic built from subsamples of reduced sizes n_k' drawn uniformly at random is quite different from that of an incomplete U-statistic based on $B = \prod_{k=1}^K \binom{n_k'}{d_k}$ terms sampled with replacement in Λ, although they involve the summation of the same number of terms. Empirical minimizers of such a complete U-statistic based on subsamples achieve a much slower learning rate of $O(\sqrt{\log(n)/n^{1/(d_1+\dots+d_K)}})$. We refer to Clémençon et al. (2016) for details and additional results.

We have seen that approximating complete U-statistics by incomplete ones is a theoretically and practically sound approach to tackle large-scale tuplewise estimation and learning problems. However, as we shall see later, the implementation is far from straightforward when data is stored and processed in standard distributed computing frameworks, whose key features are recalled below.

6.2.3 Practices in Distributed Data Processing

Data-parallelism, *i.e.* partitioning the data across different machines which operate in parallel, is a natural approach to store and efficiently process massive datasets. This strategy is especially appealing when the key stages of the computation to be executed can be run in parallel on each partition of the data. As a matter of fact, many estimation and learning problems can be reduced to (a sequence of) local computations on each machine followed by a simple aggregation step. This is the case of gradient descent-based algorithms applied to standard empirical risk minimization problems, as the objective function is nicely separable across individual data points. Optimization algorithms operating in the data-parallel setting have indeed been largely investigated in the machine learning community, see Bekkerman et al. (2011); Boyd et al. (2011); Arjevani and Shamir (2015); Smith et al. (2018) and references therein for some recent work.

Because of the prevalence of data-parallel applications in large-scale machine learning, data analytics and other fields, the past few years have seen a sustained development of distributed data processing frameworks designed to facilitate the implementation and the deployment on computing clusters. Besides the seminal MapReduce framework (Dean and Ghemawat, 2008), which is not suitable for iterative computations on the same data, one can mention Apache Spark (Zaharia et al., 2010), Apache Flink (Carbone et al., 2015) and the machine learning-oriented Petuum (Xing et al., 2015). In these frameworks, the data is typically first read from a distributed file system (such as HDFS, *Hadoop Distributed File System*) and partitioned across the memory of each machine in the form of an appropriate distributed data structure. The user can then easily specify a sequence of distributed computations to be performed on this data structure (map, filter, reduce, etc.) through a simple API which hides low-level distributed primitives (such as message passing between machines). Importantly, these frameworks natively implement fault-tolerance (allowing efficient recovery from node failures) in a way that is also completely transparent to the user.

While such distributed data processing frameworks come with a lot of benefits for the user, they also restrict the type of computations that can be performed efficiently on the data. In the rest of this chapter, we investigate these limitations in the context of tuplewise estimation and learning problems, and propose solutions to achieve a good trade-off between accuracy and scalability.

6.3 Distributed Tuplewise Statistical Estimation

In this section, we focus on the problem of tuplewise statistical estimation in the distributed setting (an extension to statistical learning is presented in Section 6.4). We consider a set of $N \geqslant 1$ workers in a complete network graph (i.e., any pair of workers can exchange messages). For convenience, we assume the presence of a master node, which can be one of the workers and whose role is to aggregate estimates computed by all workers.

For ease of presentation, we restrict our attention to the case of two sample U-statistics of degree $(1,1)$ ($K = 2$ and $d_1 = d_2 = 1$), see Remark 6.7 in Section 6.3.3 for extensions to the general case. We denote by $\mathcal{D}_n = \{X_1, \dots, X_n\}$ the first sample and by $\mathcal{Q}_m = \{Z_1, \dots, Z_m\}$ the second sample (of sizes n and m respectively). These samples are distributed across the N workers. For $i \in \{1, \dots, N\}$, we denote by $\mathcal{R}_i$ the subset of data points held by worker i and, unless otherwise noted, we assume for simplicity that all subsets are of equal size $|\mathcal{R}_i| = (n+m)/N \in \mathbb{N}$. The notations $\mathcal{R}_i^X$ and $\mathcal{R}_i^Z$ respectively denote the subset of data points held by worker i from $\mathcal{D}_n$ and $\mathcal{Q}_m$, with $\mathcal{R}_i^X \cup \mathcal{R}_i^Z = \mathcal{R}_i$. We denote their (possibly random) cardinality by $n_i = |\mathcal{R}_i^X|$ and $m_i = |\mathcal{R}_i^Z|$. Given a kernel h, the goal is to compute a good estimate of the parameter $U(h) = \mathbb{E}[h(X_1, Z_1)]$ while meeting some computational and communication constraints.

6.3.1 Naive Strategies

Before presenting our approach, we start by introducing two simple (but ineffective) strategies to compute an estimate of $U(h)$. The first one is to compute the complete two-sample U-statistic

associated with the full samples $\mathcal{D}_n$ and $\mathcal{Q}_m$:

$$U_{\mathbf{n}}(h) = \frac{1}{nm} \sum_{k=1}^{n} \sum_{l=1}^{m} h(X_k, Z_l), \tag{6.5}$$

with $\mathbf{n} = (n, m)$. While $U_{\mathbf{n}}(h)$ has the lowest variance among all unbiased estimates that can be computed from $(\mathcal{D}_n, \mathcal{Q}_m)$, computing it is a highly undesirable solution in the distributed setting where each worker only has access to a subset of the dataset. Indeed, ensuring that each possible pair is seen by at least one worker would require massive data communication over the network. Note that a similar limitation holds for incomplete versions of (6.5) as defined in Definition 6.3.

A feasible strategy to go around this problem is for each worker to compute the complete U-statistic associated with its local subsample $\mathcal{R}_i$, and to send it to the master node who averages all contributions. This leads to the estimate

$$U_{\mathbf{n},N}(h) = \frac{1}{N} \sum_{i=1}^{N} U_{\mathcal{R}_i}(h) \quad \text{where } U_{\mathcal{R}_i}(h) = \frac{1}{n_i m_i} \sum_{k \in \mathcal{R}_i^X} \sum_{l \in \mathcal{R}_i^Z} h(X_k, Z_l). \tag{6.6}$$

Note that if $\min(n_i, m_i) = 0$, we simply set $U_{\mathcal{R}_i}(h) = 0$.

Alternatively, as the $\mathcal{R}_i$'s may be large, each worker can compute an incomplete U-statistic $\widetilde{U}_{B,\mathcal{R}_i}(h)$ with B terms instead of $U_{\mathcal{R}_i}$, leading to the estimate

$$\widetilde{U}_{\mathbf{n},N,B}(h) = \frac{1}{N} \sum_{i=1}^{N} \widetilde{U}_{B,\mathcal{R}_i}(h) \quad \text{where } \widetilde{U}_{B,\mathcal{R}_i}(h) = \frac{1}{B} \sum_{(k,l) \in \mathcal{R}_{i,B}} h(X_k, Z_l), \tag{6.7}$$

with $\mathcal{R}_{i,B}$ a set of B pairs built by sampling uniformly with replacement from the local subsample $\mathcal{R}_i^X \times \mathcal{R}_i^Z$.

As shown in Section 6.3.3, strategies (6.6) and (6.7) have the undesirable property that their accuracy decreases as the number of workers N increases. This motivates our proposed approach, introduced in the following section.

6.3.2 Proposed Approach

The naive strategies presented above are either accurate but very expensive (requiring a lot of communication across the network), or scalable but potentially inaccurate. The approach we promote here is of disarming simplicity and aims at finding a sweet spot between these two extremes. The idea is based on *repartitioning* the dataset a few times across workers (we keep the repartitioning scheme abstract for now and postpone the discussion of concrete choices to subsequent sections). By alternating between parallel computation and repartitioning steps, one considers several estimates based on the same data points. This allows to observe a greater diversity of pairs and thereby refine the quality of our final estimate, at the cost of some additional communication.

Formally, let T be the number of repartitioning steps. We denote by $\mathcal{R}_i^t$ the subsample of worker i after the t-th repartitioning step, and by $U_{\mathcal{R}_i^t}(h)$ the complete U-statistic associated with $\mathcal{R}_i^t$. At each step $t \in \{1, \dots, T\}$, each worker i computes $U_{\mathcal{R}_i^t}(h)$ and sends it to the master node. After T steps, the master node has access to the following estimate:

$$\widehat{U}_{\mathbf{n},N,T}(h) = \frac{1}{T} \sum_{t=1}^{T} U_{\mathbf{n},N}^t(h), \tag{6.8}$$

where $U_{\mathbf{n},N}^t(h) = (1/N) \sum_{i=1}^{N} U_{\mathcal{R}_i^t}(h)$. Similarly as before, workers may alternatively compute incomplete U-statistics $\widetilde{U}_{B,\mathcal{R}_i^t}(h)$ with B terms. The estimate is then:

$$\widetilde{U}_{\mathbf{n},N,B,T}(h) = \frac{1}{T} \sum_{t=1}^{T} \widetilde{U}_{\mathbf{n},N,B}^t(h), \tag{6.9}$$

where $\widetilde{U}_{\mathbf{n},N,B}^t(h) = (1/N) \sum_{i=1}^{N} \widetilde{U}_{B,\mathcal{R}_i^t}(h)$. These statistics, and those introduced in Section 6.3.1 which do not rely on repartition, are summarized in Figure 6.1. Of course, the repartitioning operation is rather costly in terms of runtime so T should be kept to a reasonably small value. We illustrate this trade-off by the analysis presented in the next section.

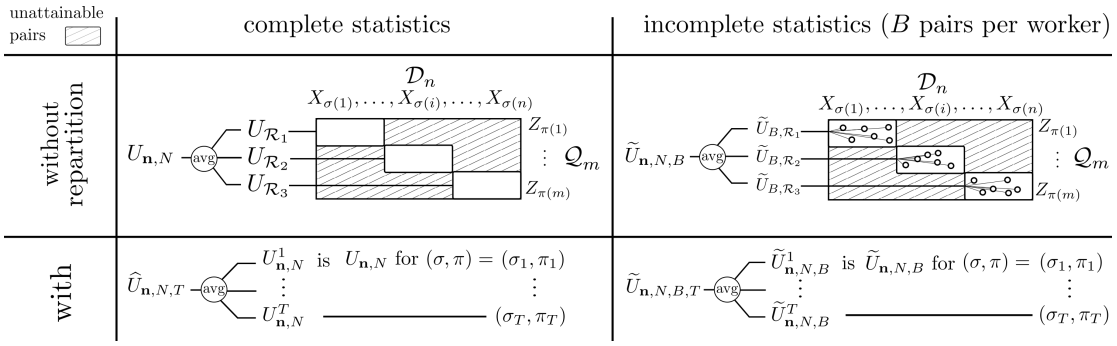


Figure 6.1: Graphical summary of the statistics that we compare: with/without repartition and with/without subsampling. Note that $\{(\sigma_t, \pi_t)\}_{t=1}^T$ denotes a set of T independent couples of random permutations in $\mathfrak{S}_n \times \mathfrak{S}_m$.

6.3.3 Analysis

In this section, we analyze the statistical properties of the various estimators introduced above. We focus here on repartitioning by *proportional sampling without replacement* (prop-SWOR). Prop-SWOR creates partitions that contain the same proportion of elements of each sample: specifically, it ensures that at any step t and for any worker i, $|\mathcal{R}_i^t| = (n+m)/N$ with $|\mathcal{R}_i^{t,X}| = n/N$ and $|\mathcal{R}_i^{t,Z}| = m/N$. We discuss the practical implementation of this repartitioning scheme as well as some alternative choices in Section 6.3.5 and Section 6.3.4.

All estimators are unbiased when repartitioning is done with prop-SWOR. We will thus compare their variance. Our main technical tool is a linearization technique for U-statistics known as Hoeffding's Decomposition (see Hoeffding (1948); Clémençon et al. (2008); Clémençon et al. (2016)).

Definition 6.4. (Hoeffding's decomposition) *Let $h_1(x) = \mathbb{E}[h(x, Z_1)]$, $h_2(z) = \mathbb{E}[h(X_1, z)]$ and $h_0(x, z) = h(x, z) - h_1(x) - h_2(z) + U(h)$. $U_{\mathbf{n}}(h) - U(h)$ can be written as a sum of three orthogonal terms:*

$$U_{\mathbf{n}}(h) - U(h) = T_n(h) + T_m(h) + W_{\mathbf{n}}(h),$$

where $T_n(h) = (1/n)\sum_{k=1}^n h_1(X_k) - U(h)$ and $T_m(h) = (1/m)\sum_{l=1}^n h_2(Z_l) - U(h)$ are sums of independent r.v.'s while $W_{\mathbf{n}}(h) = (nm)^{-1}\sum_{k=1}^n\sum_{l=1}^m h_0(X_k, Z_l)$ is a degenerate U-statistic (i.e. $\mathbb{E}[h(X_1, Z_1)|X_1] = U(h)$ and $\mathbb{E}[h(X_1, Z_1)|Z_1] = U(h)$).

This decomposition is very convenient as the two terms $T_n(h)$ and $T_m(h)$ are decorrelated and the analysis of $W_{\mathbf{n}}(h)$ (a degenerate U-statistic) is well documented (Hoeffding, 1948; Clémençon et al., 2008; Clémençon et al., 2016). It will allow us to decompose the variance of the estimators of interest into single-sample components $\sigma_1^2 = \mathrm{Var}(h_1(X))$ and $\sigma_2^2 = \mathrm{Var}(h_2(Z))$ on the one hand, and a pairwise component $\sigma_0^2 = \mathrm{Var}(h_0(X_1, Z_1))$ on the other hand. Denoting $\sigma^2 = \mathrm{Var}(h(X_1, Z_1))$, we have $\sigma^2 = \sigma_0^2 + \sigma_1^2 + \sigma_2^2$.

It is well-known that the variance of the complete U-statistic $U_{\mathbf{n}}(h)$ can be written as

$$\mathrm{Var}(U_{\mathbf{n}}(h)) = \frac{\sigma_1^2}{n} + \frac{\sigma_2^2}{m} + \frac{\sigma_0^2}{nm}.$$

Our first result gives the variance of the estimators which do not rely on a repartitioning of the data with respect to the variance of $U_{\mathbf{n}}(h)$.

Theorem 6.5. *If the data is distributed over workers using prop-SWOR, we have:*

$$\begin{aligned} Var(U_{\mathbf{n},N}(h)) &= Var(U_{\mathbf{n}}(h)) + (N-1)\frac{\sigma_0^2}{nm}, \\ Var(\widetilde{U}_{\mathbf{n},N,B}(h)) &= \left(1 - \frac{1}{B}\right) Var(U_{\mathbf{n},N}(h)) + \frac{\sigma^2}{NB}. \end{aligned}$$

Proof. First, consider $\mathrm{Var}(U_{\mathbf{n},N})$. Hoeffding's decomposition implies that:

$$U_{\mathbf{n},N}(h) - U(h) = T_n(h) + T_m(h) + \frac{1}{N}\sum_{k=1}^{N}\frac{1}{n_0 m_0}\sum_{i\in\mathcal{R}_k^{\mathcal{X}}}\sum_{j\in\mathcal{R}_k^{\mathcal{Z}}} h_0(X_i, Z_j),$$

as well as the following properties, for any $\forall((k,l),(i,j)) \in (\{1,\dots,n\}\times\{1,\dots,m\})^2$,

$$\begin{aligned} \mathrm{Cov}(h_1(X_k), h_2(Z_l)) &= 0, \\ \mathrm{Cov}(h_1(X_j), h_0(X_k, Z_l)) = 0 \quad &\text{and} \quad \mathrm{Cov}(h_2(Z_j), h_0(X_k, Z_l)) = 0, \\ \mathrm{Cov}(h_0(X_i, Z_j), h_0(X_k, Z_l)) = 0 \quad &\text{if} \quad (i,j) \neq (k,l), \end{aligned} \tag{6.10}$$

which imply the result. The variance of the complete U-statistic $U_{\mathbf{n}}$ is just the special case $N = 1$ of the variance $U_{\mathbf{n},N}$. Explicitely,

$$\mathrm{Var}(U_{n,N}(h)) = \frac{\sigma_1^2}{n} + \frac{\sigma_2^2}{m} + \frac{N\sigma_0^2}{nm}.$$

Now for $\widetilde{U}_{\mathbf{n},N,B}(h)$, since $\widetilde{U}_{\mathbf{n},N,B}$ conditioned upon the data has expectation $U_{\mathbf{n},N}(h)$, i.e.

$$\mathbb{E}\left[\widetilde{U}_{\mathbf{n},N,B}(h) | \mathcal{D}_n, \mathcal{Q}_m, (\mathcal{R}_k)_{k=1}^N\right] = U_{\mathbf{n},N}(h),$$

the law of total variance implies,

$$\begin{aligned} \mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B}(h)) &= \mathrm{Var}(U_{\mathbf{n},N}(h)) + \mathbb{E}[\mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B}(h) | \mathcal{D}_n, \mathcal{Q}_m, (\mathcal{R}_k)_{k=1}^N)], \\ &= \mathrm{Var}(U_{\mathbf{n},N}(h)) + \frac{1}{N}\mathbb{E}[\mathrm{Var}(\widetilde{U}_{\mathcal{R}_1,B}(h) | \mathcal{D}_n, \mathcal{Q}_m, (\mathcal{R}_k)_{k=1}^N)], \\ &\text{(Since the draws of } B \text{ pairs on different workers are independent)}, \\ &= \mathrm{Var}(U_{\mathbf{n},N}(h)) + \frac{1}{N}\left[-\frac{1}{B}\mathrm{Var}(U_{\mathcal{R}_1}) + \frac{1}{B}\mathrm{Var}(h(X,Z))\right], \\ &\text{(See Clémençon et al. (2016))} \\ &= \left(1 - \frac{1}{B}\right)\mathrm{Var}(U_{\mathbf{n},N}(h)) + \frac{1}{NB}\mathrm{Var}(h(X,Z)), \end{aligned}$$

which concludes our proof. Explicitly,

$$\mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B}(h)) = \left(1 - \frac{1}{B}\right)\left(\frac{\sigma_1^2}{n} + \frac{\sigma_2^2}{m} + \frac{N\sigma_0^2}{nm}\right) + \frac{1}{NB}\mathrm{Var}\left(h(X,Z)\right).$$

□

Theorem 6.5 precisely quantifies the excess variance due to the distributed setting if one does not use repartitioning. Two important observations are in order. First, the variance increase is proportional to the number of workers N, which clearly defeats the purpose of distributed processing. Second, this increase only depends on the pairwise component σ_0^2 of the variance. In other words, the average of U-statistics computed independently over the local partitions contains all the information useful to estimate the single-sample contributions, but fails to accurately estimate the pairwise contributions. The resulting estimates thus lead to significantly larger variance when the choice of kernel and the data distributions imply that σ_0^2 is large compared to σ_2^1 and/or σ_1^2. The extreme case happens when $U_{\mathbf{n}}(h)$ is a degenerate U-statistic, *i.e.* $\sigma_1^2 = \sigma_2^2 = 0$ and $\sigma_0^2 > 0$, which is verified for example when $h(x,z) = x \cdot z$ and X, Z are both centered random variables.

We now characterize the variance of the estimators that leverage data repartitioning steps.

Theorem 6.6. *If the data is distributed and repartitioned between workers using prop-SWOR, we have:*

$$\begin{aligned} Var(\widehat{U}_{\mathbf{n},N,T}(h)) &= Var(U_{\mathbf{n}}(h)) + (N-1)\frac{\sigma_0^2}{nmT}, \\ Var(\widetilde{U}_{\mathbf{n},N,B,T}(h)) &= Var(\widehat{U}_{\mathbf{n},N,T}(h)) - \frac{1}{TB}Var(U_{\mathbf{n},N}(h)) + \frac{\sigma^2}{NTB}. \end{aligned}$$

Proof. We first detail the derivation of $\mathrm{Var}(\widehat{U}_{n,N,T}(h))$. Define the Bernouilli *r.v.* $\epsilon_i^t(k)$ as equal to one if X_k is in partition i at time t, and similarly $\gamma_i^t(l)$ is equal to one if Z_l is in partition i at time t. Note that for $t \neq t_1$, $\epsilon_i^t(k)$ and $\epsilon_{i_1}^{t_1}(k_1)$ are independent, as well as $\gamma_i^t(l)$ and $\gamma_{i_1}^{t_1}(l_1)$. Additionally, $\epsilon_i^t(k)$ and $\gamma_{i_1}^{t_1}(l)$ are independent for any $t, t_0 \in \{1, \dots, T\}^2$.

Hoeffding's decomposition implies:

$$U_{\mathbf{n},N}^t(h) - U_n(h) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{nm}\sum_{k=1}^{n}\sum_{l=1}^{m}(N^2\epsilon_i^t(k)\gamma_i^t(l) - 1)h_0(X_k, Z_l).$$

The law of total variance, the fact that conditioned upon the data $\widehat{U}_{\mathbf{n},N,T}(h)$ is an average of T independent experiments and the properties of Eq. (6.10) imply:

$$\begin{aligned}
\mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) &= \mathrm{Var}\left(U_n(h)\right) + \mathbb{E}\left[\mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)|\mathcal{D}_n, \mathcal{Q}_m\right)\right], \\
&= \mathrm{Var}\left(U_n(h)\right) + \frac{1}{T}\mathbb{E}\left[\mathrm{Var}\left(U_{\mathbf{n},N}^t(h)|\mathcal{D}_n, \mathcal{Q}_m\right)\right], \\
&= \mathrm{Var}\left(U_n(h)\right) + \frac{N^2\sigma_0^2}{nmT}\sum_{i_1,i_2=1}^{N}\mathrm{Cov}\left(\epsilon_{i_1}^t(1)\gamma_{i_1}^t(1), \epsilon_{i_2}^t(1)\gamma_{i_2}^t(1)\right). \qquad (6.11)
\end{aligned}$$

On the other hand, observe that:

$$\mathrm{Cov}\left(\epsilon_{i_1}^t(1)\gamma_{i_1}^t(1), \epsilon_{i_2}^t(1)\gamma_{i_2}^t(1)\right) = \begin{cases} -N^{-4} & \text{if } i_1 \neq i_2, \\ N^{-2} - N^{-4} & \text{if } i_1 = i_2. \end{cases} \qquad (6.12)$$

The result is obtained by plugging Eq. (6.12) in Eq. (6.11). Explicitly,

$$\mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) = \mathrm{Var}\left(U_n(h)\right) + \frac{N-1}{nmT}\sigma_0^2.$$

Using that $\mathbb{E}[\widetilde{U}_{\mathbf{n},N,B,T}(h)|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma] = \widehat{U}_{\mathbf{n},N,T}(h)$, we now compute $\mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B,T}(h))$ by decomposing it as the variance of its conditional expectation plus the expectation of its conditional variance. It writes:

$$\begin{aligned}
\mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B,T}(h)) &= \mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) + \mathbb{E}\left[\mathrm{Var}\left(\widetilde{U}_{\mathbf{n},N,B,T}|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right)\right] \\
&= \mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) + \frac{1}{NT}\mathbb{E}\left[\mathrm{Var}\left(\widetilde{U}_{B,\mathcal{R}_i^t}|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right)\right] \\
&\quad \text{(Since the draws of } B \text{ pairs on different workers are independent)} \\
&= \mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) + \frac{1}{NT}\left[-\frac{1}{B}\mathrm{Var}\left(U_{\mathcal{R}_i^t}\right) + \frac{1}{B}\mathrm{Var}(h(X,Z))\right] \\
&\quad \text{(See Clémençon et al. (2016).)} \\
&= \frac{\mathrm{Var}(h(X,Z))}{NTB} + \left(1 - \frac{1}{TB}\right)\left(\frac{\sigma_1^2}{n} + \frac{\sigma_2^2}{m}\right) + \frac{\sigma_0^2}{nm}\left[1 + \frac{N-1}{T} - \frac{N}{TB}\right],
\end{aligned}$$

which gives the desired result after reorganizing the terms.

□

See Fig. 6.2 for illustrations of these formulas with two example cases. The figure represent the variance of each estimator as a function of the number of evaluated pairs for proportional SWOR.

Theorem 6.5 and Theorem 6.6 show that the influence of the kernel h and the distribution of X and Z on the expression of the variances is summarized by the variance derived from pairwise information σ_0^2, that issued from instances of the majority class σ_1^2 and that from instances of the minority class σ_2^2. The value of repartitioning arises from the fact that the term that account for pairwise variance for $\widehat{U}_{\mathbf{n},N,T}(h)$ is almost T times lower that than of $U_{\mathbf{n}}(h)$. Repartitioning is thus interesting when the pairwise variance term is significant in front of the other terms. Practical settings are presented in Section 6.5.

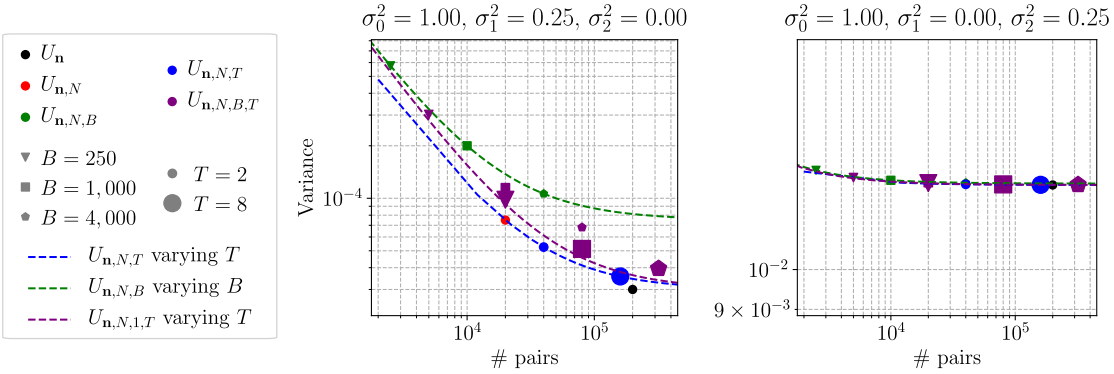


Figure 6.2: Theoretical variance as a function of the number of evaluated pairs for different estimators under prop-SWOR, with $n = 100,000$, $m = 200$ and $N = 100$.

Since $n \gg m$, it suffices that σ_2^2 be low for the influence of pairwise variance to be significant, which is illustrated in Fig. 6.2. It shows that for a same number of pairs evaluated, one can only expect a very limited precision for the estimators that do not use redistribution. The left-hand side figure in Fig. 6.2 shows the importance of redistribution in some cases. Indeed, it shows that for a same number of pairs, one can only expect a very limited precision for the estimators $\mathrm{Var}(U_{\mathbf{n},N}(h))$ and $\mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B}(h))$, compared to those presented in Theorem 6.6 that use redistribution. When σ_2^2 is not small, the variance of $U_{\mathbf{n}}$ mostly originates from the rarity of the minority class which implies that redistributing does not bring estimates that are significantly more accurate.

On the other hand, the right-hand side figure in Fig. 6.2 shows that this only holds for certain types of statistics, and that for different parameters $\sigma_0^2, \sigma_1^2, \sigma_2^2$, redistribution of the instances on the workers does not lead to more accurate estimators.

Finally, the results of Theorem 6.6 imply that, in the case of proportional SWOR and for a fixed budget of evaluated pairs, using all pairs on each worker is always a dominant strategy over using incomplete estimators, *i.e.* $\widehat{U}_{\mathbf{n},N,T}$ is always preferable to $\widetilde{U}_{\mathbf{n},N,B,T}$. This is shown by developments of the difference $\mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B,T}(h)) - \mathrm{Var}(\widehat{U}_{\mathbf{n},N,T_0}(h))$, while imposing $NBT = nmT_0/N$. Indeed,

$$
\begin{aligned}
\Delta :=& \mathrm{Var}\left(\widetilde{U}_{\mathbf{n},N,B,T}(h)\right) - \mathrm{Var}\left(\widehat{U}_{n,N,T_0}\right), \\
=& \sigma_0^2 \left[\frac{N-1}{nm}\left(\frac{1}{T} - \frac{1}{T_0}\right) - \frac{N}{nmTB} + \frac{1}{NTB}\right] + \frac{\sigma_1^2}{TB}\left[\frac{1}{N} - \frac{1}{n}\right] + \frac{\sigma_2^2}{TB}\left[\frac{1}{N} - \frac{1}{m}\right], \\
=& \sigma_0^2 \left[\frac{N-1}{nmT}\left(1 - \frac{1}{B}\right) + \frac{1}{TB}\left(\frac{1}{N^2} - \frac{1}{nm}\right)\right] + \frac{\sigma_1^2}{TB}\left[\frac{1}{N} - \frac{1}{n}\right] + \frac{\sigma_2^2}{TB}\left[\frac{1}{N} - \frac{1}{m}\right],
\end{aligned}
$$

which shows that $\Delta > 0$. Note that computing complete U-statistics also require fewer repartitioning steps to evaluate the same number of pairs (i.e., $T_0 \leqslant T$).

Optimization processes, such as stochastic gradient descent are often interested in small random batches of data over many timesteps, and the dominance of $\widehat{U}_{\mathbf{n},N,T}$ over $\widetilde{U}_{\mathbf{n},N,B,T}$ supports the hypothesis that redistributing the instances infrequently between timesteps is enough to correct for the losses in precision incurred from redistributing the data.

Remark 6.7 (Extension to high-order U-statistics)**.** *The extension of our analysis to general U-statistics is straightforward (see Clémençon et al. (2016) for a review of the relevant technical tools). We stress the fact that the benefits of repartitioning are even stronger for higher-order U-statistics ($K > 2$ and/or larger degrees) because higher-order components of the variance are also affected.*

6.3.4 Extension to Sampling With Replacement

While the use of prop-SWR is not very natural in a standard distributed setting, it is relevant in cases where workers have access to joint database that they can efficiently subsample. We have the following results for the variance of estimates based on prop-SWR.

Theorem 6.8. *If the data is distributed between workers with prop-SWR, and denoting* $\mathbf{n}_0 = (n/N, m/N)$*, we have:*

$$\begin{aligned}
Var(U_{\mathbf{n},1}(h)) &= \frac{\sigma_1^2}{n}\left(2-\frac{1}{n}\right) + \frac{\sigma_2^2}{m}\left(2-\frac{1}{m}\right) + \frac{\sigma_0^2}{nm}\left[4-2\left(\frac{1}{n}+\frac{1}{m}\right)+\frac{1}{nm}\right],\\
Var(U_{\mathbf{n},N}(h)) &= Var(U_{\mathbf{n},1}(h)) + \frac{\sigma_0^2}{nm}(N-1)\left(1-\frac{1}{n}\right)\left(1-\frac{1}{m}\right),\\
Var(\widetilde{U}_{\mathbf{n},N,B}(h)) &= Var(U_{\boldsymbol{n},N}(h)) + \frac{1}{NB}\left[\sigma^2 - Var(U_{\mathbf{n}_0,1}(h))\right].
\end{aligned}$$

Proof. First we derive the variance of $U_{\mathbf{n},N}(h)$. Since $\mathbb{E}[U_{\mathbf{n},N}(h)|\mathcal{D}_n, \mathcal{Q}_m] = U_{\mathbf{n}}(h)$, the law of total variance implies:

$$\begin{aligned}
\mathrm{Var}(U_{\mathbf{n},N}(h)) &= \mathrm{Var}(U_{\mathbf{n}}(h)) + \mathbb{E}\left[\mathrm{Var}\left(U_{\mathbf{n},N}(h)|\mathcal{D}_n, \mathcal{Q}_m\right)\right],\\
&= \mathrm{Var}(U_{\mathbf{n}}(h)) + \frac{1}{N}\mathbb{E}\left[\mathrm{Var}\left(U_{\mathcal{R}_1}(h)|\mathcal{D}_n, \mathcal{Q}_m\right)\right].
\end{aligned}$$

Introduce $\epsilon(k)$ (resp. $\gamma(l)$) as the random variable that is equal to the number of times that k has been sampled in cluster 1 for the $\mathcal{D}_n$ elements (resp. that l has been sampled in cluster 1 for the $\mathcal{Q}_m$ elements). The random variable $\epsilon(k)$ (resp. $\gamma(l)$) follows a binomial distribution with parameters $(n/N, 1/n)$ (resp. $(m/N, 1/m)$). Note that the ϵ and γ are independent and that $\sum_{k=1}^n \epsilon(k) = n/N$ and $\sum_{l=1}^m \gamma(l) = m/N$. It follows that:

$$\begin{aligned}
U_{\mathcal{R}_1}(h) - U_n(h) = U(h) &+ \frac{1}{n}\sum_{k=1}^{n}\left(N\epsilon(k)-1\right)\left(h_1(X_k)-U(h)\right)\\
&+ \frac{1}{m}\sum_{l=1}^{m}\left(N\gamma(l)-1\right)\left(h_2(Z_l)-U(h)\right)\\
&+ \frac{1}{nm}\sum_{k=1}^{n}\sum_{l=1}^{m}\left(N^2\epsilon(k)\gamma(l)-1\right)h_0(X_k,Z_l),
\end{aligned}$$

which implies, using the results of Eq. (6.10),

$$\mathbb{E}\left[\mathrm{Var}\left(U_{\mathcal{R}_1}(h)|\mathcal{D}_n, \mathcal{Q}_m\right)\right] = \frac{N^2\sigma_1^2}{n}\mathrm{Var}(\epsilon(1)) + \frac{N^2\sigma_2^2}{m}\mathrm{Var}(\gamma(1)) + \frac{N^4\sigma_0^2}{nm}\mathrm{Var}(\epsilon(1)\gamma(1)). \quad (6.13)$$

The mean and variance of a binomial distribution is known. Since $\epsilon(1)$ and $\gamma(1)$ are independent,

$$\begin{aligned}
&\mathrm{Var}(\epsilon(1)) = \frac{1}{N}\left(1-\frac{1}{n}\right), \quad \mathrm{Var}(\gamma(1)) = \frac{1}{N}\left(1-\frac{1}{m}\right),\\
&\mathrm{Var}(\epsilon(1)\gamma(1)) = \frac{1}{N^2}\left[\left(1-\frac{1}{n}\right)\left(1-\frac{1}{m}\right) + \frac{1}{N}\left(2-\frac{1}{n}-\frac{1}{m}\right)\right].
\end{aligned} \quad (6.14)$$

Plugging Eq. (6.14) into Eq. (6.13) gives the result. Explicitly,

$$\begin{aligned}
\mathrm{Var}\left(U_{\mathbf{n},N}(h)\right) =& \frac{\sigma_1^2}{n}\left(2-\frac{1}{n}\right) + \frac{\sigma_2^2}{m}\left(2-\frac{1}{m}\right)\\
&+ \frac{\sigma_0^2}{nm}\left[\left(3-\frac{1}{n}-\frac{1}{m}\right) + N\left(1-\frac{1}{n}\right)\left(1-\frac{1}{m}\right)\right].
\end{aligned}$$

Now we derive the variance of $\widetilde{U}_{\mathbf{n},N,B}(h)$. Note that $\mathbb{E}[\widetilde{U}_{\mathbf{n},N,B}|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma] = U_{\mathbf{n},N}(h)$, hence:

$$\begin{aligned}
\mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B}) &= \mathrm{Var}(U_{\mathbf{n},N}(h)) + \mathbb{E}\left[\mathrm{Var}\left(\widetilde{U}_{\mathbf{n},N,B}|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right)\right],\\
&= \mathrm{Var}(U_{\mathbf{n},N}(h)) + \frac{1}{N}\mathbb{E}\left[\mathrm{Var}\left(\widetilde{U}_{B,\mathcal{R}_1}|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right)\right]. \quad (6.15)
\end{aligned}$$

Conditioned upon $\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma$, the statistic $\widetilde{U}_{B,\mathcal{R}_1}$ is an average of B independent experiments. Introducing $\delta_{k,l}$ as equal to 1 if the pair (k,l) is selected in worker 1 as the 1th pair of $\widetilde{U}_{B,\mathcal{R}_1}$, and $\Delta_{k,l}$ its expected value, *i.e.* $\Delta_{k,l} := \mathbb{E}[\delta_{k,l}] = N^2\epsilon(k)\gamma(l)/nm$, it implies:

$$\mathrm{Var}\left(\widetilde{U}_{B,\mathcal{R}_1} \mid \mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right) = \frac{1}{B}\mathrm{Var}\left(\sum_{k=1}^{n}\sum_{l=1}^{m}\delta_{k,l}h(X_k, Z_l) \mid \mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right). \tag{6.16}$$

From the definition of $\delta_{k,l}$ we have $\delta_{k,l}\delta_{k_1,l_1} = 0$ as soon as $k \neq k_1$ or $l \neq l_1$, writing the right-hand-side of Eq. (6.16) as the second order moment minus the squared means gives:

$$\mathrm{Var}\left(\widetilde{U}_{B,\mathcal{R}_1} \mid \mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right) = \frac{1}{B}\sum_{k=1}^{m}\sum_{l=1}^{n}\Delta_{k,j}h^2(X_k, Z_l) - \frac{1}{B}\left(\sum_{k=1}^{n}\sum_{l=1}^{m}\Delta_{k,l}h(X_k, Z_l)\right)^2. \tag{6.17}$$

Taking the expectation of Eq. (6.17) gives:

$$\begin{aligned}\mathbb{E}\Big[\mathrm{Var}\left(\widetilde{U}_{B,\mathcal{R}_1}|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right)\Big] &= \frac{1}{B}\left[\mathbb{E}[h^2(X,Z)] - \mathbb{E}[U_{\mathcal{R}_1}^2]\right], \\ &= \frac{1}{B}\left[\mathrm{Var}(h(X,Z)) - \mathrm{Var}(U_{\mathcal{R}_1})\right]. \end{aligned}\tag{6.18}$$

Plugging Eq. (6.18) into Eq. (6.15) gives:

$$\mathrm{Var}(\widetilde{U}_{\mathbf{n},N,B}) = \frac{\mathrm{Var}(h(X,Z))}{BN} + \mathrm{Var}(U_{\mathbf{n},N}(h)) - \frac{\mathrm{Var}(U_{\mathcal{R}_i})}{BN},$$

and we can conclude from preceding results, since $U_{\mathcal{R}_i}$ is simply $U_{\mathbf{n}_0,1}$ with $n_0 = (n/N, m/N)$. □

Theorem 6.9. *If the data is distributed and repartitioned between workers with prop-SWR, we have:*

$$\begin{aligned}\mathit{Var}(\widehat{U}_{\mathbf{n},N,T}(h)) &= \mathit{Var}(U_{\mathbf{n}}(h)) + \frac{1}{T}\left[\mathit{Var}(U_{\mathbf{n},N}(h)) - \mathit{Var}(U_{\mathbf{n}}(h))\right], \\ \mathit{Var}(\widetilde{U}_{\mathbf{n},N,B,T}(h)) &= \mathit{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) + \frac{1}{NBT}\left[\sigma^2 - \mathit{Var}(U_{\mathbf{n}_0,1}(h))\right].\end{aligned}$$

Proof. Since $\mathbb{E}\left[\widehat{U}_{\mathbf{n},N,T}(h)|\mathcal{D}_n, \mathcal{Q}_m\right] = U_{\mathbf{n}}(h)$ the law of total covariances followed by the fact that, conditioned upon $\mathcal{D}_n, \mathcal{Q}$, the statistic $\widehat{U}_{\mathbf{n},N,T}(h)$ is an average of T independent random variables, implies:

$$\mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) = \mathrm{Var}(U_{\mathbf{n}}(h)) + \frac{1}{T}\mathbb{E}\left[\mathrm{Var}\left(U_{\mathbf{n},N}(h)|\mathcal{D}_n, \mathcal{Q}_m\right)\right].$$

The calculations above give the result. Explicitly,

$$\mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) = \mathrm{Var}(U_{\mathbf{n}}(h)) + \frac{1}{T}\left[\mathrm{Var}(U_{\mathbf{n},N}(h)) - \mathrm{Var}(U_{\mathbf{n}}(h))\right].$$

We now derive the variance of $\widetilde{U}_{\mathbf{n},N,B,T}$. Since

$$\mathbb{E}\left[\widetilde{U}_{\mathbf{n},N,B,T}(h)|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right] = \widehat{U}_{\mathbf{n},N,T}(h),$$

the law of total covariance followed by the calculations above imply the result:

$$\begin{aligned}\mathrm{Var}\left(\widetilde{U}_{\mathbf{n},N,B,T}(h)\right) &= \mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) + \mathbb{E}\left[\mathrm{Var}\left(\widetilde{U}_{\mathbf{n},N,B,T}(h)|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right)\right], \\ &= \mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) + \frac{1}{T}\mathbb{E}\left[\mathrm{Var}\left(\widetilde{U}_{\mathbf{n},N,B}(h)|\mathcal{D}_n, \mathcal{Q}_m, \epsilon, \gamma\right)\right], \\ &= \mathrm{Var}\left(\widehat{U}_{\mathbf{n},N,T}(h)\right) + \frac{1}{NBT}\left[\mathrm{Var}(h(X,Z)) - \mathrm{Var}(U_{\mathcal{R}_i})\right].\end{aligned}$$

□

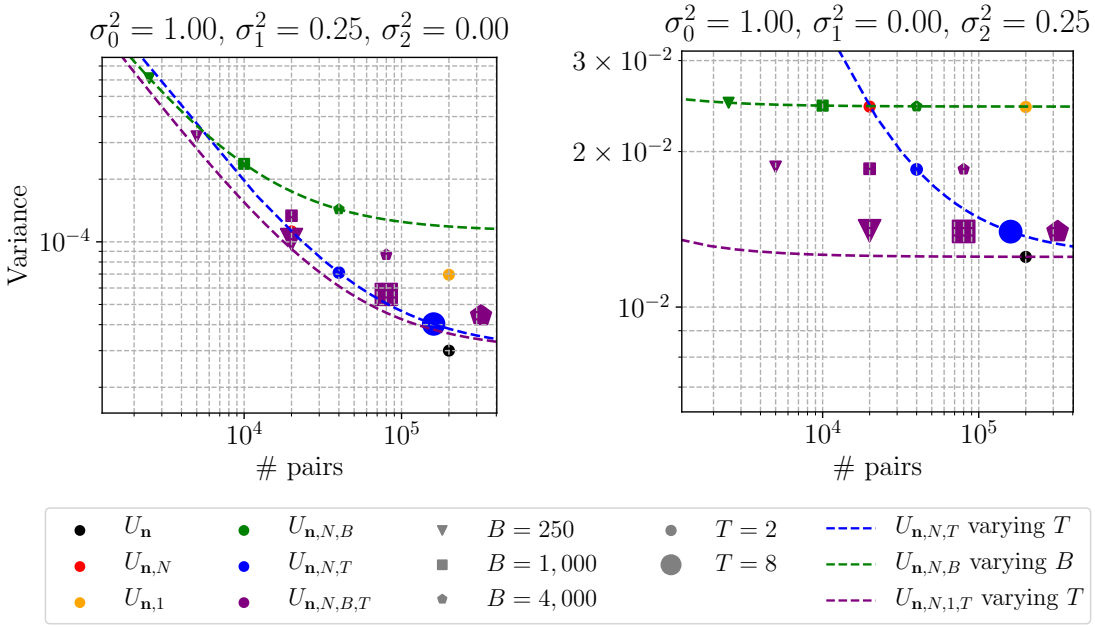


Figure 6.3: Theoretical variances as a function of the number of evaluated pairs for different estimators under prop-SWR, with $n = 100,000$, $m = 200$ and $N = 100$.

Fig. 6.3 gives a visual illustration of these results. First note that they are similar to those obtained for prop-SWOR in Fig. 6.2. Yet, the right-hand side figure shows that $\widetilde{U}_{\mathbf{n},N,B,T}$ can have a significantly lower variance than $\widehat{U}_{\mathbf{n},N,T}$, for the same number of evaluated pairs. This comes from the fact that $\widetilde{U}_{\mathbf{n},N,B,T}$ works on bootstrap re-samples of the data than $\widehat{U}_{\mathbf{n},N,T}$ and hence better corrects for the loss of information due to sampling with replacement (at the cost of more communication or disk reads). To stress this, we also represented $U_{\mathbf{n},1}$, *i.e.* the point that gives the variance of a complete estimator based on one bootstrap re-sample of the data.

6.3.5 Extension to Simple SWOR

The analysis above assumes that repartitioning is done using prop-SWOR, which has the advantage of exactly preserving the proportion of points from the two samples $\mathcal{D}_n$ and $\mathcal{Q}_m$ even in the event of significant imbalance in their size. However, a naive implementation of prop-SWOR requires some coordination between workers at each repartitioning step. To avoid exchanging many messages, we propose that the workers agree at the beginning of the protocol on a numbering of the workers, a numbering of the points in each sample, and a random seed to use in a pseudorandom number generator. This allows the workers to implement prop-SWOR without any further coordination: at each repartitioning step, they independently draw the same two random permutations over $\{1, \ldots, n\}$ and $\{1, \ldots, m\}$ using the common random seed and use these permutations to assign each point to a single worker.

Of course, other repartitioning schemes can be used instead of prop-SWOR. A natural choice is sampling without replacement (SWOR), which does not require any coordination between workers. However, the partition sizes generated by SWOR are random. This is a concern in the case of imbalanced samples, where the probability that a worker i does not get any point from the minority sample (and thus no pair to compute a local estimate) is non-negligible. For these reasons, it is difficult to obtain exact and concise theoretical variances for the SWOR case, but the results with SWOR should not deviate too much from the theoretical predictions given in Theorem 6.5 and Theorem 6.6 obtained with prop-SWOR, as illustrated numerically by the following example. Consider the kernel $h(x, z) = x \cdot z$ and random variables in $\mathbb{R}$ that follow a normal law $X \sim \mathcal{N}(\mu_X, \sigma_X)$ and $Z \sim \mathcal{N}(\mu_Z, \sigma_Z)$. In that setting, note that $\sigma_1^2 = \mu_Z^2 \sigma_X^2$, $\sigma_2^2 = \mu_X^2 \sigma_Z^2$ and $\sigma_0^2 = \sigma_X^2 \sigma_Z^2$, which means that by tweaking the parameters $\mu_X, \mu_Z, \sigma_X, \sigma_Z$, one can obtain any possible value of $\sigma_1, \sigma_2, \sigma_0$.

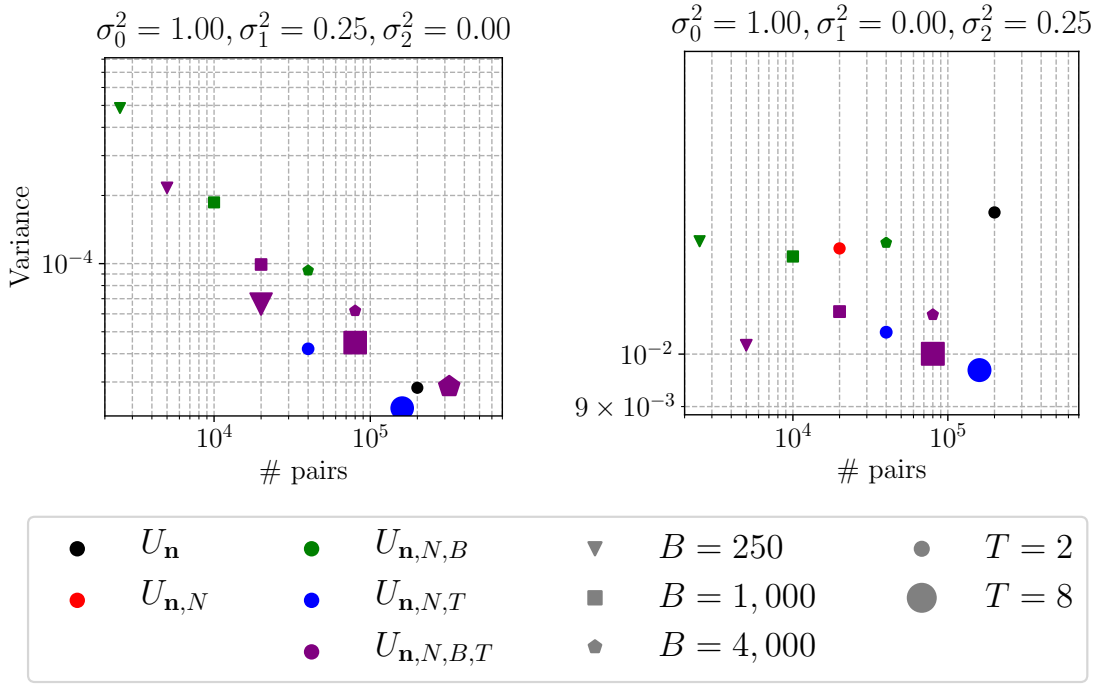


Figure 6.4: Empirical variances as a function of the number of evaluated pairs for SWOR, with $n = 100,000$, $m = 200$ and $N = 100$, evaluated over 500 runs.

The results, shown in Fig. 6.4 are very similar to those obtained for prop-SWOR in Fig. 6.2. The fact that SWOR has slightly lower variance is expected, since when no pairs are available the default value is always 0. This makes the estimator give a stable prediction, but also makes it biased.

Finally, we note that deterministic repartitioning schemes may be used in practice for simplicity. For instance, the `repartition` method in Apache Spark relies on a deterministic shuffle which preserves the size of the partitions.

6.4 Extensions to Stochastic Gradient Descent for ERM

The results of Section 6.3 can be extended to statistical learning in the empirical risk minimization framework. In such problems, given a class of kernels $\mathcal{H}$, one seeks the minimizer of (6.6) or (6.8) depending on whether repartition is used.[1] Under appropriate complexity assumptions on $\mathcal{H}$ (*e.g.*, of finite VC dimension), excess risk bounds for such minimizers can be obtained by combining our variance analysis of Section 6.3 with the control of maximal deviations based on Bernstein-type concentration inequalities as done in Clémençon et al. (2008) and Clémençon et al. (2016).

6.4.1 Gradient-based Empirical Minimization of U-statistics

In the setting of interest, the class of kernels to optimize over is indexed by a real-valued parameter $\theta \in \mathbb{R}^q$ representing the model. Adapting the notations of Section 6.3, the kernel $h : \mathcal{X}_1 \times \mathcal{X}_2 \times \mathbb{R}^q \to \mathbb{R}$ then measures the performance of a model $\theta \in \mathbb{R}^q$ on a given pair, and is assumed to be convex and smooth in θ. Empirical Risk Minimization (ERM) aims at finding $\theta \in \mathbb{R}^q$ minimizing

$$U_{\mathbf{n}}(\theta) = \frac{1}{nm} \sum_{k=1}^{n} \sum_{l=1}^{m} h(X_k, Z_l; \theta). \tag{6.19}$$

[1] Alternatively, for scalability purposes, one may instead work with their incomplete counterparts, namely (6.7) and (6.9) respectively.

The minimizer can be found by means of Gradient Descent (GD) techniques.[2] Starting at iteration $s = 1$ from an initial model $\theta_1 \in \mathbb{R}^q$ and given a learning rate $\gamma > 0$, GD consists in iterating over the following update:

$$\theta_{s+1} = \theta_s - \gamma \nabla_\theta U_{\mathbf{n}}(\theta_s). \tag{6.20}$$

Note that the gradient $\nabla_\theta U_{\mathbf{n}}(\theta)$ is itself a U-statistic with kernel given by $\nabla_\theta H$, and its computation is very expensive in the large-scale setting. In this regime, Stochastic Gradient Descent (SGD) is a natural alternative to GD which is known to provide a better trade-off between the amount of computation and the performance of the resulting model (Bottou and Bousquet, 2007). Following the discussion of Section 6.2.2, a natural idea to implement SGD is to replace the gradient $\nabla_\theta U_{\mathbf{n}}(\theta)$ in (6.20) by an unbiased estimate given by an incomplete U-statistic. The work of Papa et al. (2015) shows that SGD converges much faster than if the gradient is estimated using a complete U-statistic based on subsamples with the same number of terms.

However, as in the case of estimation, the use of standard complete or incomplete U-statistics turns out to be impractical in the distributed setting. Building upon the arguments of Section 6.3, we propose a more suitable strategy.

6.4.2 Repartitioning for Stochastic Gradient Descent

The approach we propose is to alternate between SGD steps using within-partition pairs and repartitioning the data across workers. We introduce a parameter $n_r \in \mathbb{Z}^+$ corresponding to the number of iterations of SGD between each redistribution of the data. For notational convenience, we let $r(s) := \lceil s/n_r \rceil$ so that for any worker i, $\mathcal{R}_i^{r(s)}$ denotes its data partition at iteration $s \geqslant 1$ of SGD.

Given a local batch size B, at each iteration s of SGD, we propose to adapt the strategy (6.9) by having each worker i compute a local gradient estimate using a set $\mathcal{R}_{i,B}^s$ of B randomly sampled pairs in its current local partition $\mathcal{R}_i^{r(s)}$:

$$\nabla_\theta \widetilde{U}_{B,\mathcal{R}_i^{r(s)}}(\theta_s) = \frac{1}{B} \sum_{(k,l) \in \mathcal{R}_{i,B}^s} \nabla_\theta h(X_k, Z_l; \theta_s).$$

This local estimate is then sent to the master node who averages all contributions, leading to the following global gradient estimate:

$$\nabla_\theta \widetilde{U}_{\mathbf{n},N,B}(\theta_s) = \frac{1}{N} \sum_{i=1}^N \nabla_\theta \widetilde{U}_{B,\mathcal{R}_i^{r(s)}}(\theta_s). \tag{6.21}$$

The master node then takes a gradient descent step as in (6.20) and broadcasts the updated model θ_{s+1} to the workers.

Following our analysis in Section 6.3, repartitioning the data allows to reduce the variance of the gradient estimates, which is known to greatly impact the convergence rate of SGD (see e.g. Bubeck (2015), Theorem 6.3 therein). When $n_r = +\infty$, data is never repartitioned and the algorithm minimizes an average of local U-statistics, leading to suboptimal performance. On the other hand, $n_r = 1$ corresponds to repartitioning at each iteration of SGD, which minimizes the variance but is very costly and makes SGD pointless. We expect the sweet spot to lie between these two extremes: the dominance of $\widehat{U}_{\mathbf{n},N,T}$ over $\widetilde{U}_{\mathbf{n},N,B,T}$ established in Section 6.3.3, combined with the common use of small batch size B in SGD, suggests that occasional redistributions are sufficient to correct for the loss of information incurred by the partitioning of data. We illustrate these trade-offs experimentally in the next section.

6.5 Numerical Results

In this section, we illustrate the importance of repartitioning for estimating and optimizing the Area Under the ROC Curve (AUC) through a series of numerical experiments. The corresponding

[2] When H is nonsmooth in θ, a subgradient may be used instead of the gradient.

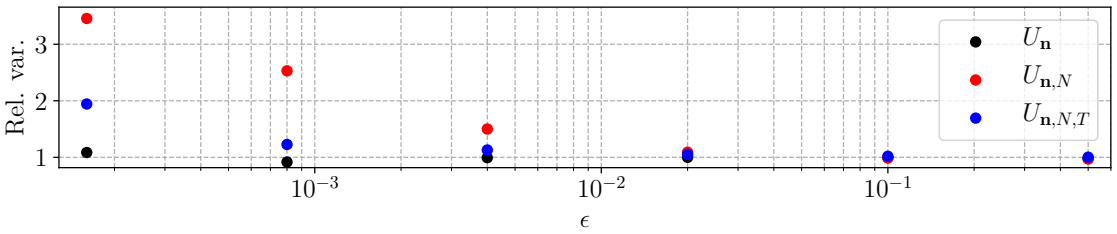


Figure 6.5: Relative variance estimated over 5000 runs, $n = 5000$, $m = 50$, $N = 10$ and $T = 4$. Results are divided by the true variance of $U_{\mathbf{n}}$ expressed from (6.22) and Theorem 6.5.

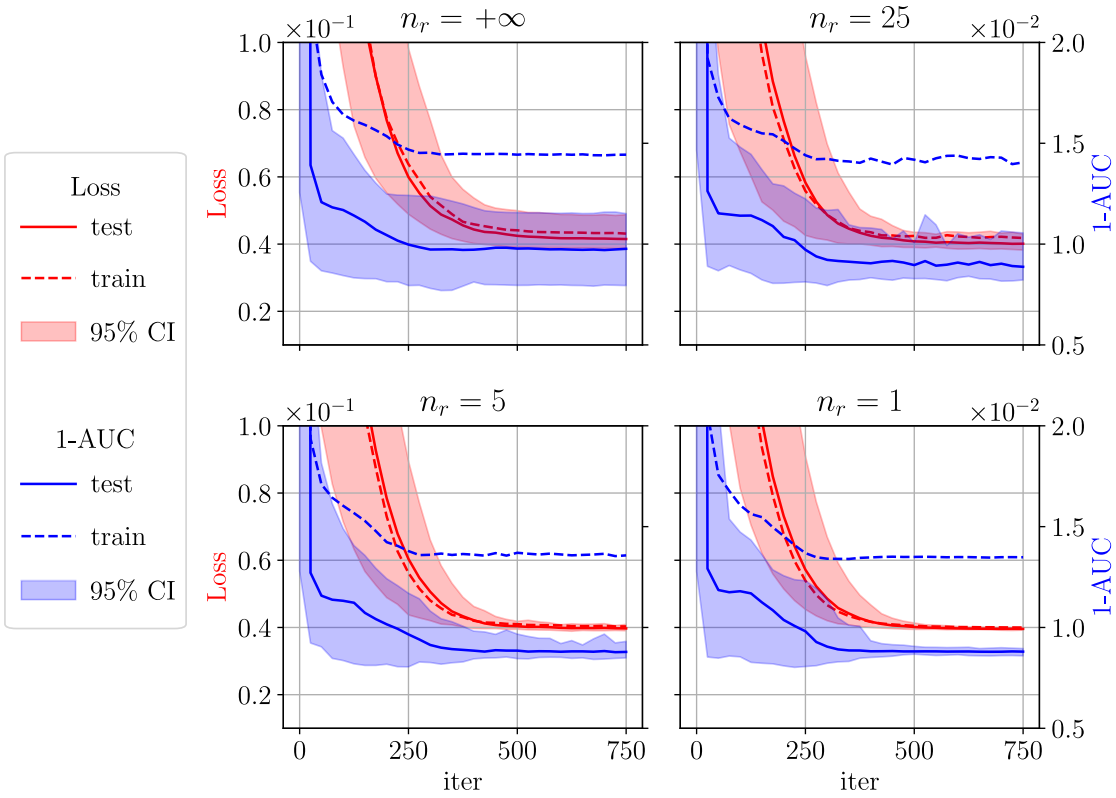


Figure 6.6: Learning dynamics for different repartition frequencies computed over 100 runs.

U-statistic is the two-sample version of the multipartite ranking VUS introduced in Example 6.2 (Section 6.2.1). The first experiment focuses on the estimation setting considered in Section 6.3. The second experiment shows that redistributing the data across workers, as proposed in Section 6.4, allows for more efficient mini-batch SGD. All experiments use prop-SWOR and are conducted in a simulated environment.

Estimation experiment. We seek to illustrate the importance of redistribution for estimating two-sample U-statistics with the concrete example of the AUC. The AUC is obtained by choosing the kernel $h(x, z) = \mathbb{I}\{z < x\}$, and is widely used as a performance measure in bipartite ranking and binary classification with class imbalance. Recall that our results of Section 6.3.3 highlighted the key role of the pairwise component of the variance σ_0^2 being large compared to the single-sample components. In the case of the AUC, this happens when the data distributions are such that the expected outcome using single-sample information is far from the truth, e.g. in the presence of hard pairs. We illustrate this on simple discrete distributions for which we can compute σ_0^2, σ_1^2 and σ_2^2 in closed form. Consider positive points $X \in \{0, 2\}$, negative points $Z \in \{-1, +1\}$ and $\mathbb{P}(X = 2) = q$, $\mathbb{P}(Z = +1) = p$. It follows that:

$$\sigma_1^2 = p^2 q(1-q), \quad \sigma_2^2 = (1-q)^2 p(1-p), \text{ and } \sigma^2 = p(1-p+pq)(1-q). \tag{6.22}$$

Assume that the scoring function has a small probability ϵ to assign a low score to a positive instance or a large score to a negative instance. In our formal setting, this translates into letting $p = 1 - q = \epsilon$ for a small $\epsilon > 0$, which implies that $\sigma_0^2/(\sigma_1^2 + \sigma_2^2) = (1-\epsilon)/(2\epsilon) \to \infty$ as $\epsilon \to 0$. We thus expect that as the true AUC $U(h) = 1 - \epsilon^2$ gets closer to 1, repartitioning the dataset becomes more critical to achieve good relative precision. This is confirmed numerically, as shown in Fig. 6.5. Note that in practice, settings where the AUC is very close to 1 are very common as they correspond to well-functioning systems, such as face recognition systems.

Learning experiment. We now turn to AUC optimization, which is the task of learning a scoring function $s : \mathcal{X} \to \mathbb{R}$ that optimizes the VUS criterion (6.2) with $K = 2$ in order to discriminate between a negative and a positive class. We learn a linear scoring function $s_{w,b}(x) = w^\top x + b$, and optimize a continuous and convex surrogate of (6.2) based on the hinge loss. The resulting loss function to minimize is a two-sample U-statistic with kernel $g_{w,b}(x, z) = \max(0, 1 + s_{w,b}(x) - s_{w,b}(z))$ indexed by the parameters (w, b) of the scoring function, to which we add a small L2 regularization term of $0.05\|w\|_2^2$ with $\|\cdot\|_2^2$ the Euclidean norm.

We use the shuttle dataset, a classic dataset for anomaly detection.[3] It contains roughly 49,000 points in dimension 9, among which only 7% (approx. 3,500) are anomalies. A high accuracy is expected for this dataset. To monitor the generalization performance, we keep 20% of the data as our test set, corresponding to 700 points of the minority class and approx. 9,000 points of the majority class. The test performance is measured with complete statistics over the 6.3 million pairs. The training set consists of the remaining data points, which we distribute over $N = 100$ workers. This leads to approx. $10,200$ pairs per worker. The gradient estimates are calculated following (6.21) with batch size $B = 100$. We use an initial step size of 0.01 with a momentum of 0.9. As there are more than 100 million possible pairs in the training dataset, we monitor the training loss and accuracy on a fixed subset of 4.5×10^5 randomly sampled pairs.

Fig. 6.6 shows the evolution of the continuous loss and the true AUC on the training and test sets along the iteration for different values of n_r, from $n_r = 1$ (repartition at each iteration) to $n_r = +\infty$ (no repartition). The lines are the median at each iteration over 100 runs, and the shaded area correspond to confidence intervals for the AUC and loss value of the testing dataset. We can clearly see the benefits of repartition: without it, the median performance is significantly lower and the variance across runs is very large. The results also show that occasional repartitions (e.g., every 25 iterations) are sufficient to mitigate these issues significantly.

6.6 Conclusion

We tackled the distributed estimation of U-statistics. We showed that the pairwise nature of U-statistics implies that naive distributed estimators may fail for some settings. We proposed new unbiased estimators, based on the idea of repartitioning the data on the available machines. Using analytical expressions, we showed that they are efficient in terms of variance, and compared their computational cost to non-distributed and naive distributed estimators of U-statistics. Finally, we proposed using the repartition procedure for minimizing U-statistics with gradient descent in a distributed environment, and confirmed the relevance of our proposition with numerical experiments.

We envision several further research questions on the topic of distributed tuplewise learning. We would like to provide a rigorous convergence rate analysis of the general distributed SGD algorithm introduced in Section 6.4. This is a challenging task because each series of iterations executed between two repartition steps can be seen as optimizing a slightly different objective function. It would also be interesting to investigate settings where the workers hold sensitive data that they do not want to share in the clear due to privacy concerns.

Similarity ranking is the pairwise bipartite ranking view of similarity ranking, introduced and studied in Chapter 5. Precisely, Chapter 5 gave guarantees for the pointwise ROC optimization or the TreeRank procedure for similarity ranking. The techniques presented in this chapter alleviate the extreme computational costs of estimating naively the quantities involved in similarity

[3] `http://odds.cs.stonybrook.edu/shuttle-dataset/`

ranking. However, the propositions for scalability of this chapter do not address the challenges specific to the optimization aspect of similarity ranking. The next chapter focuses on that aspect by means of new procedures, toy examples and illustrations, and leaves aside the scaling considerations as they were thoroughly addressed in this chapter.

Chapter 7

Practical Similarity Ranking

Summary: Chapter 5 introduced similarity ranking as the formulation of similarity learning as a pairwise bipartite ranking problem. In similarity ranking, the goal is that the larger the probability that two observations share the same label, the larger the similarity measure between them. Chapter 5 provided analyses on the generalization of selected similarity ranking problems, and Chapter 6 addressed the scalability issues associated to similarity learning. To tackle practically the problems presented in Chapter 5, optimization challenges remain. In this regard, this chapter first focuses on the pointwise ROC optimization (pROC) problem. Precisely, we give an analytical solution to empirical pROC for similarity ranking and for very specific class of functions. Then, we propose a more general gradient-based approach to pROC for bipartite ranking, that we illustrate on a toy example. The extension of that approach to similarity ranking is straightforward. The remainder of the chapter focuses on the TreeRank algorithm — see Chapter 5 for a formal presentation — and first provides illustrations of it in the pairwise setting. Then, we show that ranking forests, an ensemble method for TreeRank, can correct for a misspecficiation of its proposed splitting region. Finally, we discuss the limitations and extensions of our propositions, as well as the idea of deriving practical approaches for similarity learning from our theory of similarity ranking, a promising direction for future work.

7.1 Introduction

With the advent of deep neural networks for vision tasks, a new sub-branch of metric learning has emerged, referred to as deep metric learning. Today, all modern facial recognition algorithms are trained with deep metric learning algorithms, and some deep learning-based propositions exist for other biometries (Minaee et al. (2019)). Deep metric learning can be roughly summarized as the proposition of losses to learn a transformation of the data where natural distances have a semantic meaning, unlike in the input space $\mathcal{X}$. Formally, deep metric learning consists in learning an embedding $e : \mathcal{X} \to \mathbb{R}^d$. where $\mathbb{R}^d$ is a low-dimensional space. Then, the distance between the points $x, x' \in \mathcal{X}$ can be written as $d(e(x), e(x'))$, where $d : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is some simple distance function. In practice, the distance d is usually either the Euclidean distance or the cosine similarity.

The common idea behind all of these losses is to separate classes/identities, but they differ by the way they tackle the problem. In this section, we review several losses used in facial recognition, and refer to Wang and Deng (2018) for more details.

The original deep metric learning approach to facial recognition used the classical softmax cross-entropy (SCE). SCE is applied to a transformation of the embedding by a linear classifier $l : e(x) \mapsto W^\top e(x)$ where $W \in \mathbb{R}^{d,K}$ and K is the number of classes in the training dataset. Introduce a sample of n data points as $\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n \subset \mathcal{X} \times \{1, \dots, K\}$, as well as the notations $e_i := e(x_i) \in \mathbb{R}^d$ and $l^{(i)} := l(e(x_i)) = (l_1^{(i)}, \dots, l_K^{(i)})$ for any $i \in \{1, \dots, n\}$. Then, the

softmax cross-entropy on the observation x_i writes:

$$\mathcal{L}_{\text{SCE}} := -\log\left(\frac{\exp\left(l_{y_i}^{(i)}\right)}{\sum_{k=1}^{K}\exp\left(l_k^{(i)}\right)}\right). \tag{7.1}$$

Eq. (7.1) is simply the average of the cross-entropies between the softmax of the vectors $l^{(i)}$ and the one-hot-encoding $(\mathbb{I}\{y_i = 1\}, \dots, \mathbb{I}\{y_i = K\})$ of y_i for any i.

Separating the identities in the embedding space using the softmax cross-entropy makes intuitive sense, but is not based on the idea of computing the distance of a pair of observations, *i.e.* the end purpose of the trained system. Other losses, notably the contrastive loss and triplet loss, are based on the idea of computing distances between embeddings. With the margin parameters (ϵ_-, ϵ_+), the contrastive loss on the pair of observations $(x_i, y_i), (x_j, y_j)$ writes:

$$\mathcal{L}_{\text{cons}} := z_{i,j} \cdot \max\left(0, \|e_i - e_j\|_2 - \epsilon_+\right) + (1 - z_{i,j}) \cdot \max\left(0, \epsilon_- - \|e_i - e_j\|_2\right), \tag{7.2}$$

and $z_{i,j} = 2 \cdot \mathbb{I}\{y_i = y_j\} - 1$ for any $(i, j) \in \{1, \dots, n\}^2$. The triplet loss writes is expressed on a triplet $((x_i, y_i), (x_j, y_j), (x_k, y_k))$ of observations that satisfies $y_i = y_j$ and $y_i \neq y_k$. It writes:

$$\mathcal{L}_{\text{triplet}} := \max\left(0, \|e_i - e_j\|_2^2 - \|e_i - e_k\|_2^2 + \alpha\right), \tag{7.3}$$

In Eq. (7.3), the point (x_i, y_i) is named the *anchor point* and α is a margin parameter. Another loss, called the center loss, seeks to align the embeddings of the observations of each class on a center c_k, for any $k \in \{1, \dots, K\}$. It is expressed as follows on a single point x_i:

$$\mathcal{L}_{\text{center}} := \frac{1}{2}\|x_i - c_{y_i}\|_2^2. \tag{7.4}$$

Many other losses have been proposed in the literature, such as for example the vMF loss (Hasnat et al., 2017), the ArcFace loss (Deng et al., 2019) or the AMS loss (Wang et al., 2018). We again refer to Wang and Deng (2018) for an exhaustive presentation of those losses.

A recent experimental study, see Musgrave et al. (2020), of a non-negligible number of the losses presented in the literature has recently shown that they perform comparably when one corrects for all of the differences in models and implementations. This study suggests that most of the improvements advertised by those papers are not due to the loss functions themselves, but to the evolution of other practices in deep learning. Additionally, practitioners often consider summing a selection of those losses in different proportions when training facial recognition systems, see *e.g.* Parkhi et al. (2015). Finding the right combination requires performing extensive cross-validation for models that take several weeks to train, which is extremely costly in terms of time and computational resources. We refer to Strubell et al. (2019) for a discussion on the energy costs of deep learning algorithms.

Although these losses rely on the sound intuitions that minimizing the intra-class variations (Eq. (7.2), Eq. (7.3), Eq. (7.4)), maximizing the inter-class variations (Eq. (7.2) Eq. (7.3)) or splitting the identities (Eq. (7.1)) will improve performance, none of them relates with the evaluation of biometric systems. The evaluation of those systems essentially considers similarity learning as a ranking problem, as explained in Chapter 5. Precisely, those systems are evaluated using the ROC curve, which summarizes all attainable compromises between false positive rate and true positive rate for all possible thresholds on the similarity. In this chapter, we provide simple proof-of-concept toy experiments that draw on Chapter 5 to propose more relevant approaches to deep metric learning for biometrics. Precisely, we propose approaches designed to solve the similarity ranking problem, *i.e.* the problem of scoring on a product space (Chapter 5).

The chapter is organized as follows. Firstly, we discuss approaches to pointwise ROC optimization. Precisely, we show that for simple similarities and specific data, one can exactly optimize for specific points of the ROC curve. In a more general case, we then propose a gradient-based algorithm that explicitly optimizes for a specific point of the ROC curve. Secondly, we discuss approaches based on the TreeRank algorithm, a recursive splitting algorithm that optimizes for ranking in the ROC sense, as proven in Chapter 5. In that context, we first illustrate the output of the TreeRank algorithm for similarity ranking to provide intuition about its behavior. Then,

we show that ranking forests — a bagging procedure for TreeRank — can correct for the flaws of TreeRank. Precisely, our experiment shows that averaging ranking trees can yield a score that is indiscernible from the optimal score, even when the splitting algorithm is ill-suited for the problem. However, this setting is not covered by the analysis of Chapter 5, as we assumed that the proposed family contains the optimal splitting rules. Additionally, our experiment shows that an average of ranking trees can approximate a continuous output score, despite the fine-grained but discrete output of TreeRank.

7.2 Practical Algorithms for Pointwise ROC Optimization

Many approaches to solve the ranking problem consist in finding a score function $s : \mathcal{X} \to \mathbb{R}$ that maximizes the Area under the ROC curve (AUC), such as RankSVM in Joachims (2002), RankBoost in Freund et al. (2003) or RankNet in Burges et al. (2005). We refer to Liu (2011) (Part II) for a review of most practical approaches in ranking. However, the minimization of the AUC considers that ordering correctly the instances in the middle of the ranked list is as important as ordering those at the top, which conflicts with many applications that focus on the first best results. Some authors have proposed functionals or algorithms that focus on the top of the list, such as the p-norm push (Rudin, 2006) or the approach to optimize accuracy at the top presented in Boyd et al. (2012). Additionally, Neyman-Pearson classification Scott and Nowak (2005) concerns the optimization of the true positive rate under a constrained false positive rate. We also refer to this problem as pointwise ROC optimization (pROC). A thorough theoretical analysis of that problem was conducted in Chapter 5. Consider a sample of n data points as $\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n \subset \mathcal{X} \times \{1, \dots, K\}$. For any similarity function $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$, introduce the estimators:

$$\widehat{R}_n^-(s) = \frac{1}{n_-} \sum_{i<j} s(x_i, x_j) \cdot \mathbb{I}\{y_i \neq y_j\}, \qquad \text{and} \qquad \widehat{R}_n^+(s) \quad = \frac{1}{n_+} \sum_{i<j} s(x_i, x_j) \cdot \mathbb{I}\{y_i = y_j\},$$

with $n_+ := \sum_{i<j} \mathbb{I}\{y_i = y_j\}$ and $n_- := n(n-1)/2 - n_+$. The pROC problem then writes:

$$\text{(pROC):} \qquad \max_{s \in \mathcal{S}} \widehat{R}_n^+(s) \quad \text{subject to} \quad \widehat{R}_n^-(s) \leqslant \alpha, \tag{7.5}$$

where $\mathcal{S}$ is a proposed family of similarity functions. While this problem encompasses very natural settings, such as the optimization of a biometric system for a specific level of security, there is a lack of practical approaches for it in the literature. Notable exceptions include Scott and Nowak (2006) (Section 7.2) and Scott and Nowak (2005) (Section VI-B), which are both based on partitioning the input space $\mathcal{X}$. In that context, the objective of this section is to propose new approaches to pointwise ROC optimization. It first presents a simple experiment that learns linear similarities on simulated data, then proposes a generic gradient-descent based approach.

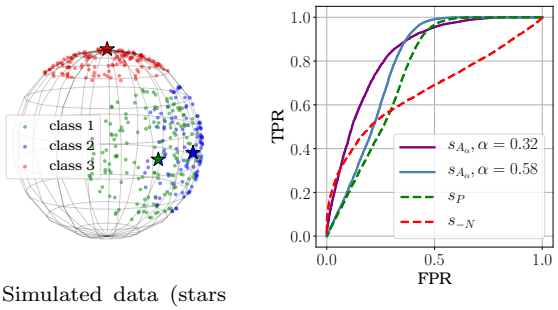
7.2.1 Exact Resolution for Bilinear Similarities

We illustrate on synthetic data that solving (7.5) for different values of α can optimize for different regions of the ROC curve. Let $\mathcal{X} \subset \mathbb{R}^d$ and let $\mathcal{S}$ be the set of bilinear similarities with norm-constrained matrices:

$$\mathcal{S} = \left\{ s_A : (x, x') \mapsto \frac{1}{2}\left(1 + x^\top A x'\right) \;\middle|\; \|A\|_F^2 \leqslant 1 \right\},$$

where $\|A\|_F^2 = \sum_{i,j=1}^d a_{ij}^2$ is the Frobenius norm of A. Note that when the data is scaled, *i.e.* $\|x\|_2 = 1$ for all $x \in \mathcal{X}$, we have $s_A(x, x') \in [0, 1]$ for all $x, x' \in \mathcal{X}$ and all $s_A \in \mathcal{S}$. Our simple experiment features $K = 3$ classes, and observations belong to the sphere in $\mathbb{R}^3$. Following the notations of Chapter 5, we denote by θ_{x,c_k} the angle between the element x and the centroid c_k of class k and set for all $k \in \{1, 2, 3\}$:

$$\mathcal{F}_k(x) \propto \mathbb{I}\left\{\theta_{x,c_k} < \frac{\pi}{4}\right\} \qquad \text{and} \qquad p_k = \frac{1}{3},$$



(a) Simulated data (stars are class centroids)

(b) ROC curves

Figure 7.1: Solving pointwise ROC optimization for a toy dataset with a bilinear similarity.

as well as $c_1 := (\cos(\pi/3), \sin(\pi/3), 0)$, $c_2 := e_2$ and $c_3 := e_3$, with (e_1, e_2, e_3) the vectors of the standard basis of $\mathbb{R}^3$. See Figure 7.1(a) for a graphical representation of the data. With the proposed family $\mathcal{S}$, Eq. (7.5) writes:

$$\max_A \frac{1}{n_+} \sum_{i<j} \mathbb{I}\{y_i = y_j\} \cdot s_A(x_i, x_j),$$
$$\text{such that } \frac{1}{n_-} \sum_{i<j} \mathbb{I}\{y_i \neq y_j\} \cdot s_A(x_i, x_j) \leqslant \alpha \quad \text{and} \quad ||A||_F^2 \leqslant 1.$$

Introduce the matrices $N, P \in \mathbb{R}^{d \times d}$, defined as:

$$P = \frac{1}{2n_+} \sum_{1 \leqslant i < j \leqslant n} \mathbb{I}\{y_i = y_j\} \cdot \left(x_i x_j^\top + x_j x_i^\top\right),$$
$$N = \frac{1}{2n_-} \sum_{1 \leqslant i < j \leqslant n} \mathbb{I}\{y_i \neq y_j\} \cdot \left(x_i x_j^\top + x_j x_i^\top\right).$$

With $\beta = 2\alpha - 1$, one can further write Eq. (7.5) as:

$$\min_A \; -\langle P, A \rangle \quad \text{s.t.} \quad \langle N, A \rangle \leqslant \beta \quad \text{and} \quad \langle A, A \rangle \leqslant 1.$$

The solutions of the problem above can be expressed in closed form using Lagrangian duality. In particular, when the constraints are saturated, the solution s_{A_α} is an increasing transformation of s_{A_α} with $A_\alpha = P - \lambda_\alpha N$, where λ_α is a positive Lagrange multiplier that decreases in α. By decreasing α, we trade-off the information contained in the positive pairs (α large, λ_α close to zero), for that in the negative pairs (α small, λ_α large). Changing α results in optimizing for different areas of the ROC curve, see Figure 7.1(b).

7.2.2 A Gradient-descent Approach for General Similarities

In this section, we propose an approach to learn a score function that solves pointwise ROC optimization by gradient descent in the case of bipartite ranking. Then, we discuss its extension to similarity ranking. Bipartite ranking considers a sample $\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n \subset \mathcal{X} \times \{-1, +1\}$ and seeks to learn a score function $s : \mathcal{X} \to \mathbb{R}_+$ that ranks positive elements $\{x_i \mid y_i = +1, 1 \leqslant i \leqslant n\}$ above negative elements $\{x_i \mid y_i = -1, 1 \leqslant i \leqslant n\}$.

We consider the resolution of Eq. (7.5) with a linear frontier on a simple 2-dimensional example. In our example, an obvious solution always exists, but is very different when α varies in $(0, 1)$. To achieve this result, we built our sample $\{(x_i, y_i)\}_{i=1}^n$ by sampling n *i.i.d.* copies of a random pair $(X, Y) \in \mathcal{X} \times \{-1, +1\}$. The *r.v.* X has the distribution F on $[0, 1]^2$ and the posterior probability $\eta(x) := \mathbb{P}\{Y = +1 \mid X = x\}$. They satisfy, for any $x \in \mathcal{X}$ such that $x = (x_1, x_2)^\top$:

$$F(x) = (4/\pi) \cdot \mathbb{I}\{x_1^2 + x_2^2 \leqslant 1,\, 0 \leqslant x_1,\, 0 \leqslant x_2\},$$
$$\eta(x) = (2/\pi) \cdot \arctan(x_2/x_1),$$

We call this distribution the quarter-circle distribution. We provide a representative sample in Fig. 7.2.

Denote by H and G respectively the distributions of the conditional random variables $X|Y=-1$ and $X|Y=+1$. Consider the region $R_\theta = \{x \in [0,1]^2 \mid \arctan(x_2/x_2) \leqslant \theta\}$. Then, simple calculations give:

$$G(R_\theta) = \mathbb{P}\left(X \in R_\theta \,|\, Y=+1\right) = \frac{4}{\pi}\left(\theta - \frac{\theta^2}{\pi}\right),$$
$$H(R_\theta) = \mathbb{P}\left(X \in R_\theta \,|\, Y=-1\right) = \frac{4\theta^2}{\pi^2}.$$

Hence, the optimal rejection region R^*_α for pointwise ROC optimization at level α simply satisfies $R^*_\alpha := \{x \,|\, \theta_x \leqslant \sqrt{\alpha}\pi/2\}$, and we have:

$$G(R^*_\alpha) = \mathbb{P}\{X \in R^*_\alpha \mid Y=+1\} = 2\sqrt{\alpha} - \alpha,$$
$$H(R^*_\alpha) = \mathbb{P}\{X \in R^*_\alpha \mid Y=-1\} = \alpha.$$

Given a region defined by a linear classifier, the oriented distance to the separator plane of the region defines a linear score function. The oriented distance to the region is the distance to the boundary of the region, with a negative sign if the point belongs to the region and a positive sign otherwise. We provide a visualization of an example sample with the optimal rejection regions in Fig. 7.2. Fig. 7.3 shows the ROC curves of the score function associated to the optimal regions.

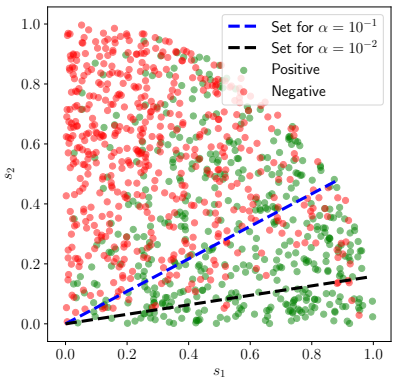


Figure 7.2: Sample and optimal rejection regions visualization with our example distribution.

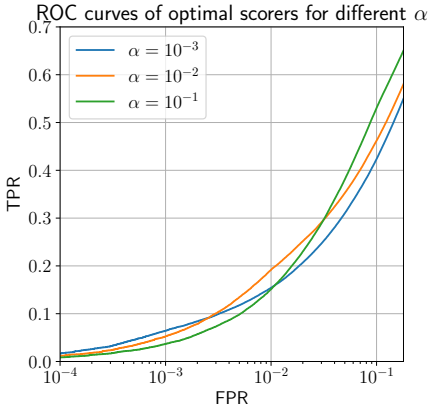


Figure 7.3: Example ROCs with optimal score functions for pointwise ROC optimization.

With the parameterized logistic function $\sigma_\lambda : t \mapsto 1/(1+\exp(-\lambda t))$, we can define relaxed versions of the true positive rate and false positive rate for the linear score function $s_{w,b}x \mapsto w^\top x$ as, respectively:

$$\widehat{G}(w,b) := \frac{1}{n_+}\sum_{Y_i=+1}\sigma_\lambda\left(w^\top x_i + b\right) \qquad \text{and} \qquad \widehat{H}(w,b) := \frac{1}{n_-}\sum_{Y_i=-1}\sigma_\lambda\left(w^\top x_i + b\right).$$

Drawing inspiration from the excess error for minimum volume set estimation of Scott and Nowak (2006), we propose the following optimization program, with $(x)_+ = \max(0,x)$:

$$\max_{(w,b)\in\mathbb{R}^d\times\mathbb{R}} \left(G(R^*_\alpha) - \widehat{G}(w,b)\right)_+ + \left(\widehat{H}(w,b) - \alpha\right)_+ \qquad \text{s.t.} \qquad \|w\|_2^2 + b^2 \leqslant 1. \tag{7.6}$$

For the toy dataset, we use a standard gradient descent approach to minimize the objective of Eq. (7.6), combined with regular projections on the unit ball to satisfy the constraint. A representative experimental result is summarized in Fig. 7.4.

Another approach to optimize for different locations on the ROC curve proposed using classification with asymmetric costs (Bach et al., 2006), *i.e.* weighting differently errors on positive and negative instances.

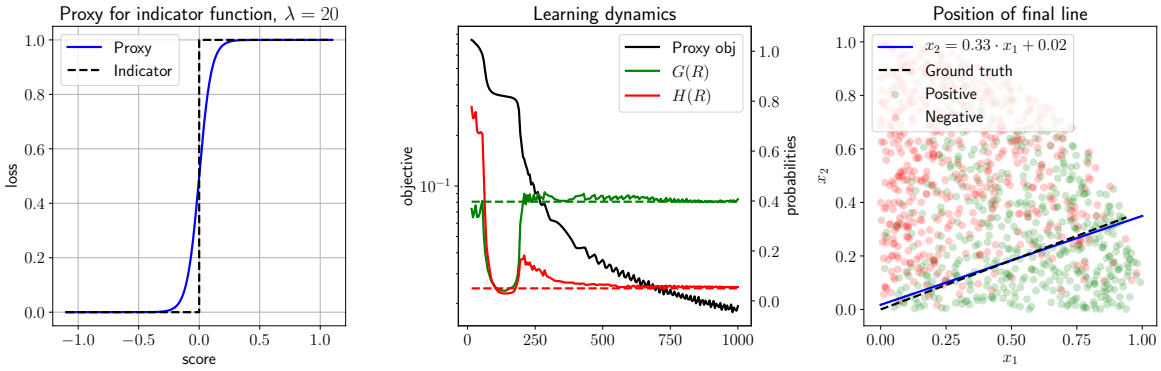


Figure 7.4: (Left) Relaxation of the indicator used. (Middle) Dynamics of the gradient-based learning process. (Right) Linear separation for the optimal and learned rejection zone.

An extension of those results could lay the foundations for differentiable losses for similarity ranking, adapted to operational constraints that can be framed as pointwise ROC optimization. Clearly, the sampling strategies described in Chapter 6 can directly contribute to the efficient learning of such similarities. Another proposition of Chapter 5 is an approach to approximate the optimal scoring function by recursive partitioning of the input space, named the TreeRank algorithm, which is the subject of the next section.

7.3 Empirical Evaluation of TreeRank

The TreeRank algorithm, first introduced in Clémençon and Vayatis (2009), tackles bipartite ranking by recursively splitting the input space. At each step, TreeRank solves classification with asymmetric costs on a specific region of the input space, and with different weights. TreeRank was first introduced with a general splitting method, referred to as LeafRank, yet only implemented with coordinate splits or combinations of those. In this section, we first discuss the extension of the TreeRank algorithm to learning similarities functions. Precisely, we illustrate the output of TreeRank for similarity ranking, i.e. with symmetric LeafRank, on the product space $\mathcal{X} \times \mathcal{X} = \mathbb{R} \times \mathbb{R}$. Then, we show the capacities of averages of scores learned with TreeRank, so-called "ranking forests", to approximate smooth score functions despite the discrete nature of each individual score. Additionally, our experiment demonstrates that averaging random ranking trees can correct for a misspecification of the LeafRank algorithm. Our analysis does not cover misspecified splitting families of sets, as it always assumes that the optimal split belongs to the family. Finally, we underline that the flexibility of TreeRank implies that LeafRank could use complex proposed regions, possibly learned using usual gradient descent algorithms. In that view, TreeRank constitute a methodology for solving bipartite ranking as a global problem, and considering variants of the original algorithm is a promising route for future work.

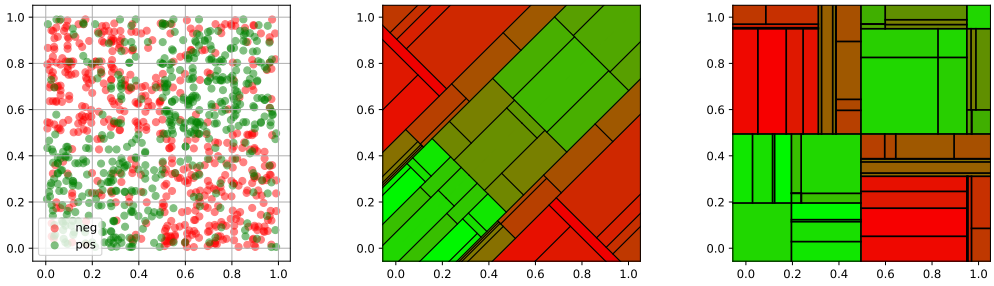
7.3.1 Symmetric Proposal Regions

Following the recommendations of Chapter 5, we introduce a symmetric transformation of the input space, *i.e.* any function $f : \mathcal{X} \times \mathcal{X} \to \mathrm{Im}(f)$ such that $f(x, x') = f(x', x)$. Then, a natural way to extend the TreeRank algorithm to similarity functions, is to choose for LeafRank the collection of regions $\mathcal{C}$, such that:

$$\mathcal{C} := \left\{ x, x' \in \mathcal{X} \times \mathcal{X} \mid f(x, x') \in D \right\}_{D \in \mathcal{D}},$$

where $\mathcal{D} \subset \mathcal{P}(\mathrm{Im}(f))$. The i-th element of the vector $f(x, x')$ is written $f_i(x, x')$. A first proposition, is to consider the transformation:

$$f(x, x') = \left(\left| x - x' \right|, \; x + x' \right)^\top.$$



(a) Training pairs (b) C_{diag} (c) C_{sq}

Figure 7.5: Representation of TreeRank score function with different simple proposed regions. The x-axis corresponds to x_1 while the y-axis corresponds to x'_1.

Introduce the collection $\mathcal{C}_{\text{diag}}$ defined as follows:

$$\mathcal{C}_{\text{diag}} := \left\{ x, x' \in \mathcal{X} \times \mathcal{X} \Big/ \sigma f_i(x, x') \geqslant \sigma A \right\}$$

where $i \in \{1, \dots, D\}$, $\sigma \in \{-1, +1\}$ and $A \in \mathbb{R}$. The collection $\mathcal{C}_{\text{diag}}$ corresponds to the set of all coordinate splits in the transformed input space. Another possible approach is to set:

$$f(x, x') = (x \vee x',\ x \wedge x')^\top,$$

where $x \vee x'$ and $x \wedge x'$ respectively stand for the element-wise maximum and minimum of x and x'. We introduce the collection $\mathcal{C}_{\text{sq}}$ of regions in $\mathcal{X} \times \mathcal{X}$ as follows:

$$\mathcal{C}_{\text{sq}} := \left\{ x, x' \in \mathcal{X} \times \mathcal{X} \Big/ \left(\sigma f_i(x, x') \geqslant \sigma A\right) \otimes \left(\sigma f_{i+D}(x, x') \leqslant \sigma A\right) \right\}$$

where $i \in \{1, \dots, D\}$, $\sigma \in \{-1, +1\}$, $A \in \mathbb{R}$ and $\otimes$ is the standard XOR.

We illustrate with Fig. 7.5 the results of the outcome of the TreeRank algorithm with either one of these two proposed families, in a simple case where $\mathcal{X} = [0, 1]$, $F = 1$ (F is the uniform distribution), $K = 2$ and:

$$\mathbb{P}\{Y = 2 | X = x\} = 0.6 \cdot \mathbb{I}\{x \geqslant 0.5\} + 0.2.$$

As shown in Fig. 7.5, the output of those decision functions is discrete, an undesirable property in many applications. Furthermore, the nature of the algorithm implies that the output score is very dependent on the first splits. This property may impair the performance of TreeRank if the proposed family of splits is underspecified. An extension of TreeRank is the notion of ranking forests, which addresses to some extent both of these issues.

7.3.2 Approaching Continuous Scores with Ranking Forests

Ranking forests is the extension by Clémençon et al. (2013) for the TreeRank algorithm of the random forests proposed in Breiman (2001). The main idea of ranking forests is to aggregate the scores of several runs of a slightly randomized TreeRank algorithm on a subset of the data, by means of a ranking aggregation procedure as argued in Clémençon et al. (2013) or simply by averaging each score function. While the output of a ranking forest is not continuous as well in theory, averaging enough trees may produce a ranking rule that is hardly distinguishable numerically from a continuous score function.

To illustrate the properties of ranking forests, we show that they can recover an optimal score for the quarter-circle distribution presented above. We considered fitting each individual tree on a randomly selected subsample of 500 elements of a training set of $n = 4,000$ points. Final performance is evaluated on an independently generated set of $n = 4,000$ points as well. We

learned in total 30 full binary trees, each of depth 4. Each LeafRank algorithm was composed of a full binary tree of recursive coordinate splits and depth 3. The aggregation of the scores was their simple average. Results are summarized in Fig. 7.7 and show that the ROC of our model is indistinguishable from the known optimal ROC on the test set. Additionally, Fig. 7.6 shows that our output score is hardly distinguishable from a continuous function on the square $[0,1]^2$.

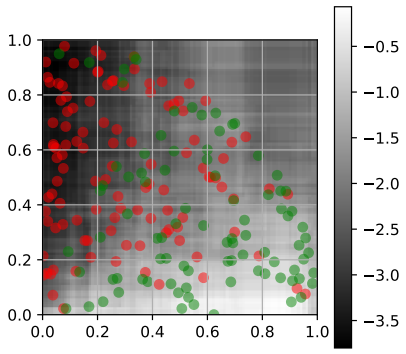


Figure 7.6: Value of the score learned with a ranking forest on the $[0,1]^2$ space.

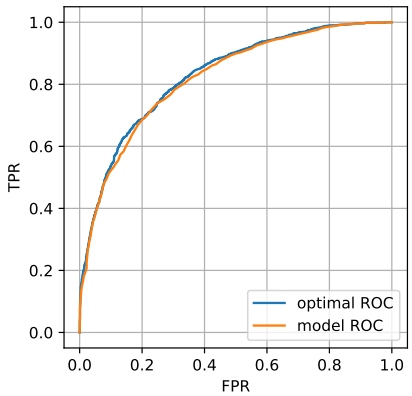


Figure 7.7: ROC curve of the score learned with a ranking forest compared with the known optimal score.

The take-home message from this example is: even though the proposed family of the LeafRank procedure is misspecified for our distribution, which breaks down the mathematical analysis of TreeRank, we can retrieve the optimal solution by averaging biased solutions. While this example is conclusive on our simple example, we can doubt its straightforward application to learning in high-dimensional spaces and/or large samples, due to its computational complexity and/or potential inaccuracy.

In that regard, more complicated splitting methods could be chosen, such as a linear decision function. In that example, LeafRank could consists in fitting an asymmetrically weighted SVM. Another approach would be learning a split of the input space using a gradient-based approach. To obtain a continous output, future work could consider smoothing the output score of TreeRank.

7.4 Conclusion

This chapter first proposed simple illustrations of optimization procedures for the pointwise ROC optimization problem presented in Chapter 5, on naive but explanatory toy examples. While the generality of those approaches is yet to prove, our examples suggest possible solutions to a problem that is rarely addressed practically in the literature, apart from few notable examples that are based on recursive partitioning. The chapter then illustrated the extension of the TreeRank algorithm to learning similarity functions instead of score functions, presented formally in Chapter 5. That approach is theoretically justified, and specifically tackles solving *similarity ranking* — *i.e.* similarity learning viewed as ranking on a product space, presented in Chapter 5 — but we address its limitations in practical scenarios. Those limitations can be alleviated with for example ranking forests.

In general, this chapter is more prospective than the others and proposes to address the optimization challenges induced by Chapter 5. As such, it initiates a dialogue between the broad statistical ranking literature of the second part of the 2000s and the rapidly expanding deep metric learning literature. It is motivated by the systematic evaluation of biometric systems with tools designed for ranking, as well as the recent rapid increase of propositions for deep metric learning that are backed solely by empirical results. In that regard, it is a promising direction for future work.

The recent improvements in similarity learning due to deep neural networks have fostered the adoption of biometric technologies, and in particular that of facial recognition. In that regard, while most of the early concerns focused on the privacy risks induced by the technology, recent research and news outlets insisted on the risks induced by its lack of reliability. The subject of the next part (Part III) of the thesis is to propose machine learning techniques to address those risks.

Precisely, we first propose a method to solve the label ranking problem with classification data (Chapter 8), which stands as a principled approach to the specific identification problem of recovering most likely suspects. Secondly, Chapter 9 addresses the issue of representativeness of a statistical population, a recurring topic in biometrics, using a reweighting scheme. Finally, while reweighting corrects some form of bias due to representation, other type of bias are integrated in the ground truth data, and can only be corrected with stronger explicit constraints on the learning procedure. In that regard, Chapter 10 proposes theoretical analyses and techniques, that address balancing predictive performance and flexible user-defined fairness criteria.

Part III

Reliable Machine Learning

Chapter 8

Ranking the Most Likely Labels

Summary: In multiclass classification, the goal is to learn a classification rule $g : \mathbb{R}^q \to \mathcal{Y}$, that accurately maps the random variable $X \in \mathbb{R}^q$ to the label $Y \in \mathcal{Y} = \{1, \ldots, K\}$. In a wide variety of situations, the task targeted may be more ambitious, and consists in sorting all the possible label values y that may be assigned to X by decreasing order of the posterior probability $\eta_y(X) = \mathbb{P}\{Y = y \mid X\}$. This chapter is devoted to the analysis of this statistical learning problem, referred to as *label ranking* here. While Part II of the thesis focuses on similarity ranking, a direct mathematization of the 1:1 biometric identification problem, label ranking corresponds to a different operational setting in biometrics. Indeed, label ranking is the problem of returning a list of most likely suspects. Formally, label ranking can be viewed as a specific variant of *ranking median regression* (RMR). Specifically, rather than observing a random permutation Σ assigned to the input vector X and drawn from a Bradley-Terry-Luce-Plackett model with conditional preference vector $(\eta_1(X), \ldots, \eta_K(X))$, the sole information available for training a label ranking rule is the label Y ranked on top, namely $\Sigma^{-1}(1)$. Inspired by recent results in RMR, we prove that under noise conditions, the One-Versus-One (OVO) approach to classification yields, as a by-product, an optimal ranking of the labels with large probability. Our theoretical analysis builds on top of finite-bounds for binary classification (Chapter 2) and probabilistic models for ranking (Chapter 4).

8.1 Introduction

In the standard formulation of the multiclass classification problem, (X, Y) is a random pair defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with unknown joint probability distribution P, where Y is a label valued in $\mathcal{Y} = \{1, \ldots, K\}$ with $K \geqslant 3$ and the *r.v.* X takes its values in a possibly high-dimensional Euclidean space, say $\mathbb{R}^q$ with $q \geqslant 1$ and models some input information that is expected to be useful to predict the output variable Y. The objective pursued is to build from training data $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, supposed to be independent copies of the generic pair (X, Y), a (measurable) classifier $g : \mathbb{R}^q \to \mathcal{Y}$ that nearly minimizes the misclassification error:

$$L(g) := \mathbb{P}\{Y \neq g(X)\}. \tag{8.1}$$

Let $\eta(x) = (\eta_1(x), \ldots, \eta_K(x))$ be the vector of posterior probabilities: $\eta_k(x) = \mathbb{P}\{Y = k \mid X = x\}$, for $x \in \mathbb{R}^q$ and $k \in \{1, \ldots, K\}$. For simplicity, we assume here that the distribution of the *r.v.* $\eta(X)$ is continuous, so that the $\eta_k(X)$'s are pairwise distinct with probability one, *i.e.* for all $(k, l) \in \{1, \ldots, K\}$ with $k < l$, then $\mathbb{P}\{\eta_k(X) = \eta_l(X)\} = 0$. It is well-known that the minimum risk is attained by the Bayes classifier

$$g^*(x) = \underset{k \in \{1, \ldots, K\}}{\arg\max}\ \eta_k(x),$$

and is equal to

$$L^* = L(g^*) = 1 - \mathbb{E}\left[\max_{1 \leqslant k \leqslant K} \eta_k(X)\right].$$

As the distribution P is unknown, a classifier must be built from the training dataset and from the perspective of statistical learning theory, the Empirical Risk Minimization (ERM) paradigm encourages us to replace the risk (8.1) by a statistical estimate $\widehat{L}_n(g)$, typically the empirical version $(1/n)\sum_{i=1}^n \mathbb{I}\{Y_i \neq g(X_i)\}$ denoting by $\mathbb{I}\{\mathcal{E}\}$ the indicator function of any event $\mathcal{E}$, and consider solutions $\widehat{g}_n$ of the optimization problem

$$\min_{g \in \mathcal{G}} \widehat{L}_n(g), \tag{8.2}$$

where the infimum is taken over a class $\mathcal{G}$ of classifier candidates, with controlled complexity (*e.g.* of finite VC dimension), though supposed rich enough to yield a small bias error $\inf_{g \in \mathcal{G}} L(g) - L^*$, *i.e.* to include a reasonable approximation of the Bayes classifier g^*. Theoretical results assessing the statistical performance of empirical risk minimizers are very well documented in the literature, see *e.g.* Devroye et al. (1996), and a wide collection of algorithmic approaches has been designed in order to solve possibly smoothed/convexified and/or penalized versions of the minimization problem (8.2).

Denoting by $\mathfrak{S}_K$ the symmetric group of order K (*i.e.* the group of permutations of $\{1, \ldots, K\}$), another natural statistical learning goal in this setup, halfway between multiclass classification and estimation of the posterior probability function $\eta(x)$ and referred to as *label ranking* throughout the chapter, is to learn, from the training data $\mathcal{D}_n$, a *ranking rule* s, *i.e.* a measurable mapping $s : \mathbb{R}^q \to \mathfrak{S}_K$, such that the permutation $s(X)$ sorts, with 'high probability', all possible label values k in $\mathcal{Y}$ by decreasing order of the posterior probability $\eta_k(X)$, that is to say in the same order as the permutation σ_X^* defined by: $\forall x \in \mathbb{R}^q$,

$$\eta_{\sigma_x^{*-1}(1)}(x) > \eta_{\sigma_x^{*-1}(2)}(x) > \ldots > \eta_{\sigma_x^{*-1}(K)}(x), \tag{8.3}$$

and σ_X^* is a random permutation that satisfies $\mathbb{P}\{\sigma_X^* = \sigma\} = \mathbb{P}\{X \in \{x \in \mathbb{R}^q \mid \sigma_x^* = \sigma\}\}$ for any $\sigma \in \mathfrak{S}_K$. This label ranking corresponds to a specific operational setting in biometrics, but not to the similarity ranking setting studied in Part II, since label ranking can be seen as returning a list of most likely suspects. Equipped with this notation, observe that $g^*(x) = \sigma_x^{*-1}(1)$ for all $x \in \mathbb{R}^q$. Given a loss function $d : \mathfrak{S}_K \times \mathfrak{S}_K \to \mathbb{R}_+$ (*i.e.* a symmetric measurable mapping *s.t.* $d(\sigma, \sigma) = 0$ for all $\sigma \in \mathfrak{S}_K$), one may formulate label ranking as the problem of finding a ranking rule s which minimizes the *ranking risk*:

$$\mathcal{R}(s) = \mathbb{E}\left[d\left(s(X), \sigma_X^*\right)\right]. \tag{8.4}$$

Except when $K = 2$ and in the case when the loss function d considered only measures the capacity of the ranking rule to recover the label that is ranked first, that is to say when $d(\sigma, \sigma') = \mathbb{I}\{\sigma^{-1}(1) \neq \sigma'^{-1}(1)\}$ (in this case, $\mathcal{R}(s) = \mathbb{P}\{g^*(X) \neq s(X)^{-1}(1)\}$), the nature of the label ranking problem significantly differs from that of multiclass classification.

There is no natural empirical counterpart of the risk (8.4) based on the observations $\mathcal{D}_n$, which makes the ERM strategy inapplicable in a straightforward fashion. It is the goal of the present chapter to show that the label ranking problem can be solved, under appropriate noise conditions, by means of the *One-Versus-One* (OVO) approach to multiclass classification. The learning strategy proposed is directly inspired from recent advances in *consensus ranking* and *ranking median regression* (RMR), see Korba et al. (2017) and Clémençon et al. (2018). an output *r.v.* Σ that takes its values in the group $\mathfrak{S}_K$ (in recommending systems, Σ may represent the preferences over a set of items indexed by $k \in \{1, \ldots, K\}$ of a given user, whose profile is described by the features X). The goal is to find a ranking rule s that minimizes $\mathbb{E}[d(s(X), \Sigma)]$, that is to say, for any $x \in \mathbb{R}^q$, a *consensus/median ranking* $s(x) \in \mathfrak{S}_K$ related to the conditional distribution of Σ given $X = x$ *w.r.t.* the metric $d(.,.)$.

In this chapter, by means of a coupling technique, we show that the label ranking problem stated above can be viewed as a variant of RMR where the output ranking is very partially observed in the training stage, through the label ranked first solely. While previous authors (Korba et al. (2018) or Brinker and Hüllermeier (2019)) tackled RMR with partial information, they do not

feature the strong theoretical guarantees that we provide. Based on this analogy, the main result of the chapter shows that the OVO method permits to recover the optimal label ranking with high probability, provided that noise conditions are fulfilled for all binary classification subproblems. Incidentally, the analysis carried out provides statistical guarantees in the form of (possibly fast) learning rate bounds for the OVO approach to multiclass classification under the hypotheses stipulated. Our contribution focuses on label ranking, but builds on top of both the OVO approach and recent literature in RMR. Finally, various numerical experiments corroborate empirically the theoretical results established in this chapter.

The chapter is organized as follows. In section 8.2, the OVO methodology for multiclass classification is recalled at length, together with recent results in RMR. The main results of the chapter are stated in section 8.3: principally, a coupling result connecting label ranking to RMR and statistical guarantees for the OVO approach to label ranking in the form of nonasymptotic probability bounds. Numerical experiments are displayed in section 8.4, while some concluding remarks are collected in section 8.5.

8.2 Preliminaries

To begin with, we recall the OVO approach for defining a multiclass classifier from binary classifiers. Basic hypotheses and results related to Ranking Median Regression (RMR) are next briefly described.

8.2.1 From Binary to Multiclass Classification

A classifier g is entirely characterized by the collection of subsets of the feature space $\mathbb{R}^q$: $(S_g(1), \ldots, S_g(K))$, where $S_g(k) = \{x \in \mathbb{R}^q : \ g(x) = k\}$ for $k \in \{1, \ldots, K\}$. Observe that the S_k's are pairwise disjoint and their union is equal to $\mathbb{R}^q$. Hence, they form a partition of $\mathbb{R}^q$, except that it may happen that a certain subset $S_k(g)$ is empty, *i.e.* a certain label k is never predicted by g.

The OVO approach. Partitioning the feature space $\mathbb{R}^q$ in more than two subsets may lead to practical difficulties and certain learning algorithms such as Support Vector Machines (SVM's) are originally tailored to the binary situation (*i.e.* to the case $K = 2$). In this case, a natural way to extend such algorithms is the 'One-Versus-One' approach to multi-class classification (Hastie and Tibshirani, 1997; Moreira and Mayoraz, 1998; Allwein et al., 2000; Fürnkranz, 2002; Wu et al., 2004). It consists in running the binary classification algorithm $K(K-1)/2$ times once for each binary subproblem. For any $1 \leqslant k < l \leqslant K$, the binary subproblem of k against l is based on the fraction of the training data with labels in $\{k, \ l\}$ only:

$$\mathcal{D}_{k,l} = \{(X_i, Y_i) \mid \ Y_i \in \{k, l\}, \ i = 1, \ \ldots, \ n\},$$

and the binary classification algorithm outputs a classification rule $g_{k,l} : \mathbb{R}^q \to \{-1, \ +1\}$ with risk:

$$L_{k,l}(g_{k,l}) = \mathbb{P}\{Y_{k,l} \neq g_{k,l}(X) \mid Y \in \{k, l\}\},$$

as small as possible, where $Y_{k,l} = \mathbb{I}\{Y = l\} - \mathbb{I}\{Y = k\}$. The OVO approach combines, for any possible input value $x \in \mathbb{R}^q$, the binary predictions $g_{k,l}(x)$ to produce a multi-class classifier $\bar{g} : \mathbb{R}^q \to \{1, \ \ldots, \ K\}$ with minimum risk $R(\bar{g})$. A possible fashion of combining the results of the $K(K-1)/2$ 'duels' is to take as predicted label which has won the largest number of duels (and stipulate a rule for breaking possible ties). The rationale behind this OVO approach lies in the fact that:

$$g^*(x) = \underset{k \in \{1, \ \ldots, \ K\}}{\arg\max} \ N_k^*(x), \tag{8.5}$$

where, for all $(k, x) \in \{1, \ \ldots, \ K\} \times \mathbb{R}^q$, $N_k^*(x)$ denotes the number of duels won by label k with optimal/Bayes classifiers for all binary subproblems, namely:

$$N_k^*(x) = \sum_{l<k} \mathbb{I}\{g_{l,k}^*(x) = +1\} + \sum_{k<l} \mathbb{I}\{g_{k,l}^*(x) = -1\},$$

where $g^*_{l,m}(x) = 2\mathbb{I}\{\eta_m(x)/(\eta_m(x) + \eta_l(x)) > 1/2\} - 1$ is the minimizer of the risk $L_{l,m}$ for $l < m$. The proof is straightforward. Indeed, it suffices to observe that, for all $i \in \{1, \ldots, K\}$, $N^*_{\sigma^*_x(i)} = K - i$.

Remark 8.1. (One-Versus-All) *An alternative to the OVO approach in order to reduce multiclass classification to binary subproblems and apply the SVM methodology consists in comparing each class to all of the others in K two-class duels. A test point is classified as follows: the signed distances from each of the K separating hyperplanes are computed, the winner being simply the class corresponding to the largest signed distance. However, other rules have been proposed in Vapnik (1998) and in Weston and Watkins (1999).*

Label Ranking. As underlined in the Introduction section, rather than learning to predict the most likely label given X, it may also be desirable to rank all possible labels according to their conditional likelihood. The goal is then to recover the permutation σ^*_X defined through (8.3). Practically, this boils down to building a predictive rule $s(x)$ from the training data $(X_1, Y_1), \ldots, (X_n, Y_n)$ that maps $\mathbb{R}^q$ to $\mathfrak{S}_K$ and minimizes the ranking risk (8.4), where $d(., .)$ is an appropriate loss function defined on $\mathfrak{S}_K \times \mathfrak{S}_K$. For instance, one may consider $\mathbb{I}\{\sigma \neq \sigma'\}$ or the Hamming distance $\sum_{k=1}^K \mathbb{I}\{\sigma(k) \neq \sigma'(k)\}$ to measure the dissimilarity between two permutations σ and σ' in $\mathfrak{S}_K$. Classic metrics on $\mathfrak{S}_K$ (see Deza and Huang (1998)) also provide natural choices for the loss function, including:

- the Kendall τ distance: $\forall(\sigma,\ \sigma') \in \mathfrak{S}_K^2$,

$$d_\tau(\sigma,\ \sigma') = \sum_{i<j} \mathbb{I}\{(\sigma(i) - \sigma(j)) \cdot (\sigma'(i) - \sigma'(j)) < 0\},$$

- the Spearman footrule: $\forall(\sigma,\ \sigma') \in \mathfrak{S}_K^2$,

$$d_1(\sigma,\ \sigma') = \sum_{i=1}^K \left|\sigma(i) - \sigma'(i)\right|,$$

- the Spearman ρ distance: $\forall(\sigma,\ \sigma') \in \mathfrak{S}_K^2$,

$$d_2(\sigma,\ \sigma') = \sum_{i=1}^K (\sigma(i) - \sigma'(i))^2.$$

As shall be explained below, the label ranking problem can be viewed as a variant of the standard ranking median regression problem.

8.2.2 Ranking Median Regression

This problem of minimizing (8.4) shares some similarity with that referred to as *ranking median regression* in Clémençon et al. (2018), also called *label ranking* sometimes, see *e.g.* Tsoumakas et al. (2009) and Vembu and Gärtner (2010). In this supervised learning problem, the output associated with the input variable X is a random vector Σ taking its values in $\mathfrak{S}_K$ (expressing the preferences on a set of items indexed by $k \in \{1, \ldots, K\}$ of a user with a profile characterized by X drawn at random in a certain statistical population) and the goal pursued is to learn from independent copies $(X_1, \Sigma_1), \ldots, (X_n, \Sigma_n)$ of the pair (X, Σ) a (measurable) ranking rule $s : \mathbb{R}^q \to \mathfrak{S}_K$ that nearly minimizes:

$$R(s) := \mathbb{E}[d(\Sigma, s(X))]. \tag{8.6}$$

The name *ranking median regression* arises from the fact that any rule mapping X to a median of Σ's conditional distribution given X *w.r.t.* the metric/loss d (refer to Korba et al. (2017) for a statistical learning formulation of the consensus/median ranking problem) is a minimizer of (8.6), see Proposition 5 in Clémençon et al. (2018). In certain situations, the minimizer of (8.6) is unique and a closed analytic form can be given for the latter, based on the pairwise probabilities: $p_{i,j}(x) = \mathbb{P}\{\Sigma(i) < \Sigma(j) \mid X = x\} := 1 - p_{j,i}(x)$ for $1 \leqslant i < j \leqslant K$ and $x \in \mathbb{R}^q$.

Assumption 8.2. *For all $x \in \mathbb{R}^q$, we have: $\forall (i,k,l) \in \{1, \ldots, K\}^3$, $p_{i,j}(x) \neq 1/2$ and*

$$p_{i,j}(x) > 1/2 \text{ and } p_{j,k}(x) > 1/2 \Rightarrow p_{i,k}(x) > 1/2. \tag{8.7}$$

Indeed, when choosing the Kendall τ distance d_τ as loss function, it has been shown that, under Assumption 8.2, referred to as *strict stochastic transitivity*, the minimizer of (8.6) is almost-surely unique and given by: $\forall k \in \{1, \ldots, K\}$, with probability one:

$$s_X^*(k) = 1 + \sum_{l \neq k} \mathbb{I}\{p_{k,l}(X) < 1/2\}. \tag{8.8}$$

Remark 8.3. (Conditional BTLP model) *A Bradley-Terry-Luce-Plackett model for Σ's conditional distribution given X, $P_{\Sigma|X}$, assumes the existence of a hidden preference vector $w(X) = (w_1(X), \ldots, w_K(X))$, where $w_k(X) > 0$ is interpreted as a preference score for item k of a user with profile X, see e.g. Bradley and Terry (1952), Luce (1959) or Plackett (1975). The conditional distribution of Σ^{-1} given X can be defined sequentially as follows: $\Sigma^{-1}(1)$ is distributed according to a multinomial distribution of size 1 with support $\mathbf{S}_1 = \{1, \ldots, K\}$ and parameters $w_k(X)/\sum_l w_l(X)$ and, for $k > 1$, $\Sigma^{-1}(k)$ is distributed according to a multinomial distribution of size 1 with support $\mathbf{S}_k = \mathbf{S}_1 \backslash \{\Sigma^{-1}(1), \ldots, \Sigma^{-1}(k-1)\}$ with parameters $w_l(X)/\sum_{m \in \mathbf{S}_k} w_m(X)$, $l \in \mathbf{S}_k$. The conditional pairwise probabilities are given by $p_{k,l}(X) = w_k(X)/(w_k(X) + w_l(X))$ and one may easily check that Assumption 8.2 is fulfilled as soon as the $w_k(X)$'s are pairwise distinct with probability one. In this case, $s^*(X)$ is the permutation that sorts the $w_k(X)$'s in decreasing order.*

In Clémençon et al. (2018), certain situations where empirical risk minimizers over classes of ranking rules fulfilling appropriate complexity assumptions can be proved to achieve fast learning rates (*i.e.* faster than $O(1/\sqrt{n})$) have been investigated. More precisely, denoting by $\operatorname{ess\,inf} Z$ the essential infimum of any real valued *r.v.* Z, the following 'noise condition' related to conditional pairwise probabilities was considered.

Assumption 8.4. *The pairwise probabilities $p_{i,j}$ satisfy:*

$$H = \operatorname{ess\,inf} \min_{i<j} |p_{i,j}(X) - 1/2| > 0. \tag{8.9}$$

Precisely, it is shown in Clémençon et al. (2018) (see Proposition 7 therein) that, under Assumptions 8.2-8.4, minimizers of the empirical version of (8.6) over a VC-major class of ranking rules with the Kendall τ distance as loss function achieves a learning rate bound of order $1/n$ (without the impact of model bias). Since $\mathbb{P}_X\{s(X) \neq s_X^*\} \leqslant (1/H) \times (R(s) - R(s_\cdot^*))$ (*cf* Eq. (13) in Clémençon et al. (2018)), a bound for the probability that the empirical risk minimizer differs from the optimal ranking rule at a random point X can be immediately derived.

8.3 Label Ranking

We now describe at length the connection between label ranking and RMR and state the main results of the chapter.

8.3.1 Label Ranking as RMR

The major difference of the multi-class classification context with label ranking lies in the fact that only the partial information $\sigma_X^{*-1}(1)$ is observable in the presence of noise — under the form of the random label Y assigned to X ($\sigma_X^{*-1}(1)$ is the mode of the conditional distribution of Y given X) — in order to mimic the optimal rule σ_X^*.

Lemma 8.5. *Let (X, Y) be a random pair on the probability space $(\Omega, \mathcal{F}. \mathbb{P})$. One may extend the sample space so as to build a random variable Σ that takes its values in $\mathfrak{S}_K$ and whose conditional distribution given X is a BTLP model with preference vector $\eta(X) = (\eta_1(X), \ldots, \eta_K(X))$ such that*

$$Y = \Sigma^{-1}(1) \text{ with probability one.} \tag{8.10}$$

Proof. Our proof shows that we can define a random permutation $\Sigma \in \mathfrak{S}_K$ with the desired properties. To begin with, define $\Sigma^{-1}(1)$ as Y. Next, given X and $\Sigma^{-1}(1) = Y$, draw Σ' as a BTLP model on the set $\mathcal{I} = \{1, \ldots, K\}\backslash\{\Sigma^{-1}(1)\}$ with preference parameters $\eta_k(X)$, $k \in \mathcal{I}$. For all $r \in \{1, \ldots, K-1\}$, set $\Sigma^{-1}(r+1) = \Sigma'^{-1}(r)$ and invert the permutation $(\Sigma^{-1}(1), \ldots, \Sigma^{-1}(K))$ to get the desired random permutation Σ. □

The noteworthy fact that the probabilities related to the optimal pairwise comparisons $\mathbb{P}\{g^*_{k,l}(X) = +1 \mid Y \in \{k, l\}\} = \eta_k(X)/(\eta_k(X) + \eta_l(X))$ are given by a BTLP model has been pointed out in Hastie and Tibshirani (1997). With the notations introduced in Lemma 8.5, we have in addition

$$\begin{aligned}\eta_{k,l}(X) :&= \mathbb{P}\left\{\Sigma(k) < \Sigma(l) \mid X\right\}, \\ &= \eta_k(X)/(\eta_k(X) + \eta_l(X)).\end{aligned}$$

Eq. (8.10) can be interpreted as follows: the label ranking problem as defined in subsection 8.2.1 can be viewed as a specific RMR problem under strict stochastic transitivity (*i.e.* Assumption 8.2 is always fulfilled since the $\eta_k(X)$ are *a.s.* pairwise distinct) with *incomplete observations*

$$\left(X_1, \Sigma_1^{-1}(1)\right), \ldots, \left(X_n, \Sigma_n^{-1}(1)\right), \tag{8.11}$$

since every observation $(X_i, \Sigma_i^{-1}(1))$ in Eq. (8.11) only contains the top element $\Sigma_i^{-1}(1)$ of the random permutation Σ, while all of Σ is available in RMR.

Due to the incomplete character of the training data, one cannot recover the optimal ranking rule σ^*_X by minimizing a statistical version of (8.6) of course. As an alternative, one may attempt to build directly an empirical version of σ^*_X based on the explicit form (8.8), which only involves pairwise comparisons, in a similar manner as in Korba et al. (2017) for consensus ranking. Indeed, in the specific RMR problem under study, Eq. (8.8) becomes

$$\sigma^*_X(k) = 1 + \sum_{l \neq k} \mathbb{I}\{g^*_{k,l}(X) = -1\}, \tag{8.12}$$

for all $k \in \{1, \ldots, K\}$. The OVO procedure precisely permits to construct such an empirical version, by replacing the optimal classifier $g^*_{k,l}$ in Eq. (8.12) by a minimizer of the empirical classification error. As shall be shown by the subsequent analysis, in spite of the very partial nature of the statistical information at disposal, the OVO approach permits to recover the optimal RMR rule σ^*_X with high probability provided that (X, Σ) fulfills (a possibly weakened version of) Assumption 8.4, combined with classic complexity conditions.

Remark 8.6. (On the noise condition) *Attention should be paid to the fact that, when applied to the random pair (X, Σ) defined in Lemma 8.5, Assumption 8.4 simply means that the classic Massart's noise condition is fulfilled for every binary classification subproblem, see Massart and Nédélec (2006).*

8.3.2 The OVO Approach to Label Ranking

Let $\mathcal{G}$ be a class of decision rules $g : \mathbb{R}^q \to \{-1, +1\}$. As stated in subsection 8.2.1, the OVO approach to multiclass classification is implemented as follows. For all $k < l$, compute a minimizer $\widehat{g}_{k,l}$ of the empirical risk:

$$\widehat{L}_{k,l}(g) := \frac{1}{n_k + n_l} \sum_{i:\ Y_i \in \{k,\, l\}} \mathbb{I}\{g(X_i) \neq Y_{k,l,i}\}, \tag{8.13}$$

over the class $\mathcal{G}$, with $Y_{k,l,i} = \mathbb{I}\{Y_i = l\} - \mathbb{I}\{Y_i = k\}$ for $i \in \{1, \ldots, n\}$ and the convention that $0/0 = 0$. We set $\widehat{g}_{l,k} = -\widehat{g}_{k,l}$ for $k < l$ by convention. Equipped with these $\binom{K}{2}$ classifiers, for any test (*i.e.* input and unlabeled) random variable X, the $\widehat{g}_{k,l}(X)$'s define a *complete directed graph* G_X with the K labels as vertices: $\forall k < l$, $l \to_X k$ if $\widehat{g}_{k,l}(X) = +1$ and $k \to_X l$ otherwise. The analysis carried out in the next subsection shows that under appropriate noise conditions, with large probability, the random graph G_X is *acyclic*, meaning that the complete binary relation

$l \to_X k$ is transitive (*i.e.* $l \to k$ and $k \to_X m \Rightarrow l \to_X m$), in other words that the *scoring function*:

$$\begin{aligned} \widehat{s}(X)(k) &:= 1 + \sum_{k \neq l} \mathbb{I}\left\{\widehat{g}_{k,l}(X) = -1\right\}, \\ &= 1 + \sum_{k \neq l} \mathbb{I}\left\{k \to_X l\right\}, \text{ for } k \in \{1, \ldots, K\}, \end{aligned} \tag{8.14}$$

defines a permutation, which, in addition, coincides with σ_X^*, *cf* Eq. (8.12). The equivalence between the transitivity of $\to_X$, the acyclicity of G_X and the membership of $\widehat{s}(X)$ in $\mathfrak{S}_K$ is straightforward, but we refer to Theorem 5's proof of Eq. (9) in Korba et al. (2017) for more details.

Remark 8.7. *The quantity* (8.14) *can be related to the Copeland score, see Copeland (1951): the score $\widehat{s}(X)(k)$ of label k being equal to* 1 *plus the number of duels it has lost, while its Copeland score $C_X(k)$ is its number of victories minus its number of defeats, so that*

$$\widehat{s}(X)(.) = (K + 1 - C_X(.))/2.$$

OVO Approach to Label Ranking

Inputs. Class $\mathcal{G}$ of classifier candidates. Training classification dataset $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$. Query point $x \in \mathbb{R}^q$.

1. (Binary classifiers.) For $k < l$, based on $\mathcal{D}_{k,l} = \{(X_i, Y_i) : Y_i \in \{k, l\}, i = 1, \ldots, n\}$, compute the ERM solution to the binary classification problem:

$$\widehat{g}_{k,l} = \arg\min_{g \in \mathcal{G}} \widehat{L}_{k,l}(g).$$

2. (Scoring.) Compute the predictions $\widehat{g}_{k,l}(x)$ and the score for the query point x:

$$\widehat{s}(x)(k) = 1 + \sum_{l \neq k} \mathbb{I}\{\widehat{g}_{k,l}(x) = -1\}.$$

Output. Break arbitrarily possible ties in order to get a prediction $\widehat{\sigma}_x$ in $\mathfrak{S}_K$ at x from $\widehat{s}(x)$.

Figure 8.1: Pseudo-code for 'OVO label ranking'

While the rest of the paper focuses on events where G_X is transitive, the end of this subsection present practical approaches in the opposite case. If G_X is not transitive, or equivalently when $\widehat{s}(X) \notin \mathfrak{S}_K$, one may build a ranking $\widehat{\sigma}_X$ from the scoring function (8.14) by breaking ties in an arbitrary fashion, as proposed below for simplicity. Alternatives could be considered of course. The issue of building a ranking/permutation of the labels in $\{1, \ldots, K\}$ from (8.14) can be connected with the *feedback set* problem for directed graphs, see *e.g.* Battista et al. (1998): for a directed graph, a minimal feedback arcset is a set of edges of smallest cardinality such that a directed acyclic graph is obtained when reversing the edges in it. We refer to *e.g.* Festa et al. (1999) for algorithms.

8.3.3 Statistical Guarantees for Label Ranking

It is the purpose of the subsequent analysis to show that, provided that the conditions listed below are fulfilled, the ranking rule σ_X^* can be fully recovered with high probability through the OVO approach previously described. We denote by F the marginal distribution of the input variable X, by F_k the conditional distribution of X given $Y = k$ and set $p_k = \mathbb{P}\{Y = k\}$ for $k \in \{1, \ldots, K\}$.

Assumption 8.8. *There exists $\alpha \in [0, 1]$ and $B > 0$ such that: for all $k < l$ and $t \geqslant 0$,*

$$\mathbb{P}\left\{|2 \cdot \eta_{k,l}(X) - 1| < t\right\} \leqslant B t^{\frac{\alpha}{1-\alpha}}.$$

Assumption 8.9. *The class $\mathcal{G}$ is of finite* VC *dimension $V < +\infty$.*

Assumption 8.10. *There exists a constant $\varepsilon > 0$, s.t. for all $k \neq l$ in $\{1, \ldots, K\}$ and $x \in \mathbb{R}^q$, $\eta_k(x) + \eta_l(x) > \varepsilon$.*

Assumption 8.8 means that Assumption 8.4 is satisfied by the random pair (X, Σ) defined in Lemma 8.5 in the case $\alpha = 1$ (notice incidentally that it is void when $\alpha = 0$) and reduces to the classic Mammen-Tsybakov noise condition in the binary case $K = 2$, see Mammen and Tsybakov (1999). Under those assumptions, one can derive Lemma 8.11 below, which provides guarantees for the (k, l) OVO classification problem.

Lemma 8.11. *Suppose that Assumptions 8.8-8.10 are fulfilled. Let $1 \leqslant k < l \leqslant K$. Then, for all $\delta \in (0, 1)$, we have $\forall n \geqslant 1$, w.p. $\geqslant 1 - \delta$:*

$$L_{k,l}(\widehat{g}_{k,l}) - L^*_{k,l} \leqslant 2\left(\inf_{g \in \mathcal{G}} L_{k,l}(g) - L^*_{k,l}\right) + r_n(\delta), \tag{8.15}$$

where, for all $n \geqslant n_0(\delta, \alpha, \epsilon, B, V)$ and $\delta \in (0, 1)$,

$$r_n(\delta) = 2\left(1/(nh)\right)^{\frac{1}{2-\alpha}} \times \left[\left(64C^2V \log n\right)^{\frac{1}{2-\alpha}} + \left(32 \log(2/\delta)\right)^{\frac{1}{2-\alpha}}\right],$$

where $C > 0$ is an universal constant and $h = \epsilon^{2-2\alpha}/(16\beta)$.

Proof. The result is a slight variant of that proved in Boucheron et al. (2005) (pages 342-346). The sole difference lies in the fact that fact that the empirical risk (and, consequently, its minimizer as well) is built from a random number of training observations (*i.e.* those with labels in $\{k,\ l\}$). Here, we detail the proof for completion.

The derivation of fast learning speeds for general classes of functions relies on a sensible use of Talagrand's inequality that exploits the upper bound on the variance of the loss provided by the noise condition, combined with convergence bounds on Rademacher averages, see P. Bartlett and Mendelson (2005).

To begin with, we define classes of functions that mirror the ones used by Boucheron et al. (2005). Those are specifically introduced for the problem of associating elements of the sample to the label k or l, with $k < l, (k, l) \in \{1, \ldots, K\}^2$ any pair of labels. Given a label $y \in \mathcal{Y}$, it corresponds to solving binary classification for $\phi_{k,l}(y) = \mathbb{I}\{y = k\} - \mathbb{I}\{y = l\}$ for all of the concerned instances, *i.e.* those with labels k or l. For each binary classifier g in $\mathcal{G}$, we introduce the cost function $c_{k,l}$ and the proportion of concerned instances $h_{k,l}$, such that, for all $x, y \in R^q \times \{1, \ldots, K\}$,

$$c_{k,l}(x, y) := \mathbb{I}\{g(x) \neq \phi_{k,l}(y), y \in \{k, l\}\} \quad \text{and} \quad h_{k,l}(y) := \mathbb{I}\{y \in \{k, l\}\}.$$

We denote by $\mathcal{F}_{k,l}$ the set that contains the regrets of all functions $g \in \mathcal{G}$, formally:

$$\mathcal{F}_{k,l} := \left\{f_{k,l} : (x, y) \mapsto \mathbb{I}\{y \in \{k, l\}\} \cdot \left(c_{k,l}(x, y) - \mathbb{I}\{g^*_{k,l}(x) \neq \phi_{k,l}(y)\}\right) \mid g \in \mathcal{G}\right\}.$$

Note that $c_{k,l}$ has an implicit dependence on a classifier g and that $\mathcal{F}_{k,l}$ has an implicit dependence on $\mathcal{G}$. With P as the expectation over X, Y and P_n as the empirical measure, one can rewrite the risk $L_{k,l}$ and empirical risk $\widehat{L}_{k,l}$ as:

$$L_{k,l}(g) = \frac{Pc_{k,l}}{Ph_{k,l}} \quad \text{and} \quad \widehat{L}_{k,l}(g) = \frac{P_n c_{k,l}}{P_n h_{k,l}}.$$

Unlike $c_{k,l}$ the empirical mean $P_n h_{k,l}$ does not depend on an element of $g \in \mathcal{G}$, thus minimizing $\widehat{L}_{k,l}$ is the same problem as minimizing $P_n c_{k,l}$. The rest of the proof consists in using Boucheron et al. (2005) (section 5.3.5) to derive an upper bound of Pf, with $f \in \mathcal{F}_{k,l}$. Talagrand's inequality is useful because of an upper-bound on the variance of the elements in $\mathcal{F}_{k,l}$.

Assumption 8.8 induces a control on the variance of the elements of $\mathcal{F}_{k,l}$. Bousquet et al. (2003) (page 202) reviewed equivalent formulations of the noise assumption, in the case of binary classification. One of those formulations is similar to the following equation:

$$\mathbb{P}\left\{g(X) \neq g^*_{k,l}(X)\right\} \leqslant \beta(L_{k,l}(g) - L^*_{k,l})^\alpha, \tag{8.16}$$

where $\beta = (B^{1-\alpha})/(\epsilon(1-\alpha)^{1-\alpha}\alpha^{\alpha})$. The proof only differs slightly from that of (Bousquet et al., 2003, page 202 therein), as it features implications of 8.10.

Set $\beta_0 = \beta(p_k + p_l)^{\alpha}$, Equation (8.16) implies, for any $f \in \mathcal{F}_{k,l}$, that, with $T(f) = \sqrt{\beta_0} \cdot (Pf)^{\alpha/2}$:

$$\mathrm{Var}(f) \leqslant \mathbb{P}\left\{g(X) \neq g^*_{k,l}(X)\right\} \leqslant \beta(L_{k,l}(g) - L^*_{k,l})^{\alpha} = \beta(p_k + p_l)^{\alpha} \cdot (Pf)^{\alpha} = T^2(f). \tag{8.17}$$

The function $T(f)$ controls the variance of the elements in $\mathcal{F}_{k,l}$, and is used to reweights its instances before applying Talagrand's inequality.

The complexity of the proposed family of functions is controlled using the notion of Rademacher average, presented in Definition 2.10 of Chapter 2 in a the restricted setting of classifiers. Let $\mathcal{F}$ be a class of functions, its Rademacher average $R_n(\mathcal{F})$ is defined as:

$$R_n(\mathcal{F}) := \mathbb{E}_{\sigma} \sup_{f \in \mathcal{F}} \frac{1}{n} \left| \sum_{i=1}^{n} \sigma_i f(X_i, Y_i) \right|.$$

Introduce $\mathcal{F}^*_{k,l}$ as the star-hull of $\mathcal{F}_{k,l}$, *i.e.* $\mathcal{F}^*_{k,l} = \{\alpha f : \alpha \in [0,1], f \in \mathcal{F}_{k,l}\}$, we define two functions that characterize the properties of the problem of interest, and are required to apply Theorem 5.8 of Boucheron et al. (2005):

$$w(r) = \sup_{f \in \mathcal{F}^*_{k,l} : Pf \leqslant r} T(f) \qquad \text{and} \qquad \psi(r) = \mathbb{E}R_n\{f \in \mathcal{F}^*_{k,l} : T(f) \leqslant r\}. \tag{8.18}$$

Finally Theorem 5.8 of Boucheron et al. (2005) implies that, for all $\delta > 0$, with $r^*_0(\delta)$ the solution of:

$$r = 4\psi(w(r)) + 2w(r)\sqrt{\frac{2\log\frac{2}{\delta}}{n}} + \frac{16\log\frac{2}{\delta}}{3n}, \tag{8.19}$$

we have that, for any $k < l$, $(k,l) \in \{1, \ldots, K\}^2$ and any $n \in \mathbb{N}^*$, *w.p.* $\geqslant 1 - \delta$,

$$L_{k,l}(\widehat{g}_{k,l}) - L^*_{k,l} \leqslant 2\left(\inf_{g \in \mathcal{G}} L_{k,l}(g) - L^*_{k,l}\right) + \frac{r^*_0(\delta)}{p_k + p_l}.$$

Now, we can conclude by combining this result with properties of w and ψ, that originate from the noise assumption and the control on the complexity of $\mathcal{G}$, respectively. Assumption 8.9 states that the proposed class $\mathcal{G}$ is of VC-dimension V. Permanence properties of VC-classes of functions, see van der Vaart and Wellner (1996) (section 2.6.5), imply that $\mathcal{F}_{k,l}$ is also VC. It follows from P. Bartlett and Mendelson (2005) that:

$$\psi(r) \leqslant Cr\sqrt{\frac{V}{n}\log n},$$

where $C > 0$ is an universal constant.

Plugging this result into Equation (8.19) gives:

$$r^*_0(\delta) \leqslant \frac{2w(r^*_0(\delta))}{\sqrt{n}}\left[2C\sqrt{V\log n}, +\sqrt{2\log\frac{2}{\delta}}\right] + \frac{16\log\frac{2}{\delta}}{3n}.$$

Combining it with the definition of w in (8.18) and the control on the variance laid forth in (8.17) yields:

$$r^*_0(\delta) \leqslant [r^*_0(\delta)]^{\alpha/2}\,\frac{2\sqrt{\beta_0}}{\sqrt{n}}\left[2C\sqrt{V\log n} + \sqrt{2\log\frac{2}{\delta}}\right] + \frac{16\log\frac{2}{\delta}}{3n}. \tag{8.20}$$

Equation (8.20) is a variational inequality, and an upper bound on the solution can be derived directly from Zou et al. (2009) (Lemma 2). It writes:

$$r^*_0(\delta) \leqslant \max\left\{\left(\frac{16\beta_0}{n}\right)^{\frac{1}{2-\alpha}}\left[2C\sqrt{V\log n} + \sqrt{2\log(2/\delta)}\right]^{\frac{2}{2-\alpha}}, \frac{32\log(2/\delta)}{3n}\right\}.$$

Using the convexity of $x \mapsto x^{2/(2-\alpha)}$, the right-hand side of the above inequality can be upper-bounded, which leads to:

$$r_0^*(\delta) \leqslant 2 \cdot \max\left\{ \left(\frac{16\beta_0}{n}\right)^{\frac{1}{2-\alpha}} \left[\left(4C^2 V \log n\right)^{\frac{1}{2-\alpha}} + \left(2\log(2/\delta)\right)^{\frac{1}{2-\alpha}} \right], \frac{32\log(2/\delta)}{3n} \right\}. \quad (8.21)$$

Assumption 8.10 implies that $(p_k + p_l)^{-1} \leqslant 1/\epsilon$. Introducing $r^*(\delta) = (p_k + p_l)^{-1} r_0^*(\delta)$, Equation (8.21) combined with the definition of β_0 implies:

$$r^*(\delta) \leqslant 2 \cdot \max\left\{ \left(\frac{16\beta}{\epsilon^{2-2\alpha} n}\right)^{\frac{1}{2-\alpha}} \left[\left(4C^2 V \log n\right)^{\frac{1}{2-\alpha}} + \left(2\log(2/\delta)\right)^{\frac{1}{2-\alpha}} \right], \frac{32\log(2/\delta)}{3\epsilon n} \right\}. \quad (8.22)$$

Introduce $n_0(\delta, \alpha, \epsilon, B, V)$ as the lowest n such that the first term in the maximum in Equation (8.22) dominates the second term, it satisfies:

$$n_0^{\frac{1-\alpha}{2-\alpha}} \left[\left(4C^2 V \log(n_0)\right)^{\frac{1}{2-\alpha}} + \left(2\log(2/\delta)\right)^{\frac{1}{2-\alpha}} \right] \geqslant \frac{32\log(2/\delta)}{3\left[16\beta\epsilon^{\alpha}\right]^{\frac{1}{2-\alpha}}}, \quad (8.23)$$

and so does any $n \geqslant n_0$.

To conclude, we have proven that for any $\delta \in (0,1)$, for any $n \geqslant n_0(\delta, \alpha, \epsilon, B, V)$, we have that *w.p* $\geqslant 1 - \delta$,

$$L_{k,l}(\widehat{g}_{k,l}) - L_{k,l}^* \leqslant 2\left(\inf_{g \in \mathcal{G}} L_{k,l}(g) - L_{k,l}^*\right) + r^*(\delta),$$

with

$$r^*(\delta) = 2\left(\frac{16\beta}{n\epsilon^{2-2\alpha}}\right)^{\frac{1}{2-\alpha}} \left[\left(4C^2 V \log n\right)^{\frac{1}{2-\alpha}} + \left(2\log(2/\delta)\right)^{\frac{1}{2-\alpha}} \right]. \quad (8.24)$$

□

The following result builds on top of Lemma 8.11 to provide nonasymptotic bounds for the ranking risk $\mathcal{R}_P(\widehat{\sigma}_X)$ of the OVO ranking rule in the case where the loss function is $\mathbb{I}\{\sigma \neq \sigma'\}$, *i.e.* for the probability of error. Extension to any other loss function $d(.,\,.)$ is straightforward, insofar as we obviously have $d(\sigma_X^*,\, \widehat{\sigma}_X) \leqslant \max_{(\sigma,\sigma') \in \mathfrak{S}_K^2} d(\sigma, \sigma') \times \mathbb{I}\{\widehat{\sigma}_X \neq \sigma_X^*\}$ with probability one.

Theorem 8.12. *Suppose that Assumptions 8.8-8.10 are fulfilled. Then, for all $\delta \in (0,1)$, we have $\forall n \geqslant n_0(\delta, \alpha, \epsilon, B, V)$, with probability (w.p.) at least $1 - \delta$:*

$$\mathbb{P}\left\{\widehat{\sigma}_X \neq \sigma_X^* \mid \mathcal{D}_n\right\} \leqslant \frac{\beta}{\varepsilon}\left\{ \binom{K}{2} r_n^{\alpha}\left(\frac{\delta}{\binom{K}{2}}\right) + \sum_{k<l} 2\left(\inf_{g \in \mathcal{G}} L_{k,l}(g) - L_{k,l}^*\right)^{\alpha} \right\},$$

where X denotes a r.v. drawn from F, independent from the training data $\mathcal{D}_n$, $L_{k,l}^* = L_{k,l}(g_{k,l}^*)$, $\beta = \beta(\alpha, B)$ and with $h := h(B, a, \epsilon)$,

$$r_n(\delta) = 2\left(1/(nh)\right)^{\frac{1}{2-\alpha}} \times \left[\left(64C^2 V \log n\right)^{\frac{1}{2-\alpha}} + \left(32\log(2/\delta)\right)^{\frac{1}{2-\alpha}} \right].$$

Proof. Fix $\delta \in 0, 1$ and let $1 \leqslant k < l \leqslant K$. Assumption 8.8 implies that the Mammen-Tsybakov noise condition is fulfilled for the binary classification problem related to the pair (X, Y) given that $Y \in \{k,\, l\}$. When Assumptions 8.9-8.10 are also satisfied, a possibly fast rate bound for the risk excess of the empirical risk minimizer $\widehat{g}_{k,l}$ can be established, as stated in Lemma 8.11.

Observe that the probabilities appearing in this proof are conditional probabilities given the training sample $\mathcal{D}_n$ and, as a consequence, must be considered as random variables. However, to simplify notations, we omit to write the conditioning *w.r.t.* $\mathcal{D}_n$ explicitly. Eq. (2.11) in Chapter 2 gives an equivalent formulation of 8.8, that writes as:

$$\mathbb{P}\left\{\widehat{g}_{k,l}(X) \neq g_{k,l}^*(X) \mid Y \in \{k,\, l\}\right\} \leqslant \beta\left(L_{k,l}(\widehat{g}_{k,l}) - L_{k,l}^*\right)^{\alpha}, \quad (8.25)$$

with $\beta = B^{1-\alpha}/((1-\alpha)^{1-\alpha}\,\alpha^{\alpha})$. Denoting by $F_{k,l} = (p_k F_k + p_l F_l)/(p_k + p_l)$ the conditional distribution of X given that $Y \in \{k,\ l\}$, observe that:

$$\mathbb{P}\left\{\widehat{g}_{k,l}(X) \neq g^*_{k,l}(X) \mid Y \in \{k,\ l\}\right\} = \mathbb{E}_X\left[\frac{dF_{k,l}}{dF}(X) \times \mathbb{I}\left\{\widehat{g}_{k,l}(X) \neq g^*_{k,l}(X)\right\}\right]. \tag{8.26}$$

Under Assumption 8.10, we almost-surely have:

$$\frac{dF_{k,l}}{dF}(X) \geqslant \frac{\varepsilon}{p_k + p_l} \geqslant \varepsilon.$$

Hence, from (8.25) and Lemma 8.11, we get that

$$\frac{\varepsilon}{\beta}\mathbb{P}\left\{\widehat{g}_{k,l}(X) \neq g^*_{k,l}(X)\right\} \leqslant \left(L_{k,l}(\widehat{g}_{k,l}) - L^*_{k,l}\right)^{\alpha} \tag{8.27}$$

$$\leqslant 2\left(\inf_{g\in\mathcal{G}} L_{k,l}(g) - L^*_{k,l}\right)^{\alpha} + r_n^{\alpha}(\delta), \tag{8.28}$$

using Minkowski's inequality. Since

$$\bigcap_{k<l}\left\{\widehat{g}_{k,l}(X) = g^*_{k,l}(X)\right\} \subset \left\{\sigma^*_X = \widehat{\sigma}_X\right\},$$

with probability one, combining the bound above with the union bound gives that, for all $\delta \in (0,1)$, *w.p.* $\geqslant 1-\delta$:

$$\begin{aligned}\mathbb{P}\left\{\sigma^*_X \neq \widehat{\sigma}_X\right\} &\leqslant \sum_{k<l}\mathbb{P}\left\{\widehat{g}_{k,l}(X) \neq g^*_{k,l}(X)\right\} \\ &\leqslant \frac{\beta}{\varepsilon}\left\{\binom{K}{2} r_n^{\alpha}\left(\frac{\delta}{\binom{K}{2}}\right) + \sum_{k<l} 2\left(\inf_{g\in\mathcal{G}} L_{k,l}(g) - L^*_{k,l}\right)^{\alpha}\right\}.\end{aligned}$$

□

Hence, for the RMR problem related to the partially observed BTLP model detailed in subsection 8.3.1, the rate bound achieved by the OVO ranking rule in Theorem 8.12 is of order $n^{-\alpha/(2-\alpha)}$, ignoring the bias term and the logarithmic factors. In the case $\alpha = 1$, it is exactly the same rate as that attained by minimizers of the ranking risk in the standard RMR setup, as stated in Proposition 7 in Korba et al. (2017). Whereas situations where the OVO multi-class classification may possibly lead to 'inconsistencies' (*i.e.* where the binary relationship $\to_X$ is not transitive) have been exhibited many times in the literature, no probability bound for the excess of classification risk of the general OVO classifier, built from ERM applied to all binary subproblems, is documented to the best our knowledge. Hence, attention should be paid to the fact that, as a by-product of the argument of Theorem 8.12's proof, generalization bounds for the OVO classifier:

$$\overline{g}(X) = \widehat{\sigma}_X^{-1}(1),$$

can be established, as stated in Corollary 8.14 below. More generally, the statistical performance of the label ranking rule $\widehat{\sigma}_x$ produced by the method described in subsection 8.3.2 can be assessed for other risks. For instance, rather than just comparing the true label Y assigned to X to the label $\widehat{\sigma}_X^{-1}(1)$ ranked first, as in OVO classification approach, one could consider $\ell_k(Y,\ \widehat{\sigma}_X)$, with $\ell_k(y,\ \sigma) = \mathbb{I}\{y \notin \{\sigma^{-1}(1),\ \ldots,\ \sigma^{-1}(k)\}\}$ for all $(y,\sigma) \in \{1,\ \ldots,\ K\} \times \mathfrak{S}_K$, equal to 1 when Y does not appear in the top k list and to 0 otherwise, where k is fixed in $\{1,\ \ldots,\ K\}$. For any ranking rule $s : \mathbb{R}^q \to \mathfrak{S}_K$, the corresponding risk is then:

$$W_k(s) = \mathbb{E}[\ell_k(Y,\ s(X))]. \tag{8.29}$$

Set $W_k^* = \min_s W_k(s)$, where the minimum is taken over the set of all possible ranking rules s. The argument leading to Theorem 10 can be adapted to prove a rate bound for the risk excess of the OVO ranking rule $\sigma^*_{\cdot} : x \in \mathbb{R}^q \mapsto \sigma^*_x$.

Proposition 8.13. *Let $k \in \{1, \ldots, K\}$ be fixed. Then:*

$$W_k^* = W_k(\sigma_\cdot^*).$$

Suppose in addition that Assumptions 8.8-8.10 are fulfilled. Then, for all $\delta \in (0,1)$, we have $\forall n \geqslant 1$ w.p. $\geqslant 1-\delta$:

$$W_k(\widehat{\sigma}_\cdot) - W_k^* \leqslant \frac{\beta}{\varepsilon}\binom{K}{k} k(K-k) \times \left(r_n^\alpha \left(\frac{\delta}{\binom{K}{2}} \right) + 2 \cdot \max_{m \neq l} \left(\inf_{g \in \mathcal{G}} L_{l,m}(g) - L_{l,m}^* \right)^\alpha \right).$$

Proof. Let us first show that $W_k^* = W_k(\sigma_\cdot^*)$.

For any ranking rule s and all $x \in \mathbb{R}^q$, we define:

$$\mathrm{Top}_k(s(x)) = \{s(X)^{-1}(1), \ldots, s(X)^{-1}(k)\},$$

and also set $\mathrm{Top}_k^*(x) = \mathrm{Top}_k(\sigma_x^*)$. Indeed, for any ranking rule s, we can write:

$$W_k(s) = \mathbb{E}\left[\mathbb{E}\left[\ell_k(Y,\ s(X)) \mid X\right]\right],$$

and we almost-surely have:

$$\mathbb{E}\left[\ell_k(Y,\ s(X)) \mid X\right] = \sum_{l=1}^{K} \eta_l(X) \mathbb{I}\{l \notin \mathrm{Top}_k(s(X))\}. \tag{8.30}$$

As σ_x^* is defined through (8.3), one easily sees that the quantity (8.30) is minimum for any ranking rule $s(x)$ *s.t.*

$$\mathrm{Top}_k(s(X)) = \mathrm{Top}_k^*(X). \tag{8.31}$$

Hence, the collection of optimal ranking rules regarding the risk (8.29) coincides with the set of ranking rules such that (8.31) holds true with probability one. Observe that, with probability one,

$$\mathbb{I}\left\{Y \notin \mathrm{Top}_k(s(X))\right\} - \mathbb{I}\left\{Y \notin \mathrm{Top}_k^*(X)\right\} \leqslant \mathbb{I}\left\{\mathrm{Top}_k^*(X) \neq \mathrm{Top}_k(s(X))\right\},$$

for any ranking rule $s(x)$, so that:

$$W_k(s) - W_k^* \leqslant \mathbb{P}\left\{\mathrm{Top}_k(s(X)) \neq \mathrm{Top}_k^*(X)\right\}.$$

In addition, notice that:

$$\begin{aligned}
W_k(\widehat{\sigma}_X) - W_k^* &\leqslant \mathbb{P}\left\{\mathrm{Top}_k^*(X) \neq \mathrm{Top}_k(\widehat{\sigma}_X)\right\} \\
&= \sum_{\mathcal{L} \subset \mathcal{Y}:\ \#\mathcal{L}=k} \mathbb{P}\left\{\mathrm{Top}_k^*(X) = \mathcal{L},\ \mathrm{Top}_k^*(X) \neq \mathrm{Top}_k(\widehat{\sigma}_X)\right\}, \\
&\leqslant \sum_{\mathcal{L} \subset \mathcal{Y}:\ \#\mathcal{L}=k} \sum_{l \in \mathcal{L},\ m \notin \mathcal{L}} \mathbb{P}\left\{\widehat{g}_{l,m}(X) \neq g_{l,m}^*(X)\right\}, \\
&\leqslant \frac{\beta}{\varepsilon}\binom{K}{k} k(K-k) \times \left(r_n^\alpha \left(\frac{\delta}{\binom{K}{2}} \right) + 2 \cdot \max_{m \neq l} \left(\inf_{g \in \mathcal{G}} L_{l,m}(g) - L_{l,m}^* \right)^\alpha \right),
\end{aligned}$$

using (8.27). □

As stated above, since we have $W_1(s) = R(g_s)$ where $g_s : x \mapsto (s(x))^{-1}(1)$ for any label ranking rule $s(x)$, in the case $k = 1$ the result above provides a generalization bound for the excess of misclassification risk of the OVO classifier $\overline{g}(x) = \widehat{\sigma}_x^{-1}(1)$, that we provide in Corollary 8.14 below.

Corollary 8.14. *Suppose that Assumptions 8.8-8.10 are fulfilled. Then, for all $\delta \in (0,1)$, we have $\forall n \geqslant n_0(\delta, \alpha, \epsilon, B, V)$, w.p. $\geqslant 1-\delta$:*

$$L(\overline{g}) - L^* \leqslant \frac{\beta}{\varepsilon} K(K-1) \times \left(r_n^\alpha \left(\frac{\delta}{\binom{K}{2}} \right) + 2 \cdot \max_{k \neq l} \left(\inf_{g \in \mathcal{G}} L_{k,l}(g) - L_{k,l}^* \right)^\alpha \right).$$

Proof. It suffices to observe that we almost-surely have:

$$\begin{aligned} L(\bar{g}) - L^* &= \mathbb{E}_X \left[|\eta(X) - 1/2| \times \mathbb{I}\{\bar{g}(X) \neq g^*(X)\} \right] \\ &\leqslant \mathbb{P}\left\{\bar{g}(X) \neq g^*(X)\right\} = \mathbb{P}\left\{\widehat{\sigma}_X^{-1}(1) \neq \sigma_X^{*-1}(1)\right\} \\ &\leqslant \mathbb{P}\left\{\widehat{\sigma}_X \neq \sigma_X^*\right\} \end{aligned}$$

and to apply next the bound stated in Theorem 8.12. □

8.4 Experimental Results

This section first illustrates the results of Theorem 8.12 using simulated datasets, of which distributions satisfy Assumption 8.8, for certain values of the noise parameter α, highlighting the impact/relevance of this condition. In the experiments based on real data next displayed, the OVO approach to top-k classification, *cf* Eq. (8.29), is shown to surpass rankings that rely on the scores output by multiclass classification algorithms.

8.4.1 Synthetic Data Experiments

In this subsection, we illustrate the fast bounds proposed in this chapter by selecting and generating data from very specific distributions. Introduce the function h_α, which for any $\alpha \in [0, 1]$:

$$h_\alpha(x) = \frac{1}{2} + \frac{1}{2}\epsilon(x)|2x - 1|^{\frac{1-\alpha}{\alpha}},$$

where $\epsilon(x) = 2\mathbb{I}\{2x > 1\} - 1$. It has good properties with regard to the usual Mammen-Tsybakov noise condition introduced in Boucheron et al. (2005). We define a warped version h_{α,x_0} of this function h_α such that:

$$h_{\alpha,x_0}(x) = \begin{cases} h_{\alpha,x_0} = h_\alpha\left(\frac{x}{2x_0}\right) & \text{if } x < x_0, \\ h_{\alpha,x_0} = h_\alpha\left(\frac{1}{2} + \frac{x - x_0}{2(1-x_0)}\right) & \text{if } x \geqslant x_0. \end{cases}$$

We use this function to define the η_k's by recursion, and assume that X follows a uniform distribution on the interval $[0, 1]$. Introduce a depth parameter D and assume that the variable Y belong to $K = 2^{D+1}$ classes. Let $b_2^{(d)}(k)$ describe the decomposition in base 2 of the value k, *i.e.* $k = \sum_{d=0}^{D} 2^{b_2^{(d)}(k)}$. We introduce an evenly spaced grid over $[0, 1]$ as the set of values

$$x_{(d,k)} = \sum_{d=1}^{D} 2^{-b_2^{(d)}(k)},$$

for $k \in \{0, 2^{D+1}\}$. Finally, we set:

$$\eta_k(x) = \prod_{d=0}^{D} h_{\alpha,x_{(d,k)}}(x).$$

By varying the parameter α, one can set the classification problems to be more complicated or more easy. If α is close to 1, the problems are very simple. If α is close to 0, the problems are more arduous.

To implement the procedure described in Figure 8.1, we learn decision stumps in $[0, 1]$, *i.e.* we optimize over the family of functions $\mathcal{G} = \{g_{s,\epsilon} \mid s \in [0, 1], \epsilon \in \{-1, +1\}\}$, where, for any $x \in [0, 1]$:

$$g_{s,\epsilon}(x) = 2\mathbb{I}\{(x - s)\epsilon \geqslant 0\} - 1.$$

Figure 8.3, Figure 8.4 and Figure 8.5 represent boxplots obtained with 100 independent estimations on 1000 test points. Precisely, Figure 8.3 represents the number of cycles in predicted permutations, Figure 8.4 the average miss probability $\mathbb{P}\{\widehat{\sigma}_X \neq \sigma_X^*\}$ for the problem of predicting permutations,

and Figure 8.5 the average Kendall distance $\mathbb{E}\left[d_\tau(\hat{\sigma}_X, \sigma_X^*)\right]$ between predictions and ground truths. These quantities are represented as a function of the number of learning points n with:

$$n \in \bigcup_{i \in \{1,2,3,4\}} \{10^i, 3 \times 10^i\} \cup \{10^5\}.$$

One sees that learning is fast when α is close to 1, as expected. Figure 8.5 shows that the average Kendall's τ distance decreases quickly when α is close to 1, as does the the proportion of cycles in predictions, see Figure 8.3. On the other hand, due to the difficulty of predicting a complete permutation, the influence of the noise parameter on the evolution of the probability of error when n grows is more subtle, see Figure 8.4.

8.4.2 Real Data Experiments

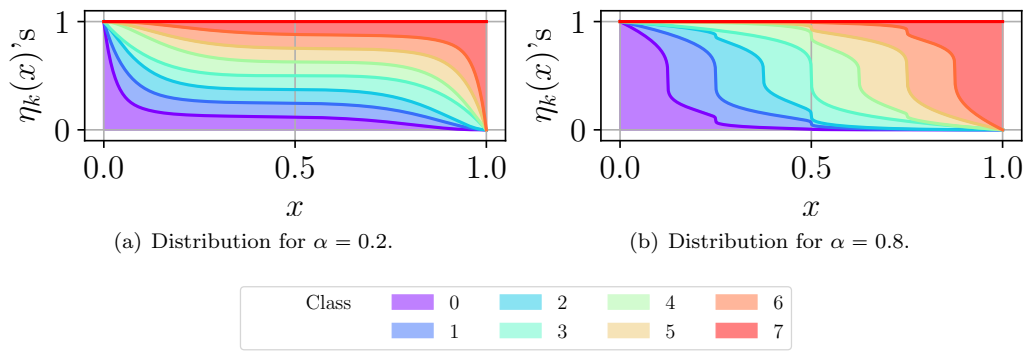
The MNIST dataset is composed of 28×28 grayscale images of digits and labels being the value of the digits. In this experiment, we learn to predict the value of the digit between $K = 10$ classes corresponding to digits between 0 and 9. The dataset contains $60,000$ images for training and $10,000$ images for testing, all equally distributed within the classes. This dataset has been praised for its accessibility, but was recently criticized for being too easy, which led to the introduction of the dataset Fashion-MNIST, see Xiao et al. (2017). It has the same structure as MNIST, with regard to train and test splits, number of classes and and image size. It consists in classifying types of clothing apparel, *e.g.* dress, coat and sandals, and is harder to classify than MNIST.

Our experiments aim to show that the OVO approach for top-k classification, *cf* Eq. (8.29), can surpass rankings relying on the scores output by multiclass classification algorithms. For that matter, we evaluated the performances of both approaches using a logistic regression to solve binary classification in the OVO case and multiclass classification in the other. For that matter, we relied on the implementations provided by the `python` package `scikit-learn`, specifically the `LogisticRegressionCV` class. The dimensionality of the data was reduced using standard PCA with enough components to retain 95% of the variance for both datasets, which makes for 153 components for MNIST and 187 components for Fashion-MNIST.

Results are summarized in Table 8.1. They show that the OVO approach performs better than the logistic regression for the top-1 accuracy, *i.e.* classification accuracy, as well as for the top-5 accuracy. While the OVO approach requires us to train $K(K-1)/2 = 45$ models, those are trained with less data and output values. Both approaches end up requiring a similar amount of time to be trained.

8.5 Conclusion

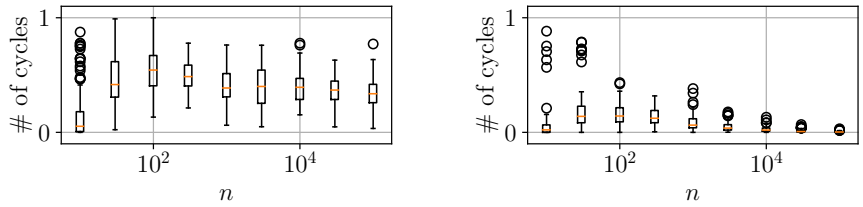
In this chapter, a statistical problem halfway between multiclass classification and posterior probability estimation, referred to as *label ranking* here, was considered. The goal was to design a method to rank, for any test observation X, all the labels y that can be possibly assigned to it by



(a) Distribution for $\alpha = 0.2$.

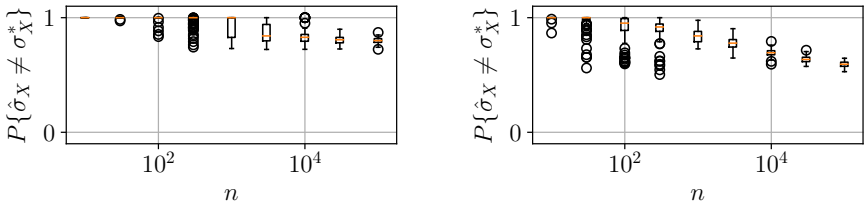
(b) Distribution for $\alpha = 0.8$.

Figure 8.2: Representation of the proportion of each class k over $[0, 1]$.



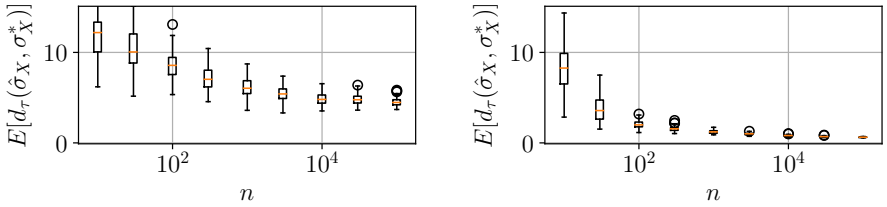
(a) Dynamics for $\alpha = 0.2$. (b) Dynamics for $\alpha = 0.8$

Figure 8.3: Boxplot of 100 independent estimations of the proportion of predictions with cycles as a function of n.



(a) Dynamics for $\alpha = 0.2$. (b) Dynamics for $\alpha = 0.8$

Figure 8.4: Boxplot of 100 independent estimations of $\mathbb{P}\{\widehat{\sigma}_X \neq \sigma_X^*\}$ as function of n.



(a) Dynamics for $\alpha = 0.2$. (b) Dynamics for $\alpha = 0.8$

Figure 8.5: Boxplot of 100 independent estimations of $\mathbb{E}\left[d_\tau(\widehat{\sigma}_X, \sigma_X^*)\right]$ as function of n.

decreasing order of magnitude of the (unknown) posterior probability $\mathbb{P}\{Y = y \mid X\}$. Formulated as a specific ranking median regression problem with incomplete observations, this problem was shown to have a solution that takes the form of a Copeland score, involving pairwise comparisons only. Based on this crucial observation, it was proved that the OVO procedure for multiclass classification permits to build, from training classification/labeled data, the optimal ranking with high probability, under appropriate hypotheses. This was also empirically supported by numerical experiments. Remarkably, the analysis carried out here incidentally provided a rate bound for the OVO classifier.

The analysis presented here advocates for a different method than usual class probabilities to predict an ordered list of the most likely labels for some observation. As such, it fits into our larger effort directed at finding more suitable loss functions for specific problems in biometrics, which increases the reliability of biometric systems. A flaw of said systems is their impaired generalization when deployed on different statistical populations than that used for training. In that regard, some authors believe that automatic facial recognition might suffer from an other-race effect, which refers to the better capacity of humans to distinguish faces of their own ethnicity, rather than those of others (Furl et al., 2002). That effect is usually explained by a more important exposure to faces of our own ethnicity. It suggests that increasing the representation of some populations during training process might help adapting the biometric system to specific

Dataset	Model	Top-1	Top-5	Fit time
MNIST	LogReg	0.924	0.995	50 min
	OVO	0.943	0.997	40 min
Fashion -MNIST	LogReg	0.857	0.997	35 min
	OVO	0.863	0.997	60 min

Table 8.1: Top-k performance. The last column is the time to fit the model.

test distributions, which is the topic of the next chapter.

Chapter 9

Selection Bias Correction

Summary: We consider statistical learning problems, when the distribution P' of the training observations $Z'_1, ..., Z'_n$ differs from the test distribution P but is still defined on the same measurable space as P and dominates it. This problem often arises in biometrics, as companies often have large databases, but must tailor their systems to specific populations. Our mathematization of 1:1 identification presented in Part II does not cover that topic. In the unrealistic case where the likelihood ratio $\Phi(z) = dP/dP'(z)$ is known, one may straightforwardly extend Empirical Risk Minimization (ERM) to this transfer learning setup using Importance Sampling (IS), *i.e.* by minimizing a weighted version of the risk functional computed on the 'biased' training data Z'_i with weights $\Phi(Z'_i)$. The importance function $\Phi(z)$ is generally unknown in practice, but frequently takes — *e.g.* when learning: with class imbalance, in a stratified population or with only positive and unlabeled data — a simple form and is directly estimated from the Z'_i's and some auxiliary information on P. Using the same tools as for finite-sample generalization bounds for binary classification (Chapter 2), we then prove that the usual generalization capacity of ERM is preserved when plugging the resulting estimates of the $\Phi(Z'_i)$'s into the weighted empirical risk. We provide experiments, that show on ImageNet — an image dataset based on the hierarchical lexical database WordNet — that correcting bias on high-level categories leads to significant performance improvements for the classification task.

9.1 Introduction

Prediction problems are of major importance in statistical learning. The main paradigm of predictive learning is *Empirical Risk Minimization* (ERM in abbreviated form), see *e.g.* Devroye et al. (1996). In the standard setup, Z is a random variable (*r.v.* in short) that takes its values in a feature space $\mathcal{Z}$ with distribution P, Θ is a parameter space and $\ell : \Theta \times \mathcal{Z} \to \mathbb{R}_+$ is a (measurable) loss function. The risk is then defined by: $\forall \theta \in \Theta$,

$$\mathcal{R}_P(\theta) = \mathbb{E}_P\left[\ell(\theta, Z)\right], \tag{9.1}$$

and more generally for any measure Q on $\mathcal{Z}$: $\mathcal{R}_Q(\theta) = \int_{\mathcal{Z}} \ell(\theta, z) dQ(z)$. In most practical situations, the distribution P involved in the definition of the risk is unknown and learning is based on the sole observation of an independent and identically distributed (*i.i.d.*) sample $Z_1, \ldots, Z_n$ drawn from P and the risk (9.1) must be replaced by an empirical counterpart (or a possibly smoothed/penalized version of it), typically:

$$\widehat{\mathcal{R}}_P(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta, Z_i) = \mathcal{R}_{\widehat{P}_n}(\theta), \tag{9.2}$$

where $\widehat{P}_n = (1/n)\sum_{i=1}^n \delta_{Z_i}$ is the empirical measure of P and δ_z denotes the Dirac measure at any point z. With the design of successful algorithms such as neural networks, support vector machines

or boosting methods to perform ERM, the practice of predictive learning has recently received a significant attention and is now supported by a sound theory based on results in empirical process theory. The performance of minimizers of (9.2) can be indeed studied by means of concentration inequalities, quantifying the fluctuations of the maximal deviations $\sup_{\theta\in\Theta}|\widehat{\mathcal{R}}_P(\theta)-\mathcal{R}_P(\theta)|$ under various complexity assumptions for the functional class $\mathcal{F}=\{\ell(\theta,\,\cdot):\ \theta\in\Theta\}$ (*e.g.* VC dimension, metric entropies, Rademacher averages), see Boucheron et al. (2013) for instance. Although, in the Big Data era, the availability of massive digitized information to train predictive rules is an undeniable opportunity for the widespread deployment of machine-learning solutions, the poor control of the data acquisition process one is confronted with in many applications puts practitioners at risk of jeopardizing the generalization ability of the rules produced by the algorithms implemented. Bias selection issues in machine-learning are now the subject of much attention in the literature, see Bolukbasi et al. (2016), Zhao et al. (2017), Hendricks et al. (2018), Liu et al. (2016) or Huang et al. (2006). In the context of face analysis, a research area including a broad range of applications such as face detection, face recognition or face attribute detection, machine learning algorithms trained with baised training data, *e.g.* in terms of gender or ethnicity, raise concerns about fairness in machine learning. Unfair algorithms may induce systemic undesired disadvantages for specific social groups, see Das et al. (2018) for further details. Several examples of bias in deep learning based face recognition systems are discussed in Nagpal et al. (2019).

Throughout the present chapter, we consider the case where the *i.i.d.* sample $Z_1',\,\ldots,\,Z_n'$ available for training is not drawn from P but from another distribution P', with respect to which P is absolutely continuous, and the goal pursued is to set theoretical grounds for the application of ideas behind Importance Sampling (IS in short) methodology to extend the ERM approach to this learning setup. This setup often arises in biometrics. Indeed, companies often have large databases, but must tailor their systems to specific populations. We highlight that the problem under study is a very particular case of *Transfer Learning* (see *e.g.* Pan and Yang (2010), Ben-David et al. (2010), Storkey (2009) and Redko et al. (2019)), a research area currently receiving much attention in the literature and encompassing general situations where the information/knowledge one would like to transfer may take a form in the *target* space very different from that in the *source* space (referred to as *domain adaptation*).

Weighted ERM (WERM). In this chapter, we investigate conditions guaranteeing that values for the parameter θ that nearly minimize (9.1) can be obtained through minimization of a weighted version of the empirical risk based on the Z_i''s, namely:

$$\widetilde{\mathcal{R}}_{w,n}(\theta)=\mathcal{R}_{\widetilde{P}_{w,n}}(\theta), \tag{9.3}$$

where $\widetilde{P}_{w,n}=(1/n)\sum_{i=1}^n w_i\delta_{Z_i'}$ and $w=(w_1,\,\ldots,\,w_n)\in\mathbb{R}_+^n$ is a certain weight vector. Of course, ideal weights w^* are given by the likelihood function $\Phi(z)=(dP/dP')(z)$: $w_i^*=\Phi(Z_i')$ for $i\in\{1,\,\ldots,\,n\}$. In this case, the quantity (9.3) is obviously an unbiased estimate of the true risk (9.1):

$$\mathbb{E}_{P'}\left[\mathcal{R}_{\widetilde{P}_{w^*,n}}(\theta)\right]=\mathcal{R}_P(\theta), \tag{9.4}$$

and generalization bounds for the $\mathcal{R}_P$-risk excess of minimizers of $\widetilde{\mathcal{R}}_{w^*,n}$ can be directly established by studying the concentration properties of the empirical process related to the Z_i''s and the class of functions $\{\Phi(\cdot)\ell(\theta,\,\cdot):\ \theta\in\Theta\}$ (see section 9.2 below). However, the *importance function* Φ is unknown in general, just like distribution P. It is the major purpose of this chapter to show that, in far from uncommon situations, the (ideal) weights w_i^* can be estimated from the Z_i's combined with auxiliary information on the target population P. As shall be seen below, such favorable cases include in particular classification problems where class probabilities in the test stage differ from those in the training step, risk minimization in stratified populations (see Bekker et al. (2019)), with strata statistically represented in a different manner in the test and training populations or even positive-unlabeled learning (PU-learning, see *e.g.* du Plessis et al. (2014)). In each of these cases, we show that the stochastic process obtained by plugging the weight estimates in the weighted empirical risk functional (9.3) is much more complex than a simple empirical process (*i.e.* a collection of *i.i.d.* averages) but can be however studied by means of *linearization techniques*, in the spirit of the ERM extensions established in Clémençon

et al. (2008) or Clémençon and Vayatis (2008). Learning rate bounds for minimizers of the corresponding risk estimate are proved and, beyond these theoretical guarantees, the performance of the weighted ERM approach is supported by convincing numerical results.

The chapter is structured as follows. In section 9.2, the ideal case where the importance function Φ is known is preliminarily considered and a first basic example where the optimal weights can be easily inferred and plugged into the risk without deteriorating the learning rate is discussed. The main results of the chapter are stated in section 9.3, which shows that the methodology promoted can be applied to two important problems in practice, risk minimization in stratified populations and PU-learning, with generalization guarantees. Illustrative numerical experiments are displayed in section 9.5, while some concluding remarks are collected in section 9.6.

9.2 Importance Sampling - Risk Minimization with Biased Data

Here and throughout, the indicator function of any event $\mathcal{E}$ is denoted by $\mathbb{I}\{\mathcal{E}\}$, the sup norm of any bounded function $h : \mathcal{Z} \to \mathbb{R}$ by $||h||_\infty$. We place ourselves in the framework of statistical learning based on biased training data previously introduced. To begin with, we consider the unrealistic situation where the importance function Φ is known, insofar as we shall subsequently develop techniques aiming at mimicking the minimization of the ideally weighted empirical risk:

$$\widetilde{\mathcal{R}}_{w*,n}(\theta) = \frac{1}{n}\sum_{i=1}^{n} w_i^* \ell(\theta, Z_i'), \tag{9.5}$$

namely the (unbiased) Importance Sampling estimator of (9.1) based on the instrumental data $Z_1', \ldots, Z_n'$. The following result describes the performance of minimizers $\widetilde{\theta}_n^*$ of (9.5). Since the goal of this chapter is to promote the main ideas of the approach rather than to state results with the highest level of generality, we assume throughout the chapter for simplicity that ℓ and Φ are both bounded functions. For $\sigma_1, \ldots, \sigma_n$ independent Rademacher random variables (*i.e.* symmetric $\{-1, 1\}$-valued *r.v.*'s), independent from the Z_i''s, we define the Rademacher average associated to any class of function $\mathcal{F}_0$ as:

$$R_n'(\mathcal{F}_0) := \mathbb{E}_\sigma\left[\sup_{f\in\mathcal{F}_0}(1/n)\left|\sum_{i=1}^{n}\sigma_i \cdot f(Z_i')\right|\right].$$

This quantity can be bounded by metric entropy methods under appropriate complexity assumptions on the class $\mathcal{F}_0$, it is for instance of order $1/\sqrt{n}$ when $\mathcal{F}_0$ is a VC-major class with finite VC dimension, see *e.g.* Boucheron et al. (2005). In particular, we are interested in the family $\mathcal{F} := \{z \in \mathcal{Z} \mapsto \ell(\theta, z) : \theta \in \Theta\}$.

Lemma 9.1. *We have $\forall n \geqslant 1$, with probability (w.p) at least $1-\delta$:*

$$\mathcal{R}_P(\widetilde{\theta}_n^*) - \min_{\theta\in\Theta}\mathcal{R}_P(\theta) \leqslant 4||\Phi||_\infty \mathbb{E}\left[R_n'(\mathcal{F})\right] + 2||\Phi||_\infty \sup_{(\theta,z)\in\Theta\times\mathcal{Z}} \ell(\theta, z)\sqrt{\frac{2\log(1/\delta)}{n}}.$$

Proof. Let $\delta \in (0, 1)$. Applying the classic maximal deviation bound stated in Theorem 3.2 of Boucheron et al. (2005) to the bounded class $\mathcal{K} = \{z \in \mathcal{Z} \mapsto \Phi(z)l(\theta, z) : \ \theta \in \Theta\}$, we obtain that, *w.p* $\geqslant 1-\delta$:

$$\sup_{\theta\in\Theta}\left|\widetilde{\mathcal{R}}_{w*,n}(\theta) - \mathbb{E}\left[\widetilde{\mathcal{R}}_{w*,n}(\theta)\right]\right| \leqslant 2\mathbb{E}\left[R_n'(\mathcal{K})\right] + ||\Phi||_\infty \sup_{(\theta,z)\in\Theta\times\mathcal{Z}} |\ell(\theta, z)|\sqrt{\frac{2\log(1/\delta)}{n}}.$$

In addition, by virtue of the contraction principle, we have $R_n'(\mathcal{K}) \leqslant ||\Phi||_\infty R_n'(\mathcal{F})$ almost-surely. The desired result can be thus deduced from the bound above combined with the classic bound:

$$\mathcal{R}_P(\tilde{\theta}_n^*) - \min_{\theta\in\Theta}\mathcal{R}_P(\theta) \leqslant 2\sup_{\theta\in\Theta}\left|\widetilde{\mathcal{R}}_{w*,n}(\theta) - \mathbb{E}\left[\widetilde{\mathcal{R}}_{w*,n}(\theta)\right]\right|.$$

□

Of course, when $P' = P$, we have $\Phi = 1$ and the bound stated above simply describes the performance of standard empirical risk minimizers. The proof is based on the standard bound:

$$\mathcal{R}_P(\widetilde{\theta}_n^*) - \min_{\theta\in\Theta} \mathcal{R}_P(\theta) \leqslant 2 \sup_{\theta\in\Theta} \left| \widetilde{\mathcal{R}}_{w*,n}(\theta) - \mathbb{E}\left[\widetilde{\mathcal{R}}_{w*,n}(\theta)\right] \right|,$$

combined with basic concentration results for empirical processes. Of course, the importance function Φ is generally unknown and must be estimated in practice. As illustrated by the elementary example below (related to binary classification, in the situation where the probability of occurence of a positive instance significantly differs in the training and test stages), in certain statistical learning problems with biased training distribution, Φ takes a simplistic form and can be easily estimated from the Z_i''s combined with auxiliary information on P.

Binary classification with varying class probabilities. The flagship problem in supervised learning corresponds to the simplest situation, where $Z = (X, Y)$, Y being a binary variable valued in $\{-1, +1\}$ say, and the *r.v.* X takes its values in a measurable space $\mathcal{X}$ and models some information hopefully useful to predict Y. The parameter space Θ is a set $\mathcal{G}$ of measurable mappings (*i.e.* classifiers) $g : \mathcal{X} \to \{-1,\ +1\}$ and the loss function is given by $\ell(g,\ (x, y)) = \mathbb{I}\{g(x) \neq y\}$ for all g in $\mathcal{G}$ and any $(x, y) \in \mathcal{X} \times \{-1,\ +1\}$. The distribution P of the random pair (X, Y) can be either described by X's marginal distribution F and the posterior probability $\eta(x) = \mathbb{P}\{Y = +1 \mid X = x\}$ or else by the triplet (p, F_+, F_-) where $p = \mathbb{P}\{Y = +1\}$ and $F_\sigma(dx)$ is X's conditional distribution given $Y = \sigma 1$ with $\sigma \in \{-,\ +\}$. It is very common that the fraction of positive instances in the training dataset is significantly lower than the rate p expected in the test stage, supposed to be known here. We thus consider the case where the distribution P' of the training data $(X_1', Y_1'),\ \ldots,\ (X_n', Y_n')$ is described by the triplet (p', F_+, F_-) with $p' < p$. The likelihood function takes the simple following form:

$$\Phi(x, y) = \mathbb{I}\{y = +1\}\frac{p}{p'} + \mathbb{I}\{y = -1\}\frac{1-p}{1-p'} := \phi(y),$$

which reveals that it depends on the label y solely, and the ideally weighted empirical risk process is:

$$\widetilde{\mathcal{R}}_{w*,n}(g) = \frac{p}{p'}\frac{1}{n}\sum_{i:Y_i'=1} \mathbb{I}\{g(X_i') = -1\} + \frac{1-p}{1-p'}\frac{1}{n}\sum_{i:Y_i'=-1} \mathbb{I}\{g(X_i') = +1\}. \tag{9.6}$$

In general the theoretical rate p' is unknown and one replaces (9.6) with:

$$\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) = \frac{p}{n_+'}\sum_{i:Y_i'=1} \mathbb{I}\{g(X_i') = -1\} + \frac{1-p}{n_-'}\sum_{i:Y_i'=-1} \mathbb{I}\{g(X_i') = +1\}, \tag{9.7}$$

where $n_+' = \sum_{i=1}^n \mathbb{I}\{Y_i' = +1\} = n - n_-'$, $\widehat{w}_i^* = \widehat{\phi}(Y_i')$ and $\widehat{\phi}(y) = \mathbb{I}\{y = +1\}np/n_+' + \mathbb{I}\{y = -1\}n(1-p)/n_-'$. The stochastic process above is not a standard empirical process but a collection of sums of two ratios of basic averages. However, the following result provides a uniform control of the deviations between the ideally weighted empirical risk and that obtained by plugging the empirical weights into the latter.

Lemma 9.2. *Let $\varepsilon \in (0,\ 1/2)$. Suppose that $p' \in (\varepsilon,\ 1-\varepsilon)$. For any $\delta \in (0, 1)$ and $n \in \mathbb{N}^*$, we have w.p $\geqslant 1 - \delta$:*

$$\sup_{g\in\mathcal{G}} \left| \widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) - \widetilde{\mathcal{R}}_{w*,n}(g) \right| \leqslant \frac{2}{\varepsilon^2}\sqrt{\frac{\log(2/\delta)}{2n}},$$

as soon as $n \geqslant 2\log(2/\delta)/\varepsilon^2$.

Proof. Apply twice the Taylor expansion:

$$\frac{1}{x} = \frac{1}{a} - \frac{x-a}{a^2} + \frac{(x-a)^2}{xa^2},$$

so as to get:

$$\begin{aligned}\frac{1}{n'_+/n} &= \frac{1}{p'} - \frac{n'_+/n - p'}{p'^2} + \frac{(n'_+/n - p')^2}{p'^2 n'_+/n},\\ \frac{1}{n'_-/n} &= \frac{1}{1-p'} - \frac{n'_-/n - 1 + p'}{(1-p')^2} + \frac{(n'_-/n - 1 + p')^2}{(1-p')^2 n'_-/n}.\end{aligned}$$

This yields the decomposition:

$$\begin{aligned}\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) - \widetilde{\mathcal{R}}_{w*,n}(g) = &- \frac{p}{p'^2}\left(\frac{n'_+}{n} - p'\right)\frac{1}{n}\sum_{i=1}^n \mathbb{I}\{g(X'_i) = -1,\ Y'_i = +1\}\\ &- \frac{1-p}{(1-p')^2}\left(\frac{n'_-}{n} - 1 + p'\right)\frac{1}{n}\sum_{i=1}^n \mathbb{I}\{g(X'_i) = +1,\ Y'_i = -1\}\\ &+ \frac{p(n'_+/n - p')^2}{p'^2 n'_+/n}\frac{1}{n}\sum_{i=1}^n \mathbb{I}\{g(X'_i) = -1,\ Y'_i = +1\}\\ &+ \frac{(1-p)(n'_-/n - 1 + p')^2}{(1-p')^2 n'_-/n}\frac{1}{n}\sum_{i=1}^n \mathbb{I}\{g(X'_i) = +1,\ Y'_i = -1\}.\end{aligned}$$

We deduce that:

$$\left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) - \widetilde{\mathcal{R}}_{w*,n}(g)\right| \leqslant \frac{|n'_+/n - p'|}{\varepsilon^2}\left(1 + |n'_+/n - p'|\left(\frac{p}{n'_+/n} + \frac{1-p}{1-n'_+/n}\right)\right).$$

By virtue of Hoeffding inequality, we obtain that, for any $\delta \in (0,1)$, we have $w.p \geqslant 1-\delta$:

$$\left|n'_+/n - p'\right| \leqslant \sqrt{\frac{\log(2/\delta)}{2n}},$$

so that, in particular, $\min\{n'_+/n,\ 1 - n'_+/n\} \geqslant \varepsilon - \sqrt{\log(2/\delta)/(2n)}$. This yields the desired result. □

Consequently, minimizing (9.7) nearly boils down to minimizing (9.6). Combining Lemmas 9.2 and 9.1, we immediately get the generalization bound stated in the result below.

Corollary 9.3. *Suppose that the hypotheses of Lemma 9.2 are fulfilled. Let $n \in \mathbb{N}^*$ and $\widetilde{g}_n$ be any minimizer of $\widetilde{\mathcal{R}}_{\widehat{w}*,n}$ over the class $\mathcal{G}$. We have w.p. $\geqslant 1-\delta$:*

$$\mathcal{R}_P(\widetilde{g}_n) - \inf_{g\in\mathcal{G}} \mathcal{R}_P(g) \leqslant \frac{2\max(p, 1-p)}{\varepsilon}\left(2\mathbb{E}[\mathfrak{R}'_n(\mathcal{G})] + \sqrt{\frac{2\log(2/\delta)}{n}}\right) + \frac{4}{\varepsilon^2}\sqrt{\frac{\log(4/\delta)}{2n}},$$

as soon as $n \geqslant 2\log(4/\delta)/\varepsilon^2$; where $\mathfrak{R}'_n(\mathcal{G}) = (1/n)\mathbb{E}_\sigma[\sup_{g\in\mathcal{G}} |\sum_{i=1}^n \sigma_i \mathbb{I}\{g(X'_i) \neq Y'_i\}|]$.

Proof. Observe first that $||\Phi||_\infty \leqslant \max(p,\ 1-p)/\varepsilon$ and

$$\mathcal{R}_P(\widetilde{g}_n) - \inf_{g\in\mathcal{G}} \mathcal{R}_P(g) \leqslant 2\sup_{g\in\mathcal{G}}\left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) - \widetilde{\mathcal{R}}_{w*,n}(g)\right| + 2\sup_{g\in\mathcal{G}}\left|\widetilde{\mathcal{R}}_{w*,n}(g) - \mathcal{R}_P(g)\right|.$$

The result then directly follows from the application of Lemmas 9.1-9.2 combined with the union bound. □

Hence, some side information (*i.e.* knowledge of parameter p) has permitted to weight the training data in order to build an empirical risk functional that approximates the target risk and to show that minimization of this risk estimate yields prediction rules with optimal (in the minimax sense) learning rates. The purpose of the subsequent analysis is to show that this remains true for more general problems. Observe in addition that the bound in Corollary 9.3 deteriorates as ε decays to zero: the method used here is not intended to solve the *few shot* learning problem, where almost no training data with positive labels is available (*i.e.* $p' \approx 0$). As shall be seen in subsection 9.3.2, alternative estimators of the importance function must be considered in this situation.

Remark 9.4. *Although the quantity* (9.7) *can be viewed as a cost-sensitive version of the empirical classification risk based on the* (X'_i, Y'_i)*'s (see e.g. Bach et al. (2006)), we point out that the goal pursued here is not to achieve an appropriate trade-off between type I and type II errors in the* P' *classification problem as in biometric applications for instance (i.e. optimization of the* (F_+, F_-)-ROC *curve at a specific point) but to transfer knowledge gained in analyzing the biased data drawn from* P' *to the classification problem related to distribution* P*.*

Related work. We point out that the natural idea of using weights in ERM problems that mimic those induced by the importance function has already been used in Sugiyama et al. (2007) for *covariate shift adaptation* problems (*i.e.* supervised situations, where the conditional distribution of the output given the input information is the same in the training and test domains), when, in contrast to the framework considered here, a test sample is additionally available (a method for estimating directly the importance function based on Kullback-Leibler divergence minimization is proposed, avoiding estimation of the test density). Importance sampling estimators have been also considered in Garcke and Vanck (2014) in the setup of *inductive transfer learning* (the tasks between source and target are different, regardless of the similarities between source and target domains), where the authors have proposed two methods to approximate the importance function, among which one is again based on minimizing the Kullback-Leibler divergence between the two distributions. In Cortes et al. (2008), the sample selection bias is assumed to be independent from the label, which is not true under our stratum-shift assumption or for the PU learning problem (see section 8.3). Lemma 9.1 assumes that the exact importance function is known, as does Cortes et al. (2010). The next section introduces new results for more realistic settings where it has to be learned from the data.

9.3 Weighted Empirical Risk Minimization - Generalization Guarantees

Through two important and generic examples, relevant for many applications, we show that the approach sketched above can be applied to general situations, where appropriate auxiliary information on the target distribution is available, with generalization guarantees. We work in this section under the condition that the loss function is bounded assuming $\sup_{(\theta,z)\in\Theta\times\mathcal{Z}} \ell(\theta, z) \leqslant L$.

9.3.1 Statistical Learning from Biased Data in a Stratified Population

A natural extension of the simplistic problem considered in section 9.2 is multiclass classification in a stratified population. The random labels Y and Y' are supposed to take their values in $\{1, \ldots, J\}$ say, with $J \geqslant 1$, and each labeled observation (X, Y) belongs to a certain random stratum S in $\{1, \ldots, K\}$ with $K \geqslant 1$. Again, the distribution P of a random element $Z = (X, Y, S)$ may be described by the parameters $\{(p_{j,k}, F_{j,k}) : 1 \leqslant j \leqslant J,\ 1 \leqslant k \leqslant K\}$ where $F_{j,k}$ is the conditional distribution of X given $(Y, S) = (j, k)$ and $p_{j,k} = \mathbb{P}_{(X,Y,S)\sim P}\{Y = j, S = k\}$. Then, we have:

$$dP(x, y, s) = \sum_{j=1}^{J} \sum_{k=1}^{K} \mathbb{I}\{y = j, s = k\} p_{j,k} dF_{j,k}(x),$$

and considering a distribution P' with $F_{j,k} = F'_{j,k}$ but possibly different class-stratum probabilities $p'_{j,k}$, the likelihood function becomes:

$$\frac{dP}{dP'}(x, y, s) = \sum_{j=1}^{J} \sum_{k=1}^{K} \frac{p_{j,k}}{p'_{j,k}} \mathbb{I}\{y = j, s = k\} := \phi(y, s).$$

A more general framework can actually encompass this specific setup by defining 'meta-strata' in $\{1, \ldots, J\} \times \{1, \ldots, K\}$. Strata may often correspond to categorical input features in practice. The formalism introduced below is more general and includes the example considered in the preceding section, where strata are defined by labels.

Learning from biased stratified data. Consider a general mixture model, where distributions P and P' are stratified over $K \geqslant 1$ strata. Namely, $Z = (X, S)$ and $Z' = (X', S')$ with auxiliary random variables S and S' (the strata) valued in $\{1, \ldots, K\}$. We place ourselves in a *stratum-shift* context, assuming that the conditional distribution of X given $S = k$ is the same as that of X' given $S' = k$, denoted by $F_k(dx)$, for any $k \in \{1, \ldots, K\}$. However, stratum probabilities $p_k = \mathbb{P}(S = k)$ and $p'_k = \mathbb{P}(S' = k)$ may possibly be different. In this setup, the likelihood function depends only on the strata and can be expressed in a very simple form, as follows:

$$\frac{dP}{dP'}(x, s) = \sum_{k=1}^{K} \mathbb{I}\{s = k\} \frac{p_k}{p'_k} := \phi(s).$$

In this case, the ideally weighted empirical risk writes:

$$\widetilde{\mathcal{R}}_{w*,n}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(\theta, Z'_i) \sum_{k=1}^{K} \mathbb{I}\{S'_i = k\} \frac{p_k}{p'_k}.$$

If the strata probabilities p_k's for the test distribution are known, an empirical counterpart of the ideal empirical risk above is obtained by simply plugging estimates of the p'_k's computed from the training data:

$$\widetilde{\mathcal{R}}_{\widehat{w}*,n}(\theta) = \sum_{i=1}^{n} \ell(\theta, Z'_i) \sum_{k=1}^{K} \mathbb{I}\{S'_i = k\} \frac{p_k}{n'_k}, \tag{9.8}$$

with $n'_k = \sum_{i=1}^{n} \mathbb{I}\{S'_i = k\}$, $\widehat{w}^*_i = \widehat{\phi}(S'_i)$ and $\widehat{\phi}(s) = \sum_{k=1}^{K} \mathbb{I}\{s = k\} n p_k / n'_k$.

A bound for the excess of risk associated to Eq. (9.8) is given in Theorem 9.6, that can be viewed as a generalization of Corollary 9.3. It relies on Lemma 9.1 and Lemma 9.5 below.

Lemma 9.5. *Let $\varepsilon \in (0,\ 1/2)$. Suppose that $p'_k \in (\varepsilon,\ 1-\varepsilon)$ for $k \in \{1,\ \ldots,\ K\}$. For any $\delta \in (0, 1)$ and $n \in \mathbb{N}^*$, we have w.p. $\geqslant 1 - \delta$:*

$$\sup_{\theta \in \Theta} \left| \widetilde{\mathcal{R}}_{\widehat{w}*,n}(\theta) - \widetilde{\mathcal{R}}_{w*,n}(\theta) \right| \leqslant \frac{2L}{\varepsilon^2} \sqrt{\frac{\log(2K/\delta)}{2n}},$$

as soon as $n \geqslant 2\log(2K/\delta)/\varepsilon^2$, where $L = \sup_{(\theta,z) \in \Theta \times \mathcal{Z}} \ell(\theta, z)$.

Proof. Apply the Taylor expansion

$$\frac{1}{x} = \frac{1}{a} - \frac{x - a}{a^2} + \frac{(x - a)^2}{xa^2},$$

so as to get for all $k \in \{1, \ldots, K\}$

$$\frac{1}{n'_k/n} = \frac{1}{p'_k} - \frac{n'_k/n - p'_k}{p'^2_k} + \frac{(n'_k/n - p'_k)^2}{p'^2_k n'_k/n}.$$

This yields the decomposition

$$\widetilde{\mathcal{R}}_{\widehat{w}*,n}(\theta) - \widetilde{\mathcal{R}}_{w*,n}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(\theta, Z'_i) \sum_{k=1}^{K} \mathbb{I}\{S'_i = k\} \left(-\frac{p_k}{p'^2_k} \left(\frac{n'_k}{n} - p'_k \right) + \frac{p_k (n'_k/n - p'_k)^2}{p'^2_k n'_k/n} \right).$$

We deduce that

$$\left| \widetilde{\mathcal{R}}_{\widehat{w}*,n}(\theta) - \widetilde{\mathcal{R}}_{w*,n}(\theta) \right| \leqslant \frac{L}{\varepsilon^2} \sum_{k=1}^{K} |n'_k/n - p'_k| p_k \left(1 + \frac{|n'_k/n - p'_k|}{n'_k/n} \right).$$

By virtue of Hoeffding inequality, we obtain that, for any $k \in \{1, \ldots, K\}$ and $\delta \in (0, 1)$, we have $w.p \geqslant 1 - \delta$:

$$\left| n'_k/n - p'_k \right| \leqslant \sqrt{\frac{\log(2/\delta)}{2n}},$$

so that, by a union bound, $\max_k \{n'_k/n\} \geqslant \varepsilon - \sqrt{\log(2K/\delta)/(2n)}$. This yields the desired result. □

Theorem 9.6. *Let $\varepsilon \in (0, 1/2)$ and assume that $p'_k \in (\varepsilon, 1-\varepsilon)$ for $k = 1, \dots, K$. Let $\widetilde{\theta}^*_n$ be any minimizer of $\widetilde{\mathcal{R}}_{\widehat{w}*,n}$ as defined in (9.8) over class Θ. For any $n \in \mathbb{N}^*$ and $\delta \in (0,1)$, we have w.p $\geqslant 1-\delta$:*

$$\mathcal{R}_P(\widetilde{\theta}^*_n) - \inf_{\theta\in\Theta} \mathcal{R}_P(\theta) \leqslant \frac{2\max_k p_k}{\varepsilon}\left(2\mathbb{E}[R'_n(\mathcal{F})] + L\sqrt{\frac{2\log(2/\delta)}{n}}\right) + \frac{4L}{\varepsilon^2}\sqrt{\frac{\log(4K/\delta)}{2n}},$$

as soon as $n \geqslant 2\log(4K/\delta)/\epsilon^2$; where $R'_n(\mathcal{F}) = (1/n)\mathbb{E}_\sigma[\sup_{\theta\in\Theta} |\sum_{i=1}^n \sigma_i \ell(\theta, Z'_i)|]$, and the loss is bounded by $L = \sup_{(\theta,z)\in\Theta\times\mathcal{Z}} \ell(\theta, z)$.

Proof. Observe first that $||\Phi||_\infty \leqslant \max_k p_k/\varepsilon$ and

$$\mathcal{R}_P(\widetilde{\theta}^*_n) - \inf_{\theta\in\Theta} \mathcal{R}_P(\theta) \leqslant 2\sup_{\theta\in\Theta}\left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(\theta) - \widetilde{\mathcal{R}}_{w*,n}(\theta)\right| + 2\sup_{\theta\in\Theta}\left|\widetilde{\mathcal{R}}_{w*,n}(\theta) - \mathcal{R}_P(\theta)\right|.$$

The result then directly follows from the application of Lemmas 9.1 and 9.5 combined with the union bound. □

Just like in Corollary 9.3, the bound in Theorem 9.6 explodes when ε vanishes, which corresponds to the situation where a stratum $k \in \{1, \dots, K\}$ is very poorly represented in the training data, *i.e.* when $p'_k << p_k$. Again, as highlighted by the experiments carried out, reweighting the losses in a frequentist (ERM) approach guarantees good generalization properties in a specific setup only, *i.e.* when the training information, though biased, is sufficiently informative.

9.3.2 Positive-Unlabeled Learning

Relaxing the *stratum-shift* assumption made in the previous subsection, the importance function becomes more complex and writes:

$$\Phi(x, s) = \frac{dP}{dP'}(x, s) = \sum_{k=1}^{K} \mathbb{I}\{s = k\}\frac{p_k}{p'_k}\frac{dF_k}{dF'_k}(x),$$

where F_k and F'_k are respectively the conditional distributions of X given $S = k$ and of X' given $S' = k$. The Positive-Unlabeled (PU) learning problem, which has recently been the subject of much attention (see *e.g.* du Plessis et al. (2014), du Plessis et al. (2015), Kiryo et al. (2017)), provides a typical example of this situation. Re-using the notations introduced in section 9.2, in the PU problem, the testing and training distributions P and P' are respectively described by the triplets (p, F_+, F_-) and (q, F_+, F), where $F = pF_+ + (1-p)F_-$ is the marginal distribution of X. Hence, the objective pursued is to solve a binary classification task, based on the sole observation of a training sample pooling data with positive labels and unlabeled data, q denoting the theoretical fraction of positive data among the dataset. As noticed in du Plessis et al. (2014) (see also du Plessis et al. (2015), Kiryo et al. (2017)), the likelihood/importance function can be expressed in a simple manner. Precisely, $\forall (x, y) \in \mathcal{X} \times \{-1, +1\}$,

$$\Phi(x, y) = \frac{p}{q}\mathbb{I}\{y = +1\} + \frac{1}{1-q}\mathbb{I}\{y = -1\} - \frac{p}{1-q}\frac{dF_+}{dF}(x)\mathbb{I}\{y = -1\}. \tag{9.9}$$

Based on an *i.i.d.* sample $(X'_1, Y'_1), \dots, (X'_n, Y'_n)$ drawn from P' combined with the knowledge of p (which can also be estimated from PU data, see *e.g.* du Plessis and Sugiyama (2014)) and using that $F_- = (1/(1-p))(F - pF_+)$, one may obtain estimators of q, F_+ and F by computing $n'_+/n = (1/n)\sum_{i=1}^n \mathbb{I}\{Y'_i = +1\}$, $\widehat{F}_+ = (1/n'_+)\sum_{i=1}^n \mathbb{I}\{Y'_i = +1\}\delta_{X'_i}$ and $\widehat{F} = (1/n'_-)\sum_{i=1}^n \mathbb{I}\{Y'_i = -1\}\delta_{X'_i}$. However, plugging these quantities into (9.9) do not permit to get a statistical version of the importance function, insofar as the probability measures $\widehat{F}_+$ and $\widehat{F}$ are mutually singular with probability one, as soon as F_+ is continuous. Of course, as proposed in du Plessis et al. (2014), one may use statistical methods (*e.g.* kernel smoothing) to build distribution estimators, that ensures absolute continuity but are subject to the curse of dimensionality. However, WERM can still be applied in this case, by observing that: $\forall g \in \mathcal{G}$,

$$\mathcal{R}_P(g) = -p + \mathbb{E}_{P'}\left[\frac{2p}{q}\mathbb{I}\{g(X') = -1,\ Y' = +1\} + \frac{1}{1-q}\mathbb{I}\{g(X') = +1,\ Y' = -1\}\right], \tag{9.10}$$

which leads to the weighted empirical risk:

$$\frac{2p}{n'_+}\sum_{i:Y'_i=+1}\mathbb{I}\{g(X'_i)=-1\}+\frac{1}{n'_-}\sum_{i:Y'_i=-1}\mathbb{I}\{g(X'_i)=+1\}. \tag{9.11}$$

Minimization of (9.11) yields rules $\widetilde{g}_n$ whose generalization ability regarding the binary problem related to (p,F_+,F_-) can be guaranteed, as shown by the following result Theorem 9.8 implied by combining Lemmas 9.1 and 9.7, the form of the weighted empirical risk in this case being quite similar to (9.7).

Lemma 9.7. *Let $\varepsilon\in(0,\ 1/2)$. Suppose that $q\in(\varepsilon,\ 1-\varepsilon)$. For any $\delta\in(0,1)$ and $n\geqslant 2\log(2/\delta)/\varepsilon^2$, we have w.p $\geqslant 1-\delta$:*

$$\sup_{g\in\mathcal{G}}\left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g)-\widetilde{\mathcal{R}}_{w*,n}(g)\right|\leqslant\frac{2(2p+1)}{\varepsilon^2}\sqrt{\frac{\log(2/\delta)}{2n}}.$$

Proof. Apply twice the Taylor expansion

$$\frac{1}{x}=\frac{1}{a}-\frac{x-a}{a^2}+\frac{(x-a)^2}{xa^2},$$

so as to get

$$\begin{aligned}\frac{1}{n'_+/n}&=\frac{1}{q}-\frac{n'_+/n-q}{q^2}+\frac{(n'_+/n-q)^2}{q^2n'_+/n},\\ \frac{1}{n'_-/n}&=\frac{1}{1-q}-\frac{n'_-/n-1+q}{(1-q)^2}+\frac{(n'_-/n-1+q)^2}{(1-q)^2n'_-/n}.\end{aligned}$$

This yields the decomposition

$$\begin{aligned}\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g)-\widetilde{\mathcal{R}}_{w*,n}(g)=&-\frac{2p}{q^2}\left(\frac{n'_+}{n}-q\right)\frac{1}{n}\sum_{i=1}^n\mathbb{I}\{g(X'_i)=-1,\ Y'_i=+1\}\\&-\frac{1}{(1-q)^2}\left(\frac{n'_-}{n}-1+q\right)\frac{1}{n}\sum_{i=1}^n\mathbb{I}\{g(X'_i)=+1,\ Y'_i=-1\}\\&+\frac{2p(n'_+/n-q)^2}{q^2n'_+/n}\frac{1}{n}\sum_{i=1}^n\mathbb{I}\{g(X'_i)=-1,\ Y'_i=+1\}\\&+\frac{(n'_-/n-1+q)^2}{(1-q)^2n'_-/n}\frac{1}{n}\sum_{i=1}^n\mathbb{I}\{g(X'_i)=+1,\ Y'_i=-1\}.\end{aligned}$$

We deduce that

$$\left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g)-\widetilde{\mathcal{R}}_{w*,n}(g)\right|\leqslant\frac{|n'_+/n-q|}{\varepsilon^2}\left(2p+1+|n'_+/n-q|\left(\frac{2p}{n'_+/n}+\frac{1}{1-n'_+/n}\right)\right).$$

By virtue of Hoeffding inequality, we obtain that, for any $\delta\in(0,1)$, we have *w.p.* $\geqslant 1-\delta$:

$$\left|n'_+/n-q\right|\leqslant\sqrt{\frac{\log(2/\delta)}{2n}},$$

so that, in particular, $\min\{n'_+/n,\ 1-n'_+/n\}\geqslant\varepsilon-\sqrt{\log(2/\delta)/(2n)}$. This yields the desired result. □

Theorem 9.8. *Let $\varepsilon\in(0,\ 1/2)$. Suppose that $q\in(\varepsilon,\ 1-\varepsilon)$. Let $\widetilde{g}_n$ be any minimizer of the weighted empirical risk (9.11) over class $\mathcal{G}$. For any $\delta\in(0,1)$ and $n\geqslant 2\log(4/\delta)/\varepsilon^2$, we have w.p $\geqslant 1-\delta$:*

$$\mathcal{R}_P(\widetilde{g}_n)-\inf_{g\in\mathcal{G}}\mathcal{R}_P(g)\leqslant\frac{2\max(2p,1)}{\varepsilon}\left(2\mathbb{E}[\mathfrak{R}'_n(\mathcal{G})]+\sqrt{\frac{2\log(2/\delta)}{n}}\right)+\frac{4(2p+1)}{\varepsilon^2}\sqrt{\frac{\log(4/\delta)}{2n}},$$

where $\mathfrak{R}'_n(\mathcal{G})=(1/n)\mathbb{E}_\sigma[\sup_{g\in\mathcal{G}}|\sum_{i=1}^n\sigma_i\mathbb{I}\{g(X'_i)\neq Y'_i\}|]$.

Proof. Observe first that $||\Phi||_\infty \leqslant \max(2p,\ 1)/\varepsilon$ and

$$\mathcal{R}_P(\widetilde{g}_n) - \inf_{g\in\mathcal{G}} \mathcal{R}_P(g) \leqslant 2\sup_{g\in\mathcal{G}} \left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) - \widetilde{\mathcal{R}}_{w*,n}(g)\right| + 2\sup_{g\in\mathcal{G}} \left|\widetilde{\mathcal{R}}_{w*,n}(g) - \mathcal{R}_P(g)\right|,$$

with weighted empirical risk $\widetilde{\mathcal{R}}_{w*,n}(g)$ defined in (9.11). The result then directly follows from the application of Lemmas 9.1 and 9.7 combined with the union bound. □

In the next section, we show that better generalization bounds can be obtained when one has an estimator $\widehat{\eta}$ of the posterior probability η, which paves the way for iterative procedures for PU learning.

9.3.3 Alternative Approach for Positive-Unlabeled Learning

In the usual PU learning approach, observe that:

$$\Phi(x,y) = \frac{p}{q}\mathbb{I}\{y=+1\} + \frac{1-\eta(x)}{1-q}\mathbb{I}\{y=-1\}, \tag{9.12}$$

in the case when an estimate $\widehat{\eta}(x)$ of $\eta(x)$ is available, one can perform WERM using the empirical weight function:

$$\widehat{\Phi}(x,y) = \frac{np}{n'_+}\mathbb{I}\{y=+1\} + \frac{1-\widehat{\eta}(x)}{1-n'_+/n}\mathbb{I}\{y=-1\}. \tag{9.13}$$

A bound that describes how this approach generalizes, depending on the accuracy of estimate $\widehat{\eta}$, can be easily established, which is summarized in Theorem 9.10, again a direct consequence of Lemmas 9.1 and 9.9.

Lemma 9.9. *Let weights $\widehat{w}*$ be defined as in Theorem 9.10. Let $\varepsilon \in (0,\ 1/2)$. Suppose that $q \in (\varepsilon,\ 1-\varepsilon)$. For any $\delta \in (0,1)$ and $n \geqslant 2\log(2/\delta)/\varepsilon^2$, we have w.p $\geqslant 1-\delta$:*

$$\sup_{g\in\mathcal{G}} \left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) - \widetilde{\mathcal{R}}_{w*,n}(g)\right| \leqslant \frac{2}{\varepsilon^2}\sqrt{\frac{\log(2/\delta)}{2n}} + 2\sup_{x\in\mathcal{X}} |\widehat{\eta}(x) - \eta(x)|.$$

Proof. Apply twice the Taylor expansion:

$$\frac{1}{x} = \frac{1}{a} - \frac{x-a}{a^2} + \frac{(x-a)^2}{xa^2},$$

so as to get:

$$\begin{aligned}
\frac{1}{n'_+/n} &= \frac{1}{q} - \frac{n'_+/n - q}{q^2} + \frac{(n'_+/n-q)^2}{q^2 n'_+/n},\\
\frac{1}{n'_-/n} &= \frac{1}{1-q} - \frac{n'_-/n - 1 + q}{(1-q)^2} + \frac{(n'_-/n-1+q)^2}{(1-q)^2 n'_-/n}.
\end{aligned}$$

This yields the decomposition:

$$\begin{aligned}
\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) - \widetilde{\mathcal{R}}_{w*,n}(g) = &-\frac{p}{q^2}\left(\frac{n'_+}{n} - q\right)\frac{1}{n}\sum_{i=1}^n \mathbb{I}\{g(X'_i) = -1,\ Y'_i = +1\}\\
&- \frac{1}{(1-q)^2}\left(\frac{n'_-}{n} - 1 + q\right)\frac{1}{n}\sum_{i=1}^n (1-\widehat{\eta}(X'_i))\mathbb{I}\{g(X'_i) = +1,\ Y'_i = -1\}\\
&+ \frac{p(n'_+/n-q)^2}{q^2 n'_+/n}\frac{1}{n}\sum_{i=1}^n \mathbb{I}\{g(X'_i) = -1,\ Y'_i = +1\}\\
&+ \frac{(n'_-/n-1+q)^2}{(1-q)^2 n'_-/n}\frac{1}{n}\sum_{i=1}^n (1-\widehat{\eta}(X'_i))\mathbb{I}\{g(X'_i) = +1,\ Y'_i = -1\}\\
&+ \frac{1}{(1-q)n}\sum_{i=1}^n (\eta(X'_i) - \widehat{\eta}(X'_i))\mathbb{I}\{g(X'_i) = +1,\ Y'_i = -1\}.
\end{aligned}$$

We deduce that:

$$\left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(g) - \widetilde{\mathcal{R}}_{w*,n}(g)\right| \leqslant \frac{|n'_+/n - q|}{\varepsilon^2}\left(1 + |n'_+/n - q|\left(\frac{1}{n'_+/n} + \frac{1}{1 - n'_+/n}\right)\right) + \frac{n'_-/n}{1-q}\sup_{x\in\mathcal{X}}|\widehat{\eta}(x) - \eta(x)|.$$

By virtue of Hoeffding inequality, we obtain that, for any $\delta \in (0,1)$, we have *w.p.* $\geqslant 1-\delta$:

$$|n'_+/n - q| \leqslant \sqrt{\frac{\log(2/\delta)}{2n}},$$

so that, in particular, $\min\{n'_+/n,\ 1 - n'_+/n\} \geqslant \varepsilon - \sqrt{\log(2/\delta)/(2n)}$. Moreover, still under this event, $n'_-/n \leqslant 2(1-q)$ if $n \geqslant \log(2/\delta)/(2\varepsilon^2)$. This yields the desired result. □

Theorem 9.10. *Let $\widehat{w}_i^* = \widehat{\Phi}(X'_i, Y'_i)$ for all $i \in \{1,\dots,n\}$ with $\widehat{\Phi}$ defined in (9.13). Suppose that the hypotheses of Lemma 9.9 are fulfilled. Let $\widetilde{g}_n$ be any minimizer of $\widetilde{\mathcal{R}}_{\widehat{w}*,n}$ over class $\mathcal{G}$. For any $\delta \in (0,1)$ and $n \geqslant 2\log(4/\delta)/\varepsilon^2$, we have w.p $\geqslant 1-\delta$:*

$$\mathcal{R}_P(\widetilde{g}_n) - \inf_{g\in\mathcal{G}}\mathcal{R}_P(g) \leqslant \frac{2\max(p,1-p)}{\varepsilon}\left(\mathbb{E}[\mathfrak{R}'_n(\mathcal{G})] + \sqrt{\frac{2\log(2/\delta)}{n}}\right) + \frac{4}{\varepsilon^2}\sqrt{\frac{\log(4/\delta)}{2n}} + 4\sup_{x\in\mathcal{X}}|\widehat{\eta}(x) - \eta(x)|.$$

Proof. Observe first that $||\Phi||_\infty \leqslant \max_k p_k/\varepsilon$ and

$$\mathcal{R}_P(\widetilde{\theta}_n^*) - \inf_{\theta\in\Theta}\mathcal{R}_P(\theta) \leqslant 2\sup_{\theta\in\Theta}\left|\widetilde{\mathcal{R}}_{\widehat{w}*,n}(\theta) - \widetilde{\mathcal{R}}_{w*,n}(\theta)\right| + 2\sup_{\theta\in\Theta}\left|\widetilde{\mathcal{R}}_{w*,n}(\theta) - \mathcal{R}_P(\theta)\right|.$$

The result then directly follows from the application of Lemmas 9.1-9.9 combined with the union bound. □

9.3.4 Learning from Censored Data

Another important example of sample bias is the censorship setting where the learner has only access to (right) censored targets $\min(Y', C')$ instead of Y'. Intuitively, this situation occurs when Y' is a duration/date, e.g. the date of death of a patient modeled by covariates X', and the study happens at a (random) date C'. Hence if $C' \leqslant Y'$, then we know that the patient is still alive at time C' but the target time Y' remains unknown. This problem has been extensively studied (see e.g. Fleming and Harrington (2011), Andersen et al. (2012) and the references therein for the asymptotic theory and Ausset et al. (2019) for finite-sample guarantees): we show here that it is an instance of WERM. Formally, we respectively denote by P and P' the testing and training distributions of the *r.v.*'s $(X, \min(Y,C), \mathbb{I}\{Y \leqslant C\})$ and $(X', \min(Y',C'), \mathbb{I}\{Y' \leqslant C'\})$ both valued in $\mathbb{R}^d \times \mathbb{R}_+ \times \{0,1\}$ (with Y, Y', C, C' all nonnegative *r.v.*'s) and such that the pairs (X,Y) and (X',Y') share the same distribution Q. Moreover, $C > Y$ with probability 1 (*i.e.* the testing data are never censored) and Y' and C' are assumed to be conditionally independent given X'. Hence, for all $(x,y,\delta) \in \mathbb{R}^d \times \mathbb{R}_+ \times \{0,1\}$:

$$dP(x,y,\delta) = \delta dQ(x,y),$$

and

$$\delta dP'(x,y,\delta) = \delta\mathbb{P}(C' \geqslant y)d\mathbb{P}(X' = x, Y' = y | C' \geqslant y) = \delta S_{C'}(y|x)dQ(x,y),$$

where $S_{C'}(y|x) = \mathbb{P}(C' \geqslant y | X' = x)$ denotes the conditional survival function of C' given X'. Then, the importance function is:

$$\forall (x,y,\delta) \in \mathbb{R}^d \times \mathbb{R}_+ \times \{0,1\}, \quad \Phi(x,y,\delta) = \frac{dP}{dP'}(x,y,\delta) = \frac{\delta}{S_{C'}(y|x)}.$$

In survival analysis, the ratio $\delta/S_{C'}(y|x)$ is called IPCW (inverse of the probability of censoring weight) and $S_{C'}(y|x)$ can be estimated by using the Kaplan-Meier approach, see Kaplan and Meier (1958).

9.4 Extension to Iterative WERM

As highlighted in Remark 2, the importance function can be expressed as a function of the ideal decision function in certain situations: Eq. (9.12) involves the regression function $\eta(x)$, that defines the optimal (Bayes) classifier $g^*(x) = 2\mathbb{I}\{\eta(x) \geqslant 1/2\} - 1$. This simple observation paves the way for a possible incremental application of the WERM approach: in the case where the solution of the WERM problem considered outputs an estimate of the optimal decision function, it can be next re-used for defining and solving a novel WERM problem. Whereas binary classification based on PU data only aims at recovering a single level set of the posterior probability $\eta(x)$, it is not the case of a more ambitious statistical learning problem, referred to as *bipartite ranking*, for which such an incremental version of WERM can be described.

Bipartite ranking based on PU data. In bipartite ranking, the statistical challenge consists of ranking all the instances $x \in \mathcal{X}$ through a *scoring function* $s : \mathcal{X} \to \mathbb{R}$ in the same order as the likelihood ratio $\Psi(X) = (dF_+/dF_-)(X)$, or, equivalently, as the regression function $\eta(x) = \mathbb{P}\{Y = +1 \mid X = x\}$, $x \in \mathcal{X}$: the higher the score $s(X)$, the more likely one should observe $Y = +1$. Let $\mathcal{S} = \{s : \mathcal{X} \to \mathbb{R} \text{ measurable}\}$ denotes the set of all scoring functions on the input space $\mathcal{X}$. A classical way of measuring "how much stochastically larger" a *c.d.f.* g on $\mathbb{R}$ is than another one, h say, consists in drawing the "probability-probability plot":

$$t \in \mathbb{R} \mapsto (1 - h(t),\ 1 - g(t)),$$

with the convention that possible jumps are connected by line segments (in order to guarantee the continuity of the curve). Equipped with this convention, this boils down to plot the graph of the mapping

$$\mathrm{ROC}_{h,g} : \alpha \in (0,1) \mapsto \mathrm{ROC}_{h,g} = 1 - g \circ h^{-1}(1 - \alpha),$$

where $\Gamma^{-1}(u) = \inf\{t \in \mathbb{R} :\ \Gamma(t) \geqslant u\}$ denotes the pseudo-inverse of any *c.d.f.* $\Gamma(t)$ on $\mathbb{R}$. The closer to the left upper corner of the unit square $[0,1]^2$, the larger the *c.d.f.* g is compared to h in a stochastic sense. This approach is known as ROC analysis. The gold standard for evaluating the ranking performance of a scoring function s is thus the ROC curve:

$$\mathrm{ROC}_s := \mathrm{ROC}_{H_s,G_s},$$

where G_s and H_s denote the conditional *c.d.f.* of $s(X)$ given $Y = +1$ and given $Y = -1$ respectively, *i.e.* the images of the class probability distributions F_+ and F_- by the mapping $s(x)$. Indeed, it follows from a standard Neyman-Pearson argument that the ROC curve ROC^* of strictly increasing transforms of $\eta(x)$ is optimal with respect to this criterion in the sense that:

$$\forall \alpha \in (0,1),\ \ \mathrm{ROC}_s(\alpha) \leqslant \mathrm{ROC}^*(\alpha),$$

for any scoring function s. We set $\mathcal{S}^* = \{T \circ \eta :\ T : (0,1) :\to \mathbb{R}\}$. A summary quantity of this functional criterion that is widely used in practice is the *Area Under the* ROC *Curve* (AUC in short), given by:

$$\mathrm{AUC}(s) = \int_{\alpha=0}^{1} \mathrm{ROC}_s(\alpha)\ d\alpha,$$

for $s \in \mathcal{S}$. Beyond its scalar nature, an attractive property of this criterion lies in the fact that it can be interpreted in a probabilistic manner, insofar as we have the relation: $\forall s \in \mathcal{S}$,

$$\mathrm{AUC}(s) = \mathbb{P}\left\{s(X) < s(X') \mid (Y,Y') = (-1,+1)\right\} + \frac{1}{2}\mathbb{P}\left\{s(X) = s(X') \mid (Y,Y') = (-1,+1)\right\}.$$

Denoting by (X_i, Y_i), $i \in \{1,\ 2\}$, independent copies of the pair (X, Y) and placing ourselves in the situation where $s(X)$'s probability distribution is continuous, as observed in Clémençon et al. (2008), we have $\mathrm{AUC}(s) = 1 - L_P(s)/(2p(1-p))$, where

$$L_P(s) := \mathbb{P}\left\{(s(X_1) - s(X_2))(Y_1 - Y_2)) < 0\right\},$$

is the *ranking risk*, the theoretical rate of discording pairs namely, that can be viewed as a pairwise classification risk. Hence, bipartite ranking can be formulated as the problem of learning a scoring function s that minimizes the ranking risk

$$L_P(s) = \mathbb{E}_{P' \otimes P'} \left[\frac{dP}{dP'}(X_1', Y_1') \frac{dP}{dP'}(X_2', Y_2') \times \mathbb{I}\left\{ (s(X_1') - s(X_2'))(Y_1' - Y_2') \right) < 0 \right\} \Big].$$

Now, using Eq. (9.12) and the fact that $\eta = p\Psi/(1-p+p\Psi)$, we have:

$$\begin{aligned} \frac{dP}{dP'}(x,y) = \Phi(x,y) &= \frac{p}{q}\mathbb{I}\{y=+1\} + \frac{1-\eta(x)}{1-q}\mathbb{I}\{y=-1\}, \\ &= \frac{p}{q}\mathbb{I}\{y=+1\} + \frac{1-p}{(1-q)(1-p+p\Psi(x))}\mathbb{I}\{y=-1\}. \end{aligned}$$

Therefore, it has been shown in Clémençon and Vayatis (2009) (see Corollary 5 therein) that for any s^* in $\mathcal{S}^*$,

$$\frac{dF_+}{dF_-}(X) = \frac{G_{s^*}}{H_{s^*}}(s^*(X)) \text{ almost-surely.}$$

For any s candidate, setting $\Psi_s(x) = H_s/G_s(s(x))$, one can define:

$$\Phi_s(x,y) = \frac{p}{q}\mathbb{I}\{y=+1\} + \frac{1-p}{(1-q)(1-p+p\Psi_s(s(x)))}\mathbb{I}\{y=-1\}.$$

From this formula, it is the easy to see how an incremental use of the WERM could be implemented.

- Start from an initial guess s for the optimal scoring functions (*e.g.* solve the empirical ranking risk minimization problem ignoring the bias issue)
- Estimate Φ_s from the (X_i', Y_i')'s and the knowledge of p, observing that one is not confronted with the curse of dimensionality in this case
- Solve the Weighted Empirical Ranking Risk Minimization problem using the weight function

  $$\widehat{\Phi}_s(x_1,y_1)\widehat{\Phi}_s(x_2,y_2),$$

  which produces a new scoring function s and iterate.

Investigating the performance of such an incremental procedure will be the subject of future research.

9.5 Numerical Experiments

This section illustrates the impact of reweighting by the likelihood ratio on classification performances, as a special case of the general strategy presented in Section 9.2. A first simple illustration on known probability distributions highlights the impact of the shapes of the distributions on the importance of reweighting. This example illustrates in the infinite-sample case that separable or almost separable data do not require reweighting, in contrast to noisy data. Since the distribution shapes are unknown for real data, we infer that reweighting will have variable effectiveness, depending on the dataset. We detail other experiments that illustrate the effectiveness of the reweighting approach on real data. Precisely, we first use reweighting on a shift in the class probabilities for the well-known dataset MNIST between training and testing. Secondly, uses the structure of ImageNet to illustrate reweighting with a stratified population and strata distribution bias or *strata bias*.

9.5.1 Importance of Reweighting for Simple Distributions

Introduce a random pair (X,Y) in $[0,1] \times \{-1,+1\}$ where $X \mid Y = +1$ has for *p.d.f.* $f_+(x)$ and $X \mid Y = -1$ has for *p.d.f.* $f_-(x)$, with, for some $\alpha > 0$ and $\beta > 0$:

$$f_-(x) := (1+\beta)(1-x)^\beta \quad \text{and} \quad f_+(x) := (1+\alpha)x^\alpha.$$

As in Section 9.2, the train and test datasets have different class probabilities p' and p for $Y = +1$. The loss ℓ is defined as $\ell(\theta, z) = \mathbb{I}\{(x - \theta)y \geqslant 0\}$ where $\theta > 0$ is a learned parameter. The true risk can be explicitly calculated. For $\theta > 0$, we have:

$$R_P(\theta) = p\theta^{1+\alpha} + (1-p)(1-\theta)^{1+\beta},$$

and the optimal threshold θ_p^* can be found by derivating the risk $R_P(\theta)$. The derivative is zero when θ satisfies:

$$p(1+\alpha)\theta^{\alpha} = (1-p)(1+\beta)(1-\theta)^{\beta}. \tag{9.14}$$

Solving Eq. (9.14) is straightforward for well-chosen values of α, β, which are detailed in Table 9.1. The excess error $\mathcal{E}(p', p) = R_P(\theta_{p'}^*) - R_P(\theta_p^*)$ for the diagonal entries of Table 9.1 are plotted in Fig. 9.1, in the infinite sample case.

	(α, β)			
	$(0,0)$	$(1/2, 1/2)$	$(1,1)$	$(2,2)$
θ_p^*	$[0,1]$	$\frac{(1-p)^2}{p^2+(1-p)^2}$	$1-p$	$\frac{\sqrt{1-p}}{\sqrt{p}+\sqrt{1-p}}$

Table 9.1: Optimal parameters θ^* for different values of α, β.

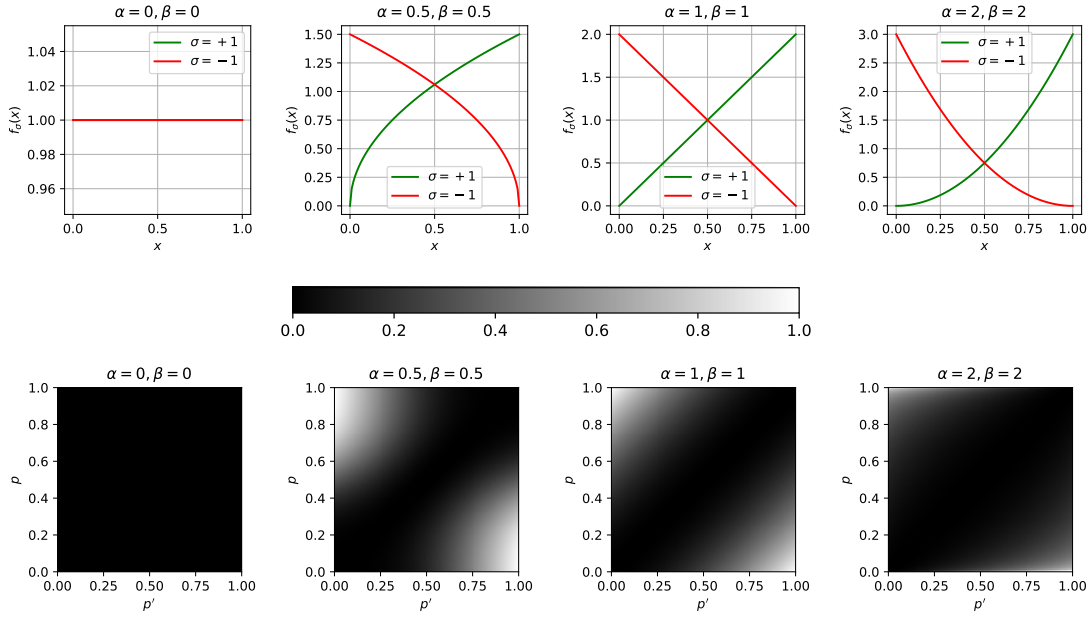


Figure 9.1: Pdf's and values of the excess risk $\mathcal{E}(p', p)$ for different values of α, β.

The results of Fig. 9.1 show that the optimum for the train distribution is significantly different from the optimum for the test distribution when the problem involves Bayes noise.

9.5.2 Generalities on Real Data Experiments

Strategy to induce bias in balanced datasets. In the two real data experiments below, the same strategy is used to induce class distribution bias or strata bias. Since both experiments involve a small test dataset, it is kept intact, while we discard elements of the train dataset to induce bias between the train and test datasets. The bias is parameterized by a single parameter γ, such that when γ is close to one, there is little strata or class bias, while when γ approaches 0, bias is extreme.

The bias we induce is inspired by a power law, which is often used to model unequal distributions. Each value of a modality, *i.e.* a possible value of the stratum or class of an instance, is given by

one of the values of the power law at random. Formally, the target train distribution $\{p'_k\}_{k=1}^K$ over a modality $S \in \{1, \dots, K\}$, is defined for all $1 \leqslant k \leqslant K$ as:

$$p'_k = \frac{\gamma^{-\frac{\lfloor K/2 \rfloor}{\sigma(k)}} p_k}{\sum_{l=1}^K \gamma^{-\frac{\lfloor K/2 \rfloor}{\sigma(k)}} p_k},$$

where σ is a random permutation in $\{1, \dots, K\}$.

To generate a train dataset with modality distribution $\{p'_k\}_{k=1}^K$, we sample instances from the original train data set $\mathcal{D}_n^\circ = \{(X'_i, Y'_i, S'_i)\}_{i=1}^n$, where Y'_i is the class, S'_i is the modality. For the MNIST experiment, $S'_i = Y'_i$, while for Section 8.4, the value S'_i is the stratum of the instance i. The output of the train dataset is noted as $\mathcal{D}_n$, see Algorithm 1 for the detailed algorithm of the train dataset generation.

Algorithm 1 Biased training dataset generation

Input: $\mathcal{D}_n^\circ = \{(X'_i, Y'_i, S'_i)\}_{i=1}^n$, $\{p'_k\}_{k=1}^K$
Output: $\mathcal{D}_n$
$D \leftarrow \varnothing$ *# Initialize the result index set.*
for $k = 1, \dots, K$ **do**
  $\mathcal{I}_k \leftarrow \{i \mid 1 \leqslant i \leqslant n, S'_i = k\}$ *# Count the candidates for each modality.*
end for
$m_{\text{samp}} \leftarrow \min(\#\mathcal{I}_1, \dots, \#\mathcal{I}_K)$
while $m_{\text{samp}} > 0$ **do**
  $m_1, \dots, m_K \leftarrow \mathcal{M}(m_{\text{samp}}, p_1, \dots, p_K)$ *# $\mathcal{M}$ is the multinomial law.*
end while
for $k = 1, \dots, K$ **do**
  $D_k \leftarrow \text{RandSet}(m_k, \mathcal{I}_k)$ *# RandSet(n, X) is a random subset of n elements of X.*
  $\mathcal{I}_k \leftarrow \mathcal{I}_k \backslash D_k$
end for
$m_{\text{samp}} \leftarrow \min(\#\mathcal{I}_1, \dots, \#\mathcal{I}_K)$
$D \leftarrow D \cup \left(\bigcup_{k=1}^K D_k \right)$
$\mathcal{D}_n \leftarrow \{(X'_i, Y'_i, S'_i) \mid i \in D\}$
Return $\mathcal{D}_n$

Models Both MNIST and ImageNet experiments compare two models: a linear model and a multilayer perceptron (MLP) with one hidden layer. Given a classification problem of input x of dimension d with K classes, precisely with $d = 784, K = 10$ for MNIST and $d = 2048, K = 1000$ for ImageNet data, a linear model simply learns the weights matrix $W \in \mathbb{R}^{d \times K}$ and the bias vector $b \in \mathbb{R}^K$ and outputs logits $l = W^\top x + b$. On the other hand, the MLP has a hidden layer of dimension $h = \lfloor (d + K)/2 \rfloor$ and learns the weights matrices $W_1 \in \mathbb{R}^{d,h}, W_2 \in \{h, K\}$ and bias vectors $b_1 \in \mathbb{R}^h, b_2 \in \mathbb{R}^K$ and outputs logits $l = W_2^\top h(W_1^\top x + b_1) + b_2$ where h is the ReLU function, *i.e.* $h : x \mapsto \max(x, 0)$. The number of parameters for each dataset and each model is summarized in Table 9.2.

	Model	
Database	Linear	MLP
MNIST	7,850	315,625
ImageNet	2,049,000	4,647,676

Table 9.2: Number of parameters for each model.

The weight decay or l2 penalization for the linear model and MLP model are written, respectively:

$$\mathcal{P} = \frac{1}{2}\|W\| \quad \text{and} \quad \mathcal{P} = \frac{1}{2}\|W_1\| + \frac{1}{2}\|W_1\|.$$

Experiment	MNIST - Section 9.5.3	ImageNet - Section 8.4
Net weights std init σ_0	0.01	0.01
Weight decay λ Unif	0.01	0.002
Weight decay λ Strata	X	0.003
Weight decay λ Class	0.01	0.003
Weight decay λ Sym data	X	0.001
Learning rate η	0.01	0.001
Momentum γ	0.9	0.9
Batch size B	1,000	1,000
MLP hidden layer size h	397	1,524

Table 9.3: Parameters of the MNIST and ImageNet experiments - Section 9.5.3 and Section 8.4.

Cost function The cost function is the Softmax Cross-Entropy (SCE), which is the most used classification loss in deep learning. Specifically, given logits $l = (l_1, \ldots, l_K) \in \mathbb{R}^K$, the softmax function is $\gamma : \mathbb{R}^k \to [0,1]^K$ with $\gamma = (\gamma_1, \ldots, \gamma_K)$ and for all $k \in \{1, \ldots, K\}$,

$$\gamma_k : l \mapsto \frac{\exp(l_k)}{\sum_{j=0}^{K} \exp(l_j)}.$$

Given an instance with logits l and ground truth class value y, the expression of the softmax cross-entropy $c(l, y)$ is

$$c(l, y) = \sum_{k=1}^{K} \mathbb{I}\{y = k\} \log\left(\gamma_k(l)\right).$$

The loss that is reweighted depending on the cases as described in Section 8.3 is this quantity $c(l, y)$. The loss on the test set is never reweighted, since the test set is the target distribution. The weights and bias of the model that yield the logits are tuned using backpropagation on this loss averaged on random batches of B elements of the training data summed with the regularization term $\lambda \cdot \mathcal{P}$ where λ is a hyperparameter that controls the strength of the regularization.

Preprocessing, optimization, parameters The images of ILSVRC were encoded using the implementation of ResNet50 provided by the library *keras*[1], see Chollet et al. (2015), by taking the flattened output of the last convolutional layer.

Optimization is performed using a momentum batch gradient descent algorithm, which updates the parameters θ_t at timestep t with an update vector v_t by performing the following operations:

$$v_t = \gamma v_{t-1} + \eta \nabla C(\theta_{t-1}),$$
$$\theta_t = \theta_{t-1} - v_t,$$

where η is the step size and γ is the momentum, as explained in Ruder (2016).

The parameters of the learning processes are summarized in Table 9.3. The weight decay parameters λ were cross-validated by trying values on a logarithmic scale, e.g. for ImageNet $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$ and then trying more fine-grained values between the two best results, e.g. for ImageNet 10^{-3} was best and 10^{-2} was second best so we tried $\{0.002, 0.003, 0.004, 0.005\}$. The standard deviation initialization of the weights was chosen by trial-and-error to avoid overflows. The step size was fixed after trying different values to have fast convergence while keeping good convergence properties.

9.5.3 Classes Bias Experiment for MNIST

The impact of the bias correction in the multi-class supervised learning setting described in Section 2 is illustrated on a widely used dataset for studying classification tasks: the MNIST dataset.

[1] https://keras.io/applications/

The MNIST dataset is composed of images $X \in \mathbb{R}^d$ of digits and labels being the value of the digits. In our experiment, we learn to predict the value of the digit so we have $K = 10$ classes corresponding to digits between 0 and 9. The dataset contains $60,000$ images for training and $10,000$ images for testing, all equally distributed within the classes. There is therefore no class bias between train and test samples in the original dataset.

Bias between classes is induced using the power law strategy described above. We deal with the classification task associated to (X, Y) with a linear model or a MLP with one hidden layer that optimizes the softmax cross-entropy (SCE) using momentum gradient descent. We compare the uniform weighting of each instance in the train set (corresponding to the case where there is no reweighting described in Eq. (9.2)) with the reweighting of each instance using the proportion of each label Y for the train and test datasets as described in Eq. (9.7).

The optimization dynamics are summarized in Fig. 9.3. We report the median over 100 runs of these values for the test set and a fixed random sample of the train set. For the test set, we represent 95% confidence-intervals in a lighter tone. The x-axis corresponds to the number of iterations of the learning process.

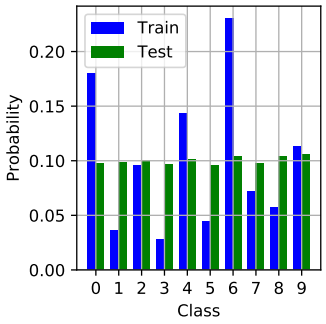
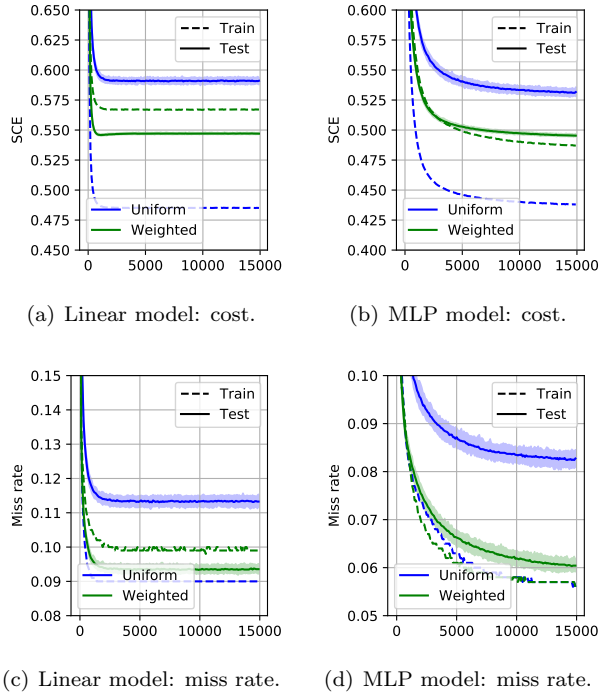


Figure 9.2: Comparison of p_k's and p'_k's.



(a) Linear model: cost.

(b) MLP model: cost.

(c) Linear model: miss rate.

(d) MLP model: miss rate.

Figure 9.3: Dynamics for the class reweighting experiment with MNIST.

For the uniform weights, we see that the misclassification rate is pretty low for the train set, but poor for the test set. By reweighting the instances, we see that we favor low error over the test set, which gives a miss probability reduced by half.

9.5.4 Strata Reweighting Experiments for ImageNet

We focus here on the *learning from biased stratified data* setting introduced in Section 9.3.1 by leveraging the ImageNet Large Scale Visual Recognition Challenge (ILSVRC); a well-known benchmark for the image classification task, see Russakovsky et al. (2014) for more details.

The challenge consists in learning a classifier from 1.3 million training images spread out over 1,000 classes. Performance is evaluated using the validation dataset of 50,000 images of ILSVRC as our test dataset. ImageNet is an image database organized according to the WordNet hierarchy, which groups nouns in sets of related words called synsets. In that context, images are examples of very precise nouns, e.g. *flamingo*, which are contained in a larger synset, e.g. *bird*.

categories	# synset	# images per synset	Total # images
amphibian	94	591	56K
animal	3822	732	2799K
appliance	51	1164	59K
bird	856	949	812K
covering	946	819	774K
device	2385	675	1610K
fabric	262	690	181K
fish	566	494	280K
flower	462	735	339K
food	1495	670	1001K
fruit	309	607	188K
fungus	303	453	137K
furniture	187	1043	195K
geological formation	151	838	127K
invertebrate	728	573	417K
mammal	1138	821	934K
musical instrument	157	891	140K
plant	1666	600	999K
reptile	268	707	190K
sport	166	1207	200K
structure	1239	763	946K
tool	316	551	174K
tree	993	568	564K
utensil	86	912	78K
vegetable	176	764	135K
vehicle	481	778	374K
person	2035	468	952K

Table 9.4: Original categories used to construct the strata for the experiment of Section 8.4.

The impact of reweighting in presence of strata bias is illustrated on the ILSVRC classification problem with broad significance synsets for strata. We detail the data preprocessing necessary to assign strata to the ILSVRC data. These were constructed using a list of 27 high-level categories found on the ImageNet website[2] and copied in Table 9.4. Each ILSVRC image has a ground truth low level synset, either from the name of the training instance, or in the validation textfile for the validation dataset, that is provided by the ImageNet website. The ImageNet API [3] provides the hierarchy of synsets in the form of *is-a* relationships, e.g. *a flamingo is a bird*. Using this information, for each synset in the validation and training database, we gathered all of its ancestors in the hierarchy that were in the table Table 9.4. Most of the synsets had only one ancestor, which then accounts for one stratum. Some of the synsets had no ancestors, or even several ancestors in the table, which accounts in for extra strata, either a *no-category* stratum or a strata composed of the union of several ancestors. The final distribution of the dataset over the created strata is summarized by Figure 9.4. Observe the presence of a *no_strata* stratum and of unions of two high-level synsets strata, e.g. *n00015388_n01905661*. A definition provided by the API of each of the synsets involved in the strata is given in Table 9.5.

To do this, we encode the data using deep neural networks. Specifically our encoding is the flattened output of the last convolutional layer of the network ResNet50 introduced in He et al. (2016). It was trained for classification on the training dataset of ILSVRC. The encodings $X_1, \ldots, X_n$ belong to a 2,048-dimensional space.

A total of 33 strata are derived from a list of high-level categories provided by ImageNet[4]. By default, strata probabilities p_k and p'_k for $1 \leqslant k \leqslant K$ are equivalent between training and testing datasets, meaning that reweighting by Φ would have little to no effect. Since our testing data

[2] `http://www.image-net.org/about-stats`
[3] `http://image-net.org/download-API`
[4] `http://www.image-net.org/about-stats`

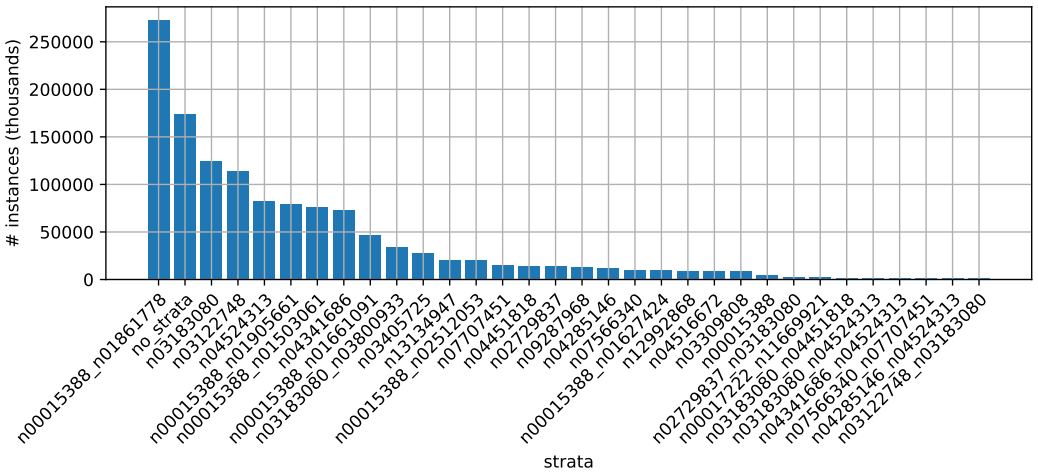


Figure 9.4: Distribution of the ImageNet train dataset over the created strata which definitions are given in Table 9.5.

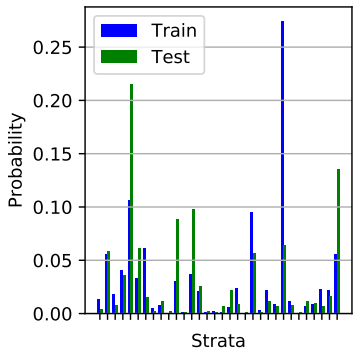


Figure 9.5: Comparison of p_k's and p'_k' for the strata of the ImageNet experiment.

Model	Reweighting	miss rate	top-5 error
Linear	Unif. $\widehat{\Phi} = 1$	0.344	0.130
	Strata $\widehat{\Phi}$	**0.329**	**0.120**
	Class $\widehat{\Phi}$	0.328	0.119
	No bias	0.297	0.102
MLP	Unif. $\widehat{\Phi} = 1$	0.371	0.143
	Strata $\widehat{\Phi}$	**0.364**	**0.138**
	Class $\widehat{\Phi}$	0.363	0.138
	No bias	0.316	0.111

Table 9.6: Table of results for the strata reweighting experiment with ImageNet.

is the validation data of ILSVRC, we have around 25 times more training than testing data. Introducing a strata bias parameter $0 \leqslant \gamma \leqslant 1$, we set the strata train probabilities such that $p'_k = \gamma^{1-\lfloor K/2 \rfloor/k} p_k$ before renormalization and remove train instances so that the train set has the right distribution over strata. When γ is close to one, there is little to no strata bias. In contrast, when γ approaches 0, strata bias is extreme.

The models used are a linear model and a multilayer perceptron (MLP) with one hidden layer. We report significantly better performance when reweighting using the strata information, compared to the case where the strata information is ignored, see Table 9.6. For comparison, we added two reference experiments: one which reweights the train instances by the class probabilities, which we do not know in a stratified population experiment, and one with more data and no strata bias because it uses all of the ILSVRC train data. The dynamics of the learning process can be found in Fig. 9.6 for the linear model and in Fig. 9.7 for the MLP model. The dominance of the linear model over the MLP can be justified by the much higher number of parameters to estimate for the MLP.

9.6 Conclusion

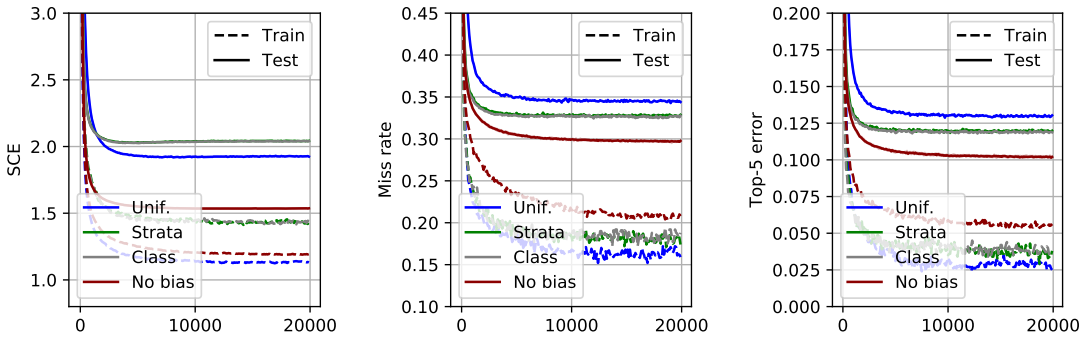
In this chapter, we have considered specific transfer learning problems, where the distribution of the test data P differs from that of the training data, P', and is absolutely continuous with respect

Strata name		Definition	
n00015388	n01861778	animal, animate being, beast (. . .)	mammal, mammalian
no strata			
n03183080		device	
n03122748		covering	
n04524313		vehicle	
n00015388	n01905661	animal, animate being, beast (. . .)	invertebrate
n00015388	n01503061	animal, animate being, beast (. . .)	bird
n04341686		structure, construction	
n00015388	n01661091	animal, animate being, beast (. . .)	reptile, reptilian
n03183080	n03800933	device	musical instrument, instrument
n03405725		furniture, piece of furniture, (. . .)	
n13134947		fruit	
n00015388	n02512053	animal, animate being, beast (. . .)	fish
n07707451		vegetable, veggie, veg	
n04451818		tool	
n02729837		appliance	
n09287968		geological formation, formation	
n04285146		sports equipment	
n00015388	n01627424	animal, animate being, beast (. . .)	amphibian
n07566340		foodstuff, food product	
n12992868		fungus	
n04516672		utensil	
n03309808		fabric, cloth, material, textile	
n00015388		animal, animate being, beast (. . .)	
n00017222	n11669921	plant, flora, plant life	flower
n02729837	n03183080	appliance	device
n03183080	n04451818	device	tool
n03122748	n03183080	covering	device
n04285146	n04524313	sports equipment	vehicle
n04341686	n04524313	structure, construction	vehicle
n07566340	n07707451	foodstuff, food product	vegetable, veggie, veg
n03183080	n04524313	device	vehicle

Table 9.5: Definitions of the strata created for the experiments in Section 8.4, which frequencies are given in Fig. 9.4.

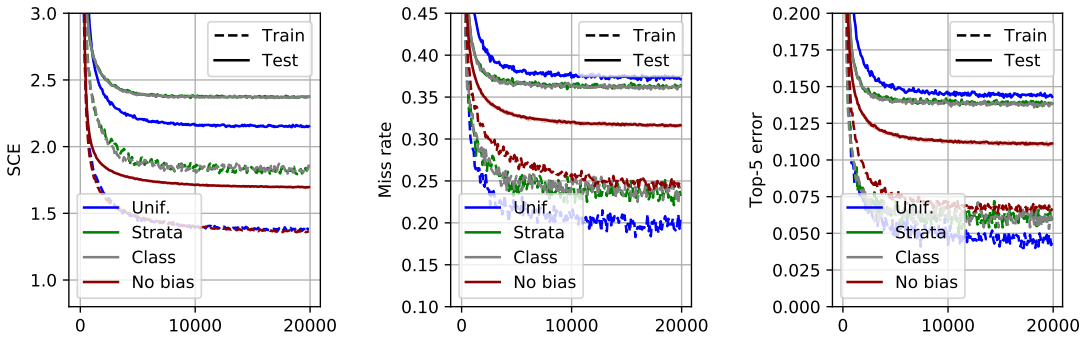
to the latter. This setup encompasses many situations in practice, where the data acquisition process is not perfectly controlled. In this situation, a simple change of measure shows that the target risk may be viewed as the expectation of a weighted version of the basic empirical risk, with ideal weights given by the importance function $\Phi = dP/dP'$, unknown in practice. Throughout this chapter, we have shown that, in statistical learning problems corresponding to a wide variety of practical applications, these ideal weights can be replaced by statistical versions based solely on the training data combined with very simple information about the target distribution. The generalization capacity of rules learned from biased training data by minimization of the weighted empirical risk has been established, with learning bounds. These theoretical results are also illustrated with several numerical experiments.

While some biases can be corrected by addressing the representativeness of data, some have other intrisic reasons. For example, in many settings, differentials in performance are observed even though representativeness is not an issue, such as with respect to age in facial recognition. In those type of situations, one may wish to learn a model that corrects for those differentials with explicit constraints, while simultaneously performing the usual optimization for predictive performance. It is the subject of fairness in machine learning, which we address in the next chapter in the context of fairness in bipartite ranking.



Dynamics for the SCE. Dynamics for the miss rate. Dynamics for the top-5 error.

Figure 9.6: Dynamics for the linear model for the strata reweighting experiment with ImageNet.



Dynamics for the SCE. Dynamics for the miss rate. Dynamics for the top-5 error.

Figure 9.7: Dynamics for the MLP model for the strata reweighting experiment with ImageNet.

Chapter 10

Learning Fair Scoring Functions

Summary: In Part II, we framed similarity learning as a scoring problem on the product space, which fits with the usual evaluation of the 1:1 biometric identification problem. In many instances, the simple reweighting procedure of Chapter 9 does not suffices to correct for discrepancies in error rates between sensitive groups. Facial recognition practitioners have confirmed the importance of those discrepancies (Chapter 1). This concern resonates with the idea of *fairness*, which has received a lot of attention in classification, but not so much in the important problem of bipartite scoring/ranking. However, bipartite ranking/scoring is a preliminary step to our formalization of 1:1 verification (Part II). In this chapter, we propose a flexible approach to group fairness for *bipartite ranking*, the standard learning task of scoring binary labeled data. We argue that the functional nature of the ROC curve, the gold standard measure of ranking performance in this context, leads to several ways of formulating fairness constraints. We introduce general classes of fairness conditions based on AUC and ROC curves, and establish generalization bounds for scoring functions learned under such constraints. Our analysis is in the form of finite-sample bounds (Chapter 2) and is supported by usual results on bipartite ranking (Chapter 3) and U-statistics (Chapter 4). Beyond the theoretical formulation and results, we design practical learning algorithms and illustrate our approach with numerical experiments on real and synthetic data.

10.1 Introduction

With the availability of data at ever finer granularity through the Internet-of-Things and the development of technological bricks to efficiently store and process this data, the infatuation with machine learning and artificial intelligence is spreading to nearly all fields (science, transportation, energy, medicine, security, banking, insurance, commerce, etc.). Expectations are high. AI is supposed to allow for the development of personalized medicine that will adapt a treatment to the patient's genetic traits. Autonomous vehicles will be safer and be in service for longer. There is no denying the opportunities, and we can rightfully hope for an increasing number of successful deployments in the near future. However, AI will keep its promises only if certain issues are addressed. In particular, machine learning systems that make significant decisions for humans, regarding for instance credit lending in the banking sector (Chen, 2018), diagnosis in medicine (Deo, 2015) or recidivism prediction in criminal justice (Rudin et al., 2018), should guarantee that they do not penalize certain groups of individuals. In the specific case of facial recognition, practitioners and governmental agencies have recorded differences in accuracy between ethnicities (Grother and Ngan, 2019), which is expected to penalize people of color.

Hence, stimulated by the societal demand, notions of *fairness* in machine learning and guarantees that they can be fulfilled by decision-making models trained under appropriate constraints have recently been the subject of a good deal of attention in the literature, see *e.g.* Dwork et al. (2012) or Kleinberg et al. (2017) among others. Fairness constraints are generally modeled by means of a

(qualitative) *sensitive variable*, indicating membership to a certain group (*e.g.*, ethnicity, gender). The vast majority of the work dedicated to algorithmic fairness in machine learning focuses on binary classification. In this context, fairness constraints force the classifiers to have the same true positive rate (or false positive rate) across the sensitive groups. For instance, Hardt et al. (2016) and Pleiss et al. (2017) propose to modify a pre-trained classifier in order to fulfill such constraints without deteriorating classification performance. Other work incorporates fairness constraints in the learning stage (see e.g., Agarwal et al. (2018); Woodworth et al. (2017); Zafar et al. (2017a,b, 2019); Menon and Williamson (2018); Bechavod and Ligett (2017); Williamson and Menon (2019)). In addition to algorithms, statistical guarantees (in the form of generalization bounds) are crucial for fair machine learning, as they ensure that the desired fairness will be met at deployment. Such learning guarantees have been established in Donini et al. (2018) for the case of fair classification.

The present chapter is also devoted to algorithmic fairness, but for a different problem: namely, learning scoring functions from binary labeled data. This statistical learning problem, known as *bipartite ranking*, is of considerable importance in applications. It covers in particular tasks such as credit scoring in banking, pathology scoring in medicine or recidivism scoring in criminal justice, for which fairness requirements are a major concern (Kallus and Zhou, 2019). While it can be formulated in the same probabilistic framework as binary classification, bipartite ranking is not a local learning problem: the goal is not to guess whether a binary label Y is positive or negative from an input observation X but to rank any collection of observations $X_1, \ldots, X_n$ by means of a scoring function $s : \mathcal{X} \to \mathbb{R}$ so that observations with positive labels are ranked higher with large probability. Due to the global nature of the task, evaluating the performance is itself a challenge. The gold standard measure, the ROC curve, is functional: it is the PP-plot of the false positive rate *vs* the true positive rate (the higher the curve, the more accurate the ranking induced by s). Sup-norm optimization of the ROC curve has been investigated in Clémençon and Vayatis (2009, 2010), while most of the literature focuses on the maximization of scalar summaries of the ROC curve such as the AUC criterion (Agarwal et al., 2005; Clémençon et al., 2008; Zhao et al., 2011) or alternative measures (Rudin, 2006; Clémençon and Vayatis, 2007; Menon and Williamson, 2016).

We propose a thorough study of fairness in bipartite ranking, where the goal is to guarantee that sensitive variables have little impact on the rankings induced by a scoring function. Similar to ranking performance, there are various possible options to measure the fairness of a scoring function. We start by introducing a general family of AUC-based fairness constraints, which encompasses recently proposed notions (Borkan et al., 2019; Beutel et al., 2019; Kallus and Zhou, 2019) in a unified framework and enables the design of generic methods and generalization bounds. Then, we argue that the AUC is not always appropriate to characterize fairness as two ROC curves with very different shapes may have the same AUC. This motivates our design of richer definitions of fairness for scoring functions related to the ROC curves themselves. Crucially, this novel functional view of fairness has strong implications for fair classification: classifiers obtained by thresholding such fair scoring functions approximately satisfy definitions of classification fairness for a wide range of thresholds. We establish the first generalization bounds for learning fair scoring functions under both AUC and ROC-based fairness constraints, following in the footsteps of Donini et al. (2018) for fair classification. Due to the complex nature of the ranking measures, our proof techniques largely differ from Donini et al. (2018) and require non standard technical tools (e.g. to control deviations of ratios of U-statistics). Beyond our theoretical analysis, we propose efficient training algorithms based on gradient descent and illustrate the practical relevance of our approach on synthetic and real datasets.

The chapter is organized as follows. Section 10.2 reviews bipartite ranking and fairness for ranking. In Section 10.3, we study AUC-based fairness constraints. We then analyze richer ROC-based constraints in Section 10.4. Section 10.5 presents numerical experiments and we conclude in Section 10.6.

10.2 Background and Related Work

In this section, we introduce the main concepts involved in the subsequent analysis.

Probabilistic framework. We place ourselves in the same binary classification setting as Chapter 2 and Chapter 2, thus introduce a random pair (X, Y) characterized by the triplet (p, H, G) or the pair (F, η) (see Section 2.2 in Chapter 2). In the context of fairness, we consider a third random variable Z which denotes the sensitive attribute taking values in $\{0, 1\}$. The pair (X, Y) is said to belong to salient group 0 (resp. 1) when $Z = 0$ (resp. $Z = 1$). The distribution of the triplet (X, Y, Z) can be expressed as a mixture of the distributions of $X, Y|Z = z$. Following the conventions described in Chapter 2, we introduce the quantities $p_z, G^{(z)}, H^{(z)}$ as well as $F^{(z)}, \eta^{(z)}$. For instance, $p_0 = \mathbb{P}\{Y = +1|Z = 0\}$ and the distribution of $X|Y = +1, Z = 0$ is written $G^{(0)}$, *i.e.* for $A \subset \mathcal{X}$, $G^{(0)}(A) = \mathbb{P}\{X \in A|Y = +1, Z = 0\}$. We denote the probability of belonging to group z by $q_z := \mathbb{P}\{Z = z\}$, with $q_0 = 1 - q_1$.

10.2.1 Bipartite Ranking

The goal of bipartite ranking is to learn an order relationship on $\mathcal{X}$ for which positive instances are ranked higher than negative ones with high probability. This order is defined by transporting the natural order on the real line to the feature space through a scoring function $s : \mathcal{X} \to \mathbb{R}$. Given any distribution F over $\mathcal{X}$ and a scoring function s, we write as F_s the cumulative distribution function of $s(X)$ when X follows F. Additionally, we introduce H_s and G_s, defined in Section 3.2 of Chapter 3.

ROC **analysis.** In bipartite ranking, one focuses on the ability of the scoring function s to separate positive from negative data. This is reflected by ROC_{H_s,G_s} (see Eq. (3.1) of Chapter 3), which gives the false negative rate *vs.* false positive rate of binary classifiers $g_{s,t} : x \mapsto 2 \cdot \mathbb{I}\{s(x) > t\} - 1$ obtained by thresholding s at all possible thresholds $t \in \mathbb{R}$. The global summary AUC_{H_s,G_s} (see Definition 3.2 of Chapter 3) serves as a standard performance measure (Clémençon et al., 2008).

ROC curves can more generally be used to visualize the dissimilarity between two real-valued distributions in many applications, *e.g.* anomaly detection, medical diagnosis, information retrieval. We present a more general definition in Definition 10.1 below.

Definition 10.1. *(*ROC ***curve)*** *Let g and h be two cumulative distribution functions on $\mathbb{R}$. The* ROC *curve related to the distributions $g(dt)$ and $h(dt)$ is the graph of the mapping $\mathrm{ROC}_{h,g} : \alpha \in [0, 1] \mapsto 1 - g \circ h^{-1}(1 - \alpha)$. When $g(dt)$ and $h(dt)$ are continuous, it can alternatively be defined as the parametric curve $t \in \mathbb{R} \mapsto (1 - h(t), 1 - g(t))$.*

The L_1 distance of $\mathrm{ROC}_{h,g}$ to the diagonal conveniently quantifies the deviation from the homogeneous case, leading to a generalization of the classic area under the ROC curve (AUC) criterion,

$$\mathrm{AUC}_{h,g} := \int \mathrm{ROC}_{h,g}(\alpha) d\alpha = \mathbb{P}\{S > S'\} + \tfrac{1}{2}\mathbb{P}\{S = S'\},$$

where S and S' denote independent random variables, drawn from $h(dt)$ and $g(dt)$ respectively.

Empirical estimates. In practice, the scoring function s is learned based on a training set $\{(X_i, Y_i)\}_{i=1}^n$ of n *i.i.d.* copies of the random pair (X, Y). Let n_+ and n_- be the number of positive and negative data points respectively, which are sums of *i.i.d.* random (indicator) variables. We introduce by $\widehat{G}_s$ and $\widehat{H}_s$ the empirical counterparts of G_s and H_s, defined formally in Section 3.2.3 of Chapter 3. For any two distributions F, F' over $\mathbb{R}$, we denote the empirical counterparts of $\mathrm{AUC}_{F,F'}$ and $\mathrm{ROC}_{F,F'}$ by:

$$\widehat{\mathrm{AUC}}_{F,F'} := \mathrm{AUC}_{\widehat{F},\widehat{F'}} \quad \text{and} \quad \widehat{\mathrm{ROC}}_{F,F'}(\cdot) := \mathrm{ROC}_{\widehat{F},\widehat{F'}}(\cdot),$$

respectively. In particular, considering $\widehat{\mathrm{AUC}}_{H_s,G_s}$ defined in Eq. (3.2) of Chapter 3, we have $\widehat{\mathrm{AUC}}_{H_s,G_s} = \mathrm{AUC}_{\widehat{H_s},\widehat{G_s}}$.

Empirical risk minimization for bipartite ranking typically consists in maximizing $\widehat{\mathrm{AUC}}_{H_s,G_s}$ over a class of scoring functions (see *e.g.* Clémençon et al. (2008); Zhao et al. (2011)).

10.2.2 Fairness in Binary Classification

In binary classification, the goal is to learn a mapping function $g : \mathcal{X} \mapsto \{-1, +1\}$ that predicts the output label Y from the input random variable X as accurately as possible as measured by a

loss function $L(g)$. One common example is the probability of error $L(g) = \mathbb{P}\{g(X) \neq Y\}$, which is minimized by the Bayes classifier g^*, where $g^*(x) = 2\mathbb{I}\{\eta(x) > 1/2\} - 1$. Any classifier g can be defined by its unique acceptance set $A_g := \{x \in \mathcal{X} \mid g(x) = +1\} \subset \mathcal{X}$.

Existing notions of fairness for binary classification (see Zafar et al., 2019, for a detailed treatment) aim to ensure that g makes similar predictions (or errors) for the two groups. We mention here the common fairness definitions that depend on the ground truth label Y. *Parity in mistreatment* requires that the proportion of errors is the same for the two groups:

$$M^{(0)}(g) = M^{(1)}(g), \tag{10.1}$$

where $M^{(z)}(g) := \mathbb{P}\{g(X) \neq Y \mid Z = z\}$. While this requirement is natural, it considers that all errors are equal: in particular, one can have a high false positive rate (FPR) $H^{(1)}(A_g)$ for one group and a high false negative rate (FNR) $G^{(0)}(A_g)$ for the other. This can be considered unfair when acceptance is an advantage, *e.g.* for job applications. A solution to this issue is to consider *parity in false positive rates*, which writes:

$$H^{(0)}(A_g) = H^{(1)}(A_g), \tag{10.2}$$

as well as *parity in false negative rates*, which writes:

$$G^{(0)}(A_g) = G^{(1)}(A_g). \tag{10.3}$$

We refer to Zafar et al. (2019) for a detailed treatment of fairness definitions for classification.

Remark 10.2 (Connection to bipartite ranking)**.** *A score function $s : \mathcal{X} \to \mathbb{R}$ induces an infinite collection of binary classifiers $\{g_{s,t}(x) := \operatorname{sign}(s(x) - t)\}_{t \in \mathbb{R}}$. While one could fix a threshold $t \in \mathbb{R}$ and try to enforce fairness on $g_{s,t}$, we are interested in notions of fairness for the score function itself, independently of a particular choice of threshold.*

10.2.3 Fairness in Ranking

Fairness for rankings has only recently become a research topic of interest, and most of the work originates from the informational retrieval and recommender systems communities. Given a set of items with *known relevance scores*, they aim to extract a (partial) ranking that balances utility and notions of fairness at the group or individual level, or through a notion of exposure over several queries (Zehlike et al., 2017; Celis et al., 2018; Biega et al., 2018; Singh and Joachims, 2018). Singh and Joachims (2019) and Beutel et al. (2019) extend the above work to the *learning to rank* framework, where the task is to learn relevance scores and ranking policies from a certain number of observed *queries* that consist of query-item features and item relevance scores (which are typically not binary). This framework is fundamentally different from the bipartite ranking setting considered here.

AUC **constraints.** In a setting closer to ours, Kallus and Zhou (2019) introduces measures to quantify the fairness of a known scoring function on binary labeled data (they do not address learning). Similar definitions of fairness are also considered by Beutel et al. (2019), and by Borkan et al. (2019) in a classification context. Below, we present these fairness measures in the unified form of *equalities between two* AUC*s*. In general, the AUC can be seen as a measure of homogeneity between distributions (Clémençon et al., 2009).

Introduce $G_s^{(z)}$ (resp. $H_s^{(z)}$) as the *c.d.f.* of the score on the positives (resp. negatives) of group $z \in \{0, 1\}$, *i.e.* $G_s^{(z)}(t) = G^{(z)}(s(X) \leqslant t)$ and $H_s^{(z)}(t) = H^{(z)}(s(X) \leqslant t)$, for any $t \in \mathbb{R}$. Both Beutel et al. (2019) and Borkan et al. (2019) proposed the following fairness constraints:

$$\mathrm{AUC}_{H_s^{(0)}, G_s^{(0)}} = \mathrm{AUC}_{H_s^{(1)}, G_s^{(1)}}, \tag{10.4}$$

$$\mathrm{AUC}_{H_s, G_s^{(0)}} = \mathrm{AUC}_{H_s, G_s^{(1)}}. \tag{10.5}$$

Eq. (10.4) is referred to as *intra-group pairwise* or *subgroup* AUC fairness and Eq. (10.5) as *pairwise accuracy* Beutel et al. (2019) or *Background Positive Subgroup Negative (BNSP)* AUC fairness Borkan et al. (2019). Eq. (10.4) requires the ranking performance to be equal *within* groups, which is relevant for instance in applications where groups are ranked separately (*e.g.*, candidates for two types of jobs). Eq. (10.5) enforces that positive instances from either group

have the same probability of being ranked higher than a negative example, and can be seen as the ranking counterpart of *parity in false negative rates* in binary classification Hardt et al. (2016), see Eq. (10.3). Borkan et al. (2019) and Kallus and Zhou (2019) also consider:

$$\mathrm{AUC}_{H_s^{(0)},G_s} = \mathrm{AUC}_{H_s^{(1)},G_s}, \quad (10.6) \qquad \mathrm{AUC}_{G_s,G_s^{(0)}} = \mathrm{AUC}_{G_s,G_s^{(1)}}. \quad (10.7)$$

The work of Borkan et al. (2019) refers to Eq. (10.6) as *Backgroup Positive Subgroup Negative (BPSN)* AUC, and can be seen as the ranking counterpart of *parity in false positive rates* in classification (Hardt et al., 2016), see Eq. (10.2). The *Average Equality Gap (AEG)* (Borkan et al., 2019) can be written as $\mathrm{AUC}(G_s, G_s^{(z)}) - 1/2$ for $z \in \{0,1\}$. Eq. (10.7) thus corresponds to an AEG of zero, *i.e.* the scores of the positives of any group are not stochastically larger than those of the other. Beutel et al. (2019) and Kallus and Zhou (2019) also define the *inter-group pairwise fairness* or xAUC *parity*:

$$\mathrm{AUC}_{H_s^{(0)},G_s^{(1)}} = \mathrm{AUC}_{H_s^{(1)},G_s^{(0)}}, \quad (10.8)$$

which imposes that the positives of a group can be distinguished from the negatives of the other group as effectively for both groups. Below, we generalize these AUC-based definitions and derive generalization bounds and algorithms for learning scoring functions under such fairness constraints.

10.3 Fair Scoring via AUC Constraints

In this section, we give a thorough treatment of the problem of statistical learning of scoring functions under AUC-based fairness constraints. First, we introduce a general family of AUC-based fairness definitions which encompasses those presented in Section 10.2.3. We then derive the first generalization bounds for the bipartite ranking problem under such AUC-based fairness constraints. Finally, we propose a practical algorithm to learn such fair scoring functions.

10.3.1 A Family of AUC-based Fairness Definitions

Many sensible fairness definitions can be expressed in terms of the AUC between two distributions. We now introduce a framework to formulate AUC-based fairness constraints as a linear combination of 5 elementary fairness constraints, and prove its generality. Given a scoring function s, we introduce the vector $C(s) = (C_1(s), \ldots, C_5(s))^\top$, where the $C_l(s)$'s, $l \in \{1, \ldots, 5\}$, are elementary fairness measurements. More precisely, the value of $|C_1(s)|$ (resp. $|C_2(s)|$) quantifies the resemblance of the distribution of the negatives (resp. positives) between the two sensitive attributes:

$$C_1(s) = \mathrm{AUC}_{H_s^{(0)},H_s^{(1)}} - 1/2, \qquad C_2(s) = 1/2 - \mathrm{AUC}_{G_s^{(0)},G_s^{(1)}},$$

while $C_3(s)$, $C_4(s)$ and $C_5(s)$ measure the difference in ability of a score to discriminate between positives and negatives for any two pairs of sensitive attributes:

$$C_3(s) = \mathrm{AUC}_{H_s^{(0)},G_s^{(0)}} - \mathrm{AUC}_{H_s^{(0)},G_s^{(1)}}, \qquad C_4(s) = \mathrm{AUC}_{H_s^{(0)},G_s^{(1)}} - \mathrm{AUC}_{H_s^{(1)},G_s^{(0)}},$$
$$C_5(s) = \mathrm{AUC}_{H_s^{(1)},G_s^{(0)}} - \mathrm{AUC}_{H_s^{(1)},G_s^{(1)}}.$$

The family of fairness constraints we consider is then the set of linear combinations of the $C_l(s) = 0$:

$$\mathcal{C}_\Gamma(s): \quad \Gamma^\top C(s) = \textstyle\sum_{l=1}^5 \Gamma_l C_l(s) = 0, \quad \text{with } \Gamma = (\Gamma_1, \ldots, \Gamma_5)^\top \in \mathbb{R}^5. \quad (10.9)$$

Theorem 10.3 (Informal)**.** *The family* $(\mathcal{C}_\Gamma(s))_{\Gamma\in\mathbb{R}^5}$ *compactly captures all relevant* AUC*-based fairness constraints, including (but not limited to) those proposed in previous work.*

Now, we detail the mathematical derivation of the formal statement of this result and give examples of new fairness constraints that can be expressed with Eq. (10.9). We will show that

the family $(\mathcal{C}_\Gamma(s))_{\Gamma\in\mathbb{R}^5}$ covers a wide array of possible fairness constraints in the form of equalities of the AUC's between mixtures of the distributions $D(s)$, with:

$$D(s) := (H_s^{(0)}, H_s^{(1)}, G_s^{(0)}, G_s^{(1)})^\top.$$

Denote by (e_1, e_2, e_3, e_4) the canonical basis of $\mathbb{R}^4$, as well as the constant vector $\mathbf{1} = \sum_{k=1}^4 e_k$. Introducing the probability vectors $\alpha, \beta, \alpha', \beta' \in \mathcal{P}$ where $\mathcal{P} = \{v \mid v \in \mathbb{R}_+^4, \mathbf{1}^\top v = 1\}$, we define the following constraint:

$$\mathrm{AUC}_{\alpha^\top D(s), \beta^\top D(s)} = \mathrm{AUC}_{\alpha'^\top D(s), \beta'^\top D(s)}. \tag{10.10}$$

Theorem 10.4 below formalizes Theorem 10.3, and rules out AUC constraints that are not satisfied when $H^{(0)} = H^{(1)}$ and $G^{(0)} = G^{(1)}$. Such undesirable fairness constraints are those which actually give an advantage to one of the groups, such as $\mathrm{AUC}_{G_s^{(0)} G_s^{(1)}} = 2\mathrm{AUC}_{H_s, G_s} - 1$ which is a special case of Eq. (10.10) that requires the scores of the positives of group 1 to be higher than those of group 0.

Theorem 10.4. *The following propositions are equivalent:*

1. *Eq.* (10.10) *is satisfied for any measurable scoring function s when $H^{(0)} = H^{(1)}$, $G^{(0)} = G^{(1)}$ and $F(\eta(X) = p) < 1$,*
2. *Eq.* (10.10) *is equivalent to $\mathcal{C}_\Gamma(s)$ for some $\Gamma \in \mathbb{R}^5$,*
3. $(e_1 + e_2)^\top[(\alpha - \alpha') - (\beta - \beta')] = 0$.

Proof. Denote $D(s) = (D_1(s), D_2(s), D_3(s), D_4(s))^\top := (H_s^{(0)}, H_s^{(1)}, G_s^{(0)}, G_s^{(1)})^\top$. For any $(i, j) \in \{1, \ldots, 4\}^2$, we introduce the notation:

$$\mathrm{AUC}_{D_i, D_j} : s \mapsto \mathrm{AUC}_{D_i(s), D_j(s)}.$$

Introduce a function M such that $M(s) \in \mathbb{R}^{4\times 4}$ for any $s : X \to \mathbb{R}$, and for any $(i, j) \in \{1, \ldots, 4\}$, the (i, j) coordinate of M writes:

$$M_{i,j} = \mathrm{AUC}_{D_i, D_j} - \frac{1}{2}.$$

First note that, for any s, $M(s)$ is antisymmetric *i.e.* $M_{j,i}(s) = -M_{i,j}(s)$ for any $(i, j) \in \{1, \ldots, 4\}^2$. Then, with $(\alpha, \beta) \in \mathcal{P}^2$, we have that:

$$\mathrm{AUC}_{\alpha^\top D, \beta^\top D} = \alpha^\top M \beta + \frac{1}{2} = \langle M, \alpha\beta^\top \rangle + \frac{1}{2},$$

where $\langle M, M' \rangle = \mathrm{tr}(M^\top M')$ is the standard dot product between matrices. Eq. (10.10) writes as:

$$\langle M, \alpha\beta^\top - \alpha'\beta'^\top \rangle = 0. \tag{10.11}$$

Case of $\alpha = \alpha'$ and $\beta - \beta' = \delta(e_i - e_j)$.

Consider the specific case where $\alpha = \alpha'$ and $\beta - \beta' = \delta(e_i - e_j)$ with $i \neq j$ and $\delta \neq 0$, then

$$\langle M, \alpha(\beta - \beta')^\top \rangle = \delta K_{i,j}^{(\alpha)},$$

where:

$$\begin{aligned} K_{i,j}^{(\alpha)} &:= \langle M, \alpha(e_i - e_j)^\top \rangle = \sum_{k=1}^4 \alpha_k \left[\mathrm{AUC}_{D_k, D_i} - \mathrm{AUC}_{D_k, D_j}\right], \\ &= (\alpha_i + \alpha_j)\left[\frac{1}{2} - \mathrm{AUC}_{D_i, D_j}\right] + \sum_{k \notin \{i,j\}} \alpha_k \left[\mathrm{AUC}_{D_k, D_i} - \mathrm{AUC}_{D_k, D_j}\right]. \end{aligned}$$

The preceding definition implies that $K^{(\alpha)}_{i,j} = -K^{(\alpha)}_{j,i}$. Using $\sum_{k=1}^K \alpha_k = 0$, we can express every $K^{(\alpha)}_{i,j}$ as a linear combinations of the C_l's plus a remainder, precisely:

$$\begin{aligned}
K^{(\alpha)}_{1,2} &= -(\alpha_1+\alpha_2)\,C_1 - \alpha_3(C_3+C_4) - \alpha_4(C_4+C_5),\\
K^{(\alpha)}_{1,3} &= \left(\frac{1}{2} - \mathrm{AUC}_{D_1,D_3}\right) + \alpha_2(-C_1+C_3+C_4) + \alpha_4(-C_2+C_3),\\
K^{(\alpha)}_{1,4} &= \left(\frac{1}{2} - \mathrm{AUC}_{D_1,D_4}\right) + \alpha_2(-C_1+C_4+C_5) + \alpha_3(C_2-C_3-C_4),\\
K^{(\alpha)}_{2,3} &= \left(\frac{1}{2} - \mathrm{AUC}_{D_2,D_3}\right) + \alpha_1(C_1-C_3-C_4) + \alpha_4(-C_2+C_5),\\
K^{(\alpha)}_{2,4} &= \left(\frac{1}{2} - \mathrm{AUC}_{D_2,D_4}\right) + \alpha_1(C_1-C_4-C_5) + \alpha_3(C_2-C_5),\\
K^{(\alpha)}_{3,4} &= (\alpha_3+\alpha_4)\,C_2 + \alpha_1 C_3 + \alpha_2 C_5.
\end{aligned}$$

Hence, it suffices that $\{i,j\} = \{1,2\}$ or $\{i,j\} = \{3,4\}$ for Eq. (10.11) to be equivalent to $\mathcal{C}_\Gamma$ for some $\Gamma \in \mathbb{R}^5$.

Case of $\alpha = \alpha'$.

Any of the $\beta - \beta'$ writes as a positive linear combination of $e_i - e_j$ with $i \neq j$, since:

$$\beta - \beta' = \frac{1}{4}\sum_{i\neq j} \left(\beta_i + \beta'_j\right)(e_i - e_j)\,,$$

which means that, since $K^{(\alpha)}_{i,j} = -K^{(\alpha)}_{j,i}$:

$$\langle M, \alpha(\beta-\beta')^\top\rangle = \frac{1}{4}\sum_{i\neq j}\left(\beta_i+\beta'_j\right)K^{(\alpha)}_{i,j} = \frac{1}{4}\sum_{i<j}\left([\beta_i-\beta_j]-[\beta'_i-\beta'_j]\right)K^{(\alpha)}_{i,j}. \tag{10.12}$$

Note that any linear combination of the $K^{(\alpha)}_{1,3}$, $K^{(\alpha)}_{1,4}$, $K^{(\alpha)}_{2,3}$ and $K^{(\alpha)}_{2,4}$:

$$\gamma_1 \cdot K^{(\alpha)}_{1,3} + \gamma_2 \cdot K^{(\alpha)}_{1,4} + \gamma_3 \cdot K^{(\alpha)}_{2,3} + \gamma_4 \cdot K^{(\alpha)}_{2,4}\,,$$

where $\gamma \in R^4$ with $\mathbf{1}^\top\gamma = 0$ writes as a weighted sum of the C_l for $l \in \{1,\ldots,5\}$.

Hence, it suffices that $\beta_1 + \beta_2 = \beta'_1 + \beta'_2$ for Eq. (10.12) to be equivalent to some $\mathcal{C}_\Gamma$ for some $\Gamma \in \mathbb{R}^5$.

General case.

Note that, using the antisymmetry of M and Eq. (10.12):

$$\begin{aligned}
\langle M, \alpha\beta^\top - \alpha'\beta'^\top\rangle &= \langle M, \alpha(\beta-\beta')^\top\rangle + \langle M, (\alpha-\alpha')\beta'^\top\rangle,\\
&= \langle M, \alpha(\beta-\beta')^\top\rangle - \langle M, \beta'(\alpha-\alpha')^\top\rangle,\\
&= \frac{1}{4}\sum_{i<j}\Big[\left([\beta_i-\beta_j]-[\beta'_i-\beta'_j]\right)K^{(\alpha)}_{i,j} - \left([\alpha_i-\alpha_j]-[\alpha'_i-\alpha'_j]\right)K^{(\beta')}_{i,j}\Big].
\end{aligned}$$

Hence, it suffices that $(e_1+e_2)^\top[(\alpha-\alpha')-(\beta-\beta')] = 0$ for Eq. (10.11) to be equivalent to some $\mathcal{C}_\Gamma$ for some $\Gamma \in \mathbb{R}^5$.

Conclusion.

We denote the three propositions of Theorem 10.4 as P_1, P_2 and P_3.

Assume that $H^{(0)} = H^{(1)}$, $G^{(0)} = G^{(1)}$ and $F(\eta(X) = 1/2) < 1$, then $C_l = 0$ for any $l \in \{1,\ldots,5\}$, which gives:

$$\begin{aligned}
&\langle M(s), \alpha\beta^\top - \alpha'\beta'^\top\rangle\\
&= \frac{1}{4}\left(\frac{1}{2} - \mathrm{AUC}_{H_s,G_s}\right)\left(\sum_{i\in\{1,2\}}\sum_{j\in\{3,4\}}\left[\left([\beta_i-\beta_j]-[\beta'_i-\beta'_j]\right)-\left([\alpha_i-\alpha_j]-[\alpha'_i-\alpha'_j]\right)\right]\right),\\
&= \left(\frac{1}{2} - \mathrm{AUC}_{H_s,G_s}\right)(e_1+e_2)^\top[(\alpha-\alpha')-(\beta-\beta')],
\end{aligned}$$

Table 10.1: Value of $\Gamma = (\Gamma_l)_{l=1}^5$ for all of the AUC-based fairness constraints in the chapter for the general formulation of Eq. (10.9).

Eq.	Γ_1	Γ_2	Γ_3	Γ_4	Γ_5
(10.4)	0	0	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$
(10.5)	0	0	$\frac{q_0(1-p_0)}{1-p}$	0	$\frac{q_1(1-p_1)}{1-p}$
(10.6)	0	0	$\frac{q_0 p_0}{2p}$	$\frac{1}{2}$	$\frac{q_1 p_1}{2p}$
(10.7)	0	1	0	0	0
(10.8)	0	0	0	1	0
(10.13)	0	p_0	$1-p_0$	0	0

It is known that:

$$\mathrm{AUC}_{H_\eta, G_\eta} = \frac{1}{2} + \frac{1}{4p(1-p)} \iint \big|\eta(x) - \eta(x')\big| dF(x) dF(x'),$$

which means that $\mathrm{AUC}_{H_\eta, G_\eta} = 1/2$ implies that $\eta(X) = p$ almost surely (a.s.), and the converse is true.

Assume P_1 is true, then $\mathrm{AUC}_{H_\eta, G_\eta} > 1/2$, hence $(e_1 + e_2)^\top [(\alpha - \alpha') - (\beta - \beta')] = 0$ because Eq. (10.11) is verified for η, and we have shown $P_1 \implies P_3$.

Assume P_3 is true, then $\langle M, \alpha\beta^\top - \alpha'\beta'^\top \rangle$ writes as a linear combination of the C_l's, $l \in \{1, \ldots, 5\}$, and we have shown that $P_3 \implies P_2$.

Assume P_2 is true, then observe that if $H^{(0)} = H^{(1)}$ and $G^{(0)} = G^{(1)}$, then any $\mathcal{C}_\Gamma$ is satisfied for any $\Gamma \in \mathbb{R}^5$, and we have shown that $P_2 \implies P_1$, which concludes the proof. □

Recovering existing AUC-based fairness constraints. All AUC-based fairness constraints proposed in previous work (see Section 10.2.3) write as instances of our general definition for a specific choice of Γ, see Table 10.1. Note that Γ might depend on the quantities q_0, p_0, q_1, p_1.

Expressing new AUC-based fairness constraints. Relevant fairness constraints that have not been considered in previous work can be expressed using our general formulation. Denoting $F_s^{(0)} = (1 - p_0) H_s^{(0)} + p_0 G_s^{(0)}$, consider for instance the following constraint:

$$\mathrm{AUC}_{F_s^{(0)}, G_s^{(0)}} = \mathrm{AUC}_{F_s^{(0)}, G_s^{(1)}}. \tag{10.13}$$

It equalizes the expected position of the positives of each group with respect to a *reference group* (here group 0). Another fairness constraint of interest is based on the rate of misranked pairs when one element is in a specific group:

$$\begin{aligned} E(s, z) :=& (1/2) \cdot \mathbb{P}\{s(X) = s(X) \mid Y \neq Y', Z = z\} \\ &+ \mathbb{P}\{(s(X) - s(X))(Y - Y') > 0 \mid Y \neq Y', Z = z\}. \end{aligned}$$

The equality $E(s, 0) = E(s, 1)$ can be seen as the analogue of *parity in mistreatment* for the task of ordering pairs, see Eq. (10.1). It is easy to see that this constraint can be written in the form of Eq. (10.10) and that point 1 of Theorem 10.4 holds, hence it is equivalent to $\mathcal{C}_\Gamma(s)$ for some $\Gamma \in \mathbb{R}^5$.

As we show below, our unifying framework enables the design of general problem formulations, statistical guarantees and algorithms which can then be instantiated to the specific notion of AUC-based fairness that the practitioner is interested in.

10.3.2 Learning Problem and Statistical Guarantees

We now formulate the problem of fair ranking based on the fairness definitions introduced above. Introducing fairness as a hard constraint is tempting, but may be costly in terms of ranking performance. In general, there is indeed a trade-off between the ranking performance and the level of fairness.

For a family of scoring functions $\mathcal{S}$ and some instantiation Γ of our general fairness definition in Eq. (10.9), we thus define the learning problem as follows:

$$\max_{s\in\mathcal{S}} \quad \mathrm{AUC}_{H_s,G_s} - \lambda\big|\Gamma^\top C(s)\big|, \tag{10.14}$$

where $\lambda \geqslant 0$ is a hyperparameter balancing ranking performance and fairness.

For the sake of simplicity and concreteness, we focus on the special case of the fairness definition in Eq. (10.4) — one can easily extend our analysis to any other instance of our general definition in Eq. (10.9). Thus, we denote by s^*_λ the scoring function that maximizes the objective $L_\lambda(s)$ of Eq. (10.14), where:

$$L_\lambda(s) := \mathrm{AUC}_{H_s,G_s} - \lambda\big|\mathrm{AUC}_{H_s^{(0)},G_s^{(0)}} - \mathrm{AUC}_{H_s^{(1)},G_s^{(1)}}\big|.$$

Given a training set $\{(X_i,Y_i,Z_i)\}_{i=1}^n$ of n *i.i.d.* copies of the random triplet (X,Y,Z), we denote by $n^{(z)}$ the number of points in group $z\in\{0,1\}$, and by $n_+^{(z)}$ (resp. $n_-^{(z)}$) the number of positive (resp. negative) points in this group. The empirical counterparts of $H_s^{(z)}$ and $G_s^{(z)}$ are then given by, respectively:

$$\widehat{H}_s^{(z)}(t) = (1/n_-^{(z)})\textstyle\sum_{i=1}^n \mathbb{I}\left\{Z_i = z, Y_i = -1, s(X_i)\leqslant t\right\},$$
$$\widehat{G}_s^{(z)}(t) = (1/n_+^{(z)})\textstyle\sum_{i=1}^n \mathbb{I}\left\{Z_i = z, Y_i = +1, s(X_i)\leqslant t\right\}.$$

Recalling the notation $\widehat{\mathrm{AUC}}_{F,F'} := \mathrm{AUC}_{\widehat{F},\widehat{F}'}$ from Section 10.2.1, the empirical problem writes:

$$\widehat{L}_\lambda(s) := \widehat{\mathrm{AUC}}_{H_s,G_s} - \lambda\big|\widehat{\mathrm{AUC}}_{H_s^{(0)},G_s^{(0)}} - \widehat{\mathrm{AUC}}_{H_s^{(1)},G_s^{(1)}}\big|.$$

We denote its maximizer by $\widehat{s}_\lambda$. We can now state our statistical learning guarantees for fair ranking.

Theorem 10.5. *Assume the class of functions $\mathcal{S}$ is VC-major with finite VC-dimension $V<+\infty$ and that there exists $\epsilon>0$ s.t. $\min_{z\in\{0,1\},y\in\{-1,1\}}\mathbb{P}\{Y=y,Z=z\}\geqslant\epsilon$. Then, for any $\delta>0$, for all $n>1$, we have with probability (w.p.) at least $1-\delta$:*

$$\epsilon^2\cdot\left[L_\lambda(s^*_\lambda) - L_\lambda(\widehat{s}_\lambda)\right] \leqslant\ C\sqrt{\frac{V}{n}}\cdot(4\lambda+1/2) + \sqrt{\frac{\log(13/\delta)}{n-1}}\cdot\Big(4\lambda+(4\lambda+2)\epsilon\Big) + O(n^{-1}).$$

Proof. Usual arguments imply that: $L_\lambda(s^*_\lambda) - L_\lambda(\widehat{s}_\lambda) \leqslant 2\cdot\sup_{s\in\mathcal{S}}\Big|\widehat{L}_\lambda(s) - L_\lambda(s)\Big|$. Introduce the quantities:

$$\widehat{\Delta} = \sup_{s\in\mathcal{S}}\Big|\widehat{\mathrm{AUC}}_{H_s,G_s} - \mathrm{AUC}_{H_s,G_s}\Big|, \qquad \widehat{\Delta}_0 = \sup_{s\in\mathcal{S}}\Big|\widehat{\mathrm{AUC}}_{H_s^{(0)},G_s^{(0)}} - \mathrm{AUC}_{H_s^{(0)},G_s^{(0)}}\Big|,$$
$$\text{and} \quad \widehat{\Delta}_1 = \sup_{s\in\mathcal{S}}\Big|\widehat{\mathrm{AUC}}_{H_s^{(1)},G_s^{(1)}} - \mathrm{AUC}_{H_s^{(1)},G_s^{(1)}}\Big|.$$

The triangular inequality implies that: $\sup_{s\in\mathcal{S}}\Big|\widehat{L}_\lambda(s) - L_\lambda(s)\Big| \leqslant \widehat{\Delta} + \lambda\widehat{\Delta}_0 + \lambda\widehat{\Delta}_1$.

Case of $\widehat{\Delta}$: Note that:

$$\widehat{\mathrm{AUC}}_{H_s,G_s} = (n(n-1)/2n_+n_-)\cdot\widehat{U}_K(s),$$
$$\text{where} \quad \widehat{U}_K(s) = \frac{2}{n(n-1)}\sum_{1\leqslant i<j\leqslant n} K((s(X_i),Y_i,Z_i),(s(X_j),Y_j,Z_j)),$$

and $K((t,y,z),(t',y',z')) = \mathbb{I}\{(y-y')(t-t')>0\} + (1/2)\cdot\mathbb{I}\{y\neq y', t=t'\}$. The quantity $\widehat{U}_K(s)$ is a known type of statistic and is called a U-statistic, see *e.g.* Lee (1990) for an overview. We write $U_K(s) := \mathbb{E}[\widehat{U}_K(s)] = 2p(1-p)\mathrm{AUC}_{H_s,G_s}$.

Introducing $\widehat{m} := n_+n_-/n^2 - p(1-p)$, we have that, since $\sup_{s\in\mathcal{S}}|\widehat{U}_K(s)| \leqslant 2n_+n_-/(n(n-1))$:

$$\widehat{\Delta} \leqslant \left|\frac{n(n-1)}{2n_+n_-} - \frac{1}{2p(1-p)}\right|\cdot\sup_{s\in\mathcal{S}}\Big|\widehat{U}_K(s)\Big| + \frac{1}{2p(1-p)}\cdot\sup_{s\in\mathcal{S}}\Big|\widehat{U}_K(s) - U_K(s)\Big|,$$
$$\leqslant \frac{1}{p(1-p)}\left|\widehat{m} + \frac{n_+n_-}{n^2(n-1)}\right| + \frac{1}{2p(1-p)}\cdot\sup_{s\in\mathcal{S}}\Big|\widehat{U}_K(s) - U_K(s)\Big|.$$

The properties of the shatter coefficient described in Györfi (2002) (Theorem 1.12 therein) and the fact that $\mathcal{S}$ is VC-major, imply that the class of sets:

$$\{((x,y),(x',y')) \mid (s(x)-s(x'))(y-y') > 0\}_{s\in\mathcal{S}},$$

is VC with dimension V. The right-hand side term above is covered by Corollary 3 of Clémençon et al. (2008), presented in the preliminaries as Corollary 4.5 in Chapter 4, and we deal now with the left-hand side term.

Hoeffding's inequality implies that, for all $n \geqslant 1$, *w.p.* $\geqslant 1-\delta$,

$$\left|\frac{n_+}{n} - p\right| \leqslant \sqrt{\frac{\log(\frac{2}{\delta})}{2n}}. \tag{10.15}$$

Since $n_- = n - n_+$, we have that:

$$\widehat{m} = (1-2p)\left(\frac{n_+}{n} - p\right) - \left(\frac{n_+}{n} - p\right)^2.$$

It follows that:

$$\left|\widehat{m} + \frac{n_+ n_-}{n^2(n-1)}\right| \leqslant |\widehat{m}| + \frac{1}{4(n-1)} \leqslant (1-2p)\sqrt{\frac{\log(2/\delta)}{2n}} + A_n(\delta),$$

where:

$$A_n(\delta) = \frac{\log(2/\delta)}{2n} + \frac{1}{4(n-1)} = O\left(n^{-1}\right).$$

Finally, a union bound between Corollary 3 of Clémençon et al. (2008) and Eq. (10.15) gives that, using the upper-bound $1/(2n) \leqslant 1/(n-1)$: for any $n > 1$, *w.p.* $\geqslant 1-\delta$,

$$p(1-p)\cdot\widehat{\Delta} \leqslant C\sqrt{\frac{V}{n}} + 2(1-p)\sqrt{\frac{\log(3/\delta)}{n-1}} + A_n(2\delta/3). \tag{10.16}$$

Case of $\widehat{\Delta}_0$*:* Note that:

$$\widehat{\mathrm{AUC}}_{H_s^{(0)},G_s^{(0)}} = \left(n(n-1)/2n_+^{(0)}n_-^{(0)}\right)\cdot\widehat{U}_{K^{(0)}}(s),$$

where $K^{(0)}((t,y,z),(t',y',z')) = \mathbb{I}\{z=0,z'=0\}\cdot K((t,y,z),(t',y',z'))$. We denote:

$$U_{K^{(0)}}(s) := \mathbb{E}[\widehat{U}_{K^{(0)}}(s) = 2q_0^2 p_0(1-p_0)\cdot \mathrm{AUC}_{H_s^{(0)},G_s^{(0)}}.$$

Following the proof of the bound for $\widehat{\Delta}$, introducing $\widehat{m}_0 := n_+^{(0)}n_-^{(0)}/n^2 - q_0^2p_0(1-p_0)$,

$$\widehat{\Delta}_0 \leqslant \frac{1}{q_0^2p_0(1-p_0)}\left|\widehat{m}_0 + \frac{n_+^{(0)}n_-^{(0)}}{n^2(n-1)}\right| + \frac{1}{2q_0^2p_0(1-p_0)}\cdot\sup_{s\in\mathcal{S}}\left|\widehat{U}_{K^{(0)}}(s) - U_{K^{(0)}}(s)\right|.$$

The right-hand side term above is once again covered by Lemma 1. We deal now with the left-hand side term, note that:

$$\begin{aligned}
\widehat{m}_0 &= \frac{n_+^{(0)}n^{(0)}}{n^2} - q_0^2p_0 - \left(\left[\frac{n_+^{(0)}}{n}\right]^2 - q_0^2p_0^2\right), \\
&= q_0p_0\left(\frac{n^{(0)}}{n} - q_0\right) + q_0(1-2p_0)\left(\frac{n_+^{(0)}}{n} - q_0p_0\right) \\
&\quad + \left(\frac{n_+^{(0)}}{n} - q_0p_0\right)\left(\frac{n^{(0)}}{n} - q_0\right) - \left(\frac{n_+^{(0)}}{n} - q_0p_0\right)^2.
\end{aligned}$$

A union bound of two Hoeffding inequalities gives that for any $n > 1$, *w.p.* $\geqslant 1 - \delta$, we have simultaneously:

$$\left| \frac{n^{(0)}}{n} - q_0 \right| \leqslant \sqrt{\frac{\log(4/\delta)}{2n}} \quad \text{and} \quad \left| \frac{n_+^{(0)}}{n} - q_0 p_0 \right| \leqslant \sqrt{\frac{\log(4/\delta)}{2n}}. \tag{10.17}$$

It follows that:

$$\left| \widehat{m}_0 + \frac{n_+^{(0)} n_-^{(0)}}{n^2(n-1)} \right| \leqslant |\widehat{m}_0| + \left| \frac{\left(n^{(0)}\right)^2}{4n^2(n-1)} \right| \leqslant q_0(1-p_0)\sqrt{\frac{\log(4/\delta)}{2n}} + B_n(\delta),$$

where

$$B_n(\delta) = \frac{1}{4(n-1)} + \frac{\log(4/\delta)}{n}.$$

Finally, a union bound between (Clémençon et al., 2008, Corollary 3) and Eq. (10.17) gives, using the upper-bound $1/(2n) \leqslant 1/(n-1)$: for any $n > 1$, *w.p.* $\geqslant 1 - \delta$,

$$q_0^2 p_0(1-p_0) \cdot \widehat{\Delta}_0 \leqslant C\sqrt{\frac{V}{n}} + (1 + q_0(1-p_0))\sqrt{\frac{\log(5/\delta)}{n}} + B_n(4\delta/5). \tag{10.18}$$

Case of $\widehat{\Delta}_1$*:*

One can prove a similar result as Eq. (10.18) for $\widehat{\Delta}_1$: for any $n > 1$, *w.p.* $\geqslant 1 - \delta$,

$$q_1^2 p_1(1-p_1) \cdot \widehat{\Delta}_1 \leqslant C\sqrt{\frac{V}{n}} + (1 + q_1(1-p_1))\sqrt{\frac{\log(5/\delta)}{n}} + B_n(4\delta/5). \tag{10.19}$$

Conclusion:

Under the assumption $\min_{z\in\{0,1\}} \min_{y\in\{-1,1\}} \mathbb{P}\{Y = y, Z = z\} \geqslant \epsilon$, note that $\min(p, 1-p) \geqslant 2\epsilon$. A union bound between Eq. (10.16), Eq. (10.18), and Eq. (10.19). gives that: for any $\delta > 0$ and $n > 1$, *w.p.* $\geqslant 1 - \delta$,

$$\epsilon^2 \cdot (L_\lambda(s_\lambda^*) - L_\lambda(\widehat{s}_\lambda)) \leqslant \; C\sqrt{\frac{V}{n}} \cdot \left(4\lambda + \frac{1}{2}\right) + \sqrt{\frac{\log(13/\delta)}{n-1}} \cdot \Big(4\lambda + (4\lambda+2)\epsilon\Big) + O(n^{-1}),$$

which concludes the proof. □

Theorem 10.5 establishes a learning rate of $O(1/\sqrt{n})$ for our problem of ranking under AUC-based fairness constraints, which holds for any distribution of (X, Y, Z) as long as the probability of observing each combination of label and group is bounded away from zero. As the natural estimate of the AUC involves sums of dependent random variables, the proof of Theorem 10.5 does not follow from usual concentration inequalities on standard averages. Indeed, it requires controlling the uniform deviation of ratios of U-processes indexed by a class of functions of controlled complexity.

10.3.3 Training Algorithm

Maximizing directly $\widehat{L}_\lambda$ by gradient ascent (GA) is not feasible, since the criterion is not continuous, hence not differentiable. Hence, we decided to approximate any indicator function $x \mapsto \mathbb{I}\{x > 0\}$ by a logistic function $\sigma : x \mapsto 1/(1+e^{-x})$.

We learn with stochastic gradient descent using batches $\mathcal{B}_N$ of N elements sampled with replacement in the training set $\mathcal{D}_n = \{(X_i, Y_i, Z_i)\}_{i=1}^n$, with $\mathcal{B}_N = \{(x_i, y_i, z_i)\}_{i=1}^N$. We assume the existence of a small validation dataset $\mathcal{V}_m$, with $\mathcal{V}_m = \{(x_i^{(v)}, y_i^{(v)}, z_i^{(v)})\}_{i=1}^m$. In practice, one splits a total number of instances $n + m$ between the train and validation dataset.

The approximation of $\widehat{\mathrm{AUC}}_{H_s,G_s}$ on the batch writes:

$$\widetilde{\mathrm{AUC}}_{H_s,G_s} = \frac{1}{N_+ N_-} \sum_{i<j} \sigma\left[(s(x_i) - s(x_j))(y_i - y_j)\right],$$

where $N_+ := \sum_{i=1}^N \mathbb{I}\{y_i = +1\}$ and $N_- := N - N_+$ is the number of positive instances in the batch. Similarly, we denote by $N_+^{(z)} := N^{(z)} - N_-^{(z)}$ the number of positive instances of group z in the batch, with:

$$N^{(z)} := \sum_{i=1}^N \mathbb{I}\{z_i = z\} \quad \text{and} \quad N_+^{(z)} := \sum_{i=1}^N \mathbb{I}\{z_i = z, y_i = +1\}.$$

Due to the high number of term involved involved in the summation, the computation of $\widetilde{\mathrm{AUC}}_{H_s,G_s}$ can be very expensive, and we rely on approximations called *incomplete U-statistics*, which simply average a random sample of B nonzero terms of the summation, see Lee (1990). We refer to Clémençon et al. (2016) and Papa et al. (2015) for details on their statistical efficiency and use in the context of SGD algorithms, respectively. Formally, we define the incomplete approximation with $B \in \mathbb{N}$ pairs of $\widetilde{\mathrm{AUC}}_{H_s,G_s}$ as:

$$\widetilde{\mathrm{AUC}}^{(B)}_{H_s,G_s} := \frac{1}{B} \sum_{(i,j)\in\mathcal{D}_B} \sigma\left[(s(x_i) - s(x_j))(y_i - y_j)\right],$$

where $\mathcal{D}_B$ is a random set of B pairs in the set of all possible pairs $\{(i,j) \mid 1 \leqslant i < j \leqslant N\}$.

Here, we give more details on our algorithm for the case of the AUC-based constraint Eq. (10.4). The generalization to other AUC-based fairness constraints is straightforward. For any $z \in \{0,1\}$ the relaxation of $\widehat{\mathrm{AUC}}_{H^{(z)},G^{(z)}}$ on the batch writes:

$$\widetilde{\mathrm{AUC}}_{H_s^{(z)},G_s^{(z)}} = \frac{1}{N_+^{(z)} N_-^{(z)}} \sum_{\substack{i<j \\ z_i=z_j=z}} \sigma\left[(s(x_i) - s(x_j))(y_i - y_j)\right].$$

Similarly as $\widetilde{\mathrm{AUC}}_{H_s,G_s}$, we introduce the sampling-based approximations $\widetilde{\mathrm{AUC}}^{(B)}_{H_s^{(z)},G_s^{(z)}}$ for any $z \in \{0,1\}$.

To minimize the absolute value in Eq. (10.14), we introduce a parameter $c \in [-1,+1]$, which is modified slightly every n_{adapt} iterations so that it has the same sign as the evaluation of $\Gamma^\top C(s)$ on $\mathcal{V}_m$. This allows us to write a cost in the form of a weighted sum of AUC's, with weights that vary during the optimization process. Precisely, it is defined as:

$$\widetilde{L}_{\lambda,c}(s) := \left(1 - \widetilde{\mathrm{AUC}}_{H_s,G_s}\right) + \lambda \cdot c \left(\widetilde{\mathrm{AUC}}_{H_s^{(1)},G_s^{(1)}} - \widetilde{\mathrm{AUC}}_{H_s^{(0)},G_s^{(0)}}\right) + \frac{\lambda_{\text{reg}}}{2} \cdot \|W\|_2^2,$$

where λ_{reg} is a regularization parameter and $\|W\|_2^2$ is the sum of the squared L_2 norms of all of the weights of the model. The sampling-based approximation of $\widetilde{L}_{\lambda,c}$ writes:

$$\widetilde{L}^{(B)}_{\lambda,c}(s) := \left(1 - \widetilde{\mathrm{AUC}}^{(B)}_{H_s,G_s}\right) + \lambda \cdot c \left(\widetilde{\mathrm{AUC}}^{(B)}_{H_s^{(1)},G_s^{(1)}} - \widetilde{\mathrm{AUC}}^{(B)}_{H_s^{(0)},G_s^{(0)}}\right) + \frac{\lambda_{\text{reg}}}{2} \cdot \|W\|_2^2.$$

The algorithm is detailed in Algorithm 2, where sng is the sign function, *i.e.* $\mathrm{sgn}(x) = 2\mathbb{I}\{x > 0\} - 1$ for any $x \in \mathbb{R}$.

10.4 Richer ROC-based Fairness Constraints

The equality between two AUC's considered as fairness constraints in Section 10.3 only quantifies a stochastic order between distributions, not the equality between these distributions. In particular, two very different distributions can be indistinguishable in terms of AUC. As a matter of fact, for continuous ROCs, the equality between their two AUCs only implies that the two ROCs intersect at one unknown point. As a consequence, AUC-based fairness can only guarantee that there exists *some* threshold $t \in \mathbb{R}$ that induces a non-trivial classifier $g_{s,t} := \mathrm{sign}(s(x) - t)$ satisfying a notion of fairness for classification, as detailed below in Section 10.4.1. Unfortunately, the value of t and the corresponding point α of the ROC curve are not known in advance and are difficult to control.

Algorithm 2 Practical algorithm for learning with the AUC-based constraint Eq. (10.4).

Input: training set $\mathcal{D}_n$, validation set $\mathcal{V}_m$
$c \leftarrow 0$
for $i = 1$ **to** n_{iter} **do**
  $\mathcal{B}_N \leftarrow N$ observations sampled with replacement from $\mathcal{D}_n$
  $s \leftarrow$ updated scoring function using a gradient-based algorithm (*e.g.* ADAM), using the derivative of $\widetilde{L}^{(B)}_{\lambda,c}(s)$ on $\mathcal{B}_N$
  if $(n_{\text{iter}} \bmod n_{\text{adapt}}) = 0$ **then**
    $\Delta\text{AUC} \leftarrow \widehat{\text{AUC}}^{(B_\text{v})}_{H_s^{(1)},G_s^{(1)}} - \widehat{\text{AUC}}^{(B_\text{v})}_{H_s^{(0)},G_s^{(0)}}$ computed on $\mathcal{V}_m$
    $c \leftarrow c + \text{sgn}(\Delta\text{AUC}) \cdot \Delta c$
    $c \leftarrow \min(1, \max(-1, c))$
  end if
end for
Output: scoring function s

To see why this can be a problem for fairness, consider for instance pairwise accuracy fairness in Eq. (10.5), which specifies that the probability of scoring a positive instance of a given group higher than a negative example should be the same across groups. Despite this, it is possible for positives from one group to appear less often in the top 1% scores than positives from the other group (via "compensation" in other regions of the score distribution). In a use-case where only top 1% individuals get some advantage (or obtain better advantages), this would be unfair for one group. Learning with AUC-based constraints can thus lead to scoring functions that are inadequate for the use-case of interest. These limitations serve as a motivation to introduce new ROC-based fairness constraints in Section 10.4.2.

10.4.1 Limitations of AUC-based Constraints

In this section, we clarify the relationship between known propositions for fairness in classification on the one hand, and our AUC-based and ROC-fairness for bipartite ranking on the other hand. In a nutshell, we show that: (i) if a scoring function s satisfies an AUC-based fairness constraint, there exists a certain threshold t such that the classifier $g_{s,t}$ obtained by thresholding s at t satisfies fair classification constraints, and (ii) ROC-based fairness constraints allow to directly control the value of t for which $g_{s,t}$ is fair, and more generally to achieve classification fairness for a whole range of thresholds, which is useful to address task-specific operational constraints.

Pointwise ROC equality and fairness in binary classification. As mentioned in the main text, a scoring function $s : \mathcal{X} \to \mathbb{R}$ induces an infinite family of binary classifiers $g_{s,t} : x \mapsto 2 \cdot \mathbb{I}\{s(x) > t\} - 1$ indexed by thresholds $t \in \mathbb{R}$. The following proposition shows that one of those classifiers satisfies a fairness constraint as soon as appropriate group-wise ROC curves are equal for some value $\alpha \in [0, 1]$.

Proposition 10.6. *Under appropriate conditions on the score function s (i.e., $s \in \mathcal{S}$ where $\mathcal{S}$ satisfies Assumption 10.9), we have that:*

- *If $p_0 = p_1$ and s satisfies:*

$$\text{ROC}_{H_s^{(0)},G_s^{(0)}}(\alpha) = \text{ROC}_{H_s^{(1)},G_s^{(1)}}(\alpha), \tag{10.20}$$

  for some $\alpha \in [0, 1]$, then there exists $(t_0, t_1) \in (0, T)^2$, s.t. $M^{(0)}(g_{s,t_0}) = M^{(1)}(g_{s,t_1})$, which resembles parity in mistreatment (see Eq. 10.1).

- *If s satisfies:*

$$\text{ROC}_{H_s,G_s^{(0)}}(\alpha) = \text{ROC}_{H_s,G_s^{(1)}}(\alpha), \tag{10.21}$$

  for some $\alpha \in [0, 1]$, then $g_{s,t}$ satisfies fairness in FNR (see 10.3) for some threshold $t \in (0, T)$.

- *If s satisfies:*

$$\mathrm{ROC}_{H_s^{(0)},G_s}(\alpha) = \mathrm{ROC}_{H_s^{(1)},G_s}(\alpha), \tag{10.22}$$

for some $\alpha \in [0,1]$, then $g_{s,t}$ satisfies parity in FPR (see 10.2) for some threshold $t \in (0,T)$.

Proof. We go over each case.

Case of Eq. (10.20). Eq. (10.20) also writes:

$$G_s^{(0)} \circ \left(H_s^{(0)}\right)^{-1}(\alpha) = G_s^{(1)} \circ \left(H_s^{(1)}\right)^{-1}(\alpha),$$

Introduce $t_z = (H_s^{(z)})^{-1}(\alpha)$ then $G_s^{(z)}(t_z) = H_s^{(z)}(t_z) = \alpha$ for any $z \in \{0,1\}$, since $H_s^{(z)}$ is increasing. Also,

$$\begin{aligned} M^{(z)}(g_{s,t_z}) &= \mathbb{P}\{g_{s,t_z}(X) \neq Y \mid Z = z\}, \\ &= p_z G_s^{(z)}(t_z) + (1-p_z)(1 - H_s^{(z)}(t_z)), \\ &= (2\alpha - 1)p_z + (1-\alpha), \end{aligned}$$

which implies the result.

Case of Eq. (10.21). Eq. (10.21) also writes:

$$G_s^{(0)} \circ H_s^{-1}(\alpha) = G_s^{(1)} \circ H_s^{-1}(\alpha),$$

which translates to:

$$G^{(0)}\left(s(X) \leqslant H_s^{-1}(\alpha)\right) = G^{(1)}\left(s(X) \leqslant H_s^{-1}(\alpha)\right),$$

hence $g_{s,t}$ satisfies fairness in FNR (Eq. (10.3)) for the threshold $t = H_s^{-1}(\alpha)$.

Case of Eq. (10.22). Eq. (10.22) also writes:

$$G_s \circ (H_s^{(0)})^{-1}(\alpha) = G_s \circ (H_s^{(1)})^{-1}(\alpha),$$

which implies, since G_s, $H_s^{(0)}$ and $H_s^{(1)}$ are increasing:

$$H_s^{(0)} \circ (H_s^{(0)})^{-1}(\alpha) = H_s^{(1)} \circ (H_s^{(0)})^{-1}(\alpha),$$

and:

$$H^{(0)}\left(s(X) > (H_s^{(0)})^{-1}(\alpha)\right) = H^{(1)}\left(s(X) > (H_s^{(0)})^{-1}(\alpha)\right),$$

hence $g_{s,t}$ satisfies fairness in FPR (Eq. (10.2)) for the threshold $t = (H_s^{(0)})^{-1}(\alpha)$. □

Relation with AUC-based fairness. For continuous ROCs, the equality between their two AUCs implies that the two ROCs intersect at some unknown point, as shown by Proposition 10.7 (a simple consequence of the mean value theorem), proven below. Theorem 3.3 in Borkan et al. (2019) corresponds to the special case of Proposition 10.7 when $h = g, h' \neq g'$.

Proposition 10.7. *Let h, g, h', g' be c.d.f.s on $\mathbb{R}$ such that $\mathrm{ROC}_{h,g}$ and $\mathrm{ROC}_{h',g'}$ are continuous. If $\mathrm{AUC}_{h,g} = \mathrm{AUC}_{h',g'}$, then there exists $\alpha \in (0,1)$, such that $\mathrm{ROC}_{h,g}(\alpha) = \mathrm{ROC}_{h',g'}(\alpha)$.*

Proof. Consider $\psi : [0,1] \mapsto [-1,1]$: $\psi(\alpha) = \mathrm{ROC}_{h,g}(\alpha) - \mathrm{ROC}_{h',g'}(\alpha)$, it is continuous, hence integrable, and with:

$$\Psi(t) = \int_0^t \psi(\alpha) dt,$$

Note that $\Psi(1) = \mathrm{AUC}_{h,g} - \mathrm{AUC}_{h',g'} = 0 = \Psi(0)$. The mean value theorem implies that there exists $\alpha \in (0,1)$ such that:

$$\mathrm{ROC}_{h,g}(\alpha) = \mathrm{ROC}_{h',g'}(\alpha).$$

□

Proposition 10.7, combined with Proposition 10.6, implies that when a scoring function s satisfies some AUC-based fairness constraint, there exists a threshold $t \in \mathbb{R}$ inducing a non-trivial classifier $g_{s,t} := \text{sign}(s(x) - t)$ that satisfies some notion of fairness for classification at some unknown threshold t. For example, it is straightforward from Proposition 10.6 and Proposition 10.7 that:

- Eq. (10.4) implies parity in mistreatment for some thresholds,
- Eq. (10.5), Eq. (10.7) and Eq. (10.13) all imply parity in FNR for some threshold,
- Eq. (10.6) implies parity in FPR for some threshold.

The principal drawback of AUC-based fairness constraints is that it guarantees the existence of a single (unknown) t for which the fair binary classification properties are verified by $g_{s,t}$, and that the corresponding ROC point α cannot be easily controlled.

Relation with ROC-based fairness. In contrast to AUC-based fairness, ROC-based fairness allows to directly control the points α in Proposition 10.6 at which one obtains fair classifiers as it precisely consists in enforcing equality of $\text{ROC}_{G_s^{(0)}, G_s^{(1)}}$ and $\text{ROC}_{H_s^{(0)}, H_s^{(1)}}$ at specific points.

Furthermore, one can impose the equalities Eq. (10.20), Eq. (10.21) and Eq. (10.22) for several values of α such that thresholding the score behaves well for many critical situations. Specifically, under Assumption 10.9, we prove in Proposition 10.8 below that pointwise constraints over a discretization of the interval of interest approximates its satisfaction on the whole interval. This behavior, confirmed by our empirical results (see Sections 10.5 and 10.5.3), is relevant for many real-world problems that requires fairness in binary classification to be satisfied for a whole range of thresholds t in a specific region. For instance, in biometric verification, one is interested in low false positive rates (*i.e.*, large thresholds t). We refer to Grother and Ngan (2019) for an evaluation of the fairness of facial recognition systems in the context of 1:1 verification.

Proposition 10.8. *Under Assumption 10.9, if there exists $F \in \{H, G\}$ s.t. for every $k \in \{1, \dots, m_F\}$, $|\Delta_{F,\alpha_F^{(k)}}(s)| \leqslant \epsilon$, then:*

$$\sup_{\alpha \in [0,1]} \left|\Delta_{F,\alpha}(s)\right| \leqslant \epsilon + \frac{B+b}{2b} \max_{k \in \{0,\dots,m\}} \left|\alpha_F^{(k+1)} - \alpha_F^{(k)}\right|,$$

with the convention $\alpha_F^{(0)} = 0$ and $\alpha_F^{(m_F+1)} = 1$.

Proof. For any $F \in \{H, G\}$, note that:

$$\sup_{\alpha \in [0,1]} |\Delta_{F,\alpha}(s)| \leqslant \max_{k \in \{0,\dots,m\}} \sup_{x \in [\alpha_F^{(k)}, \alpha_F^{(k+1)}]} |\Delta_{F,\alpha}(s)|.$$

$\text{ROC}_{F_s^{(0)}, F_s^{(1)}}$ is differentiable, and its derivative is bounded by B/b. Indeed, for any $K_1, K_2 \in \mathcal{K}$, since K_1 is continuous and increasing, the inverse function theorem implies that $(K_1)^{-1}$ is differentiable. It follows that $K_2 \circ K_1^{-1}$ is differentiable and that its derivative satisfies:

$$\left(K_2 \circ K_1^{-1}\right)' = \frac{K_2' \circ K_1^{-1}}{K_1' \circ K_1^{-1}} \leqslant \frac{B}{b}.$$

Let $k \in \{0, \dots, m\}$, and $\alpha \in [\alpha_F^{(k)}, \alpha_F^{(k+1)}]$. Since $\alpha \mapsto \Delta_{F,\alpha}(s)$ is continuously differentiable, then α simultaneously satisfies, with the assumption that $|\Delta_{F,\alpha_F^{(k)}}(s)| \leqslant \epsilon$ for any $k \in \{1, \dots, K\}$:

$$|\Delta_{F,\alpha}(s)| \leqslant \epsilon + \left(1 + \frac{B}{b}\right) \left|\alpha_F^{(k)} - \alpha\right| \quad \text{and} \quad |\Delta_{F,\alpha}(s)| \leqslant \epsilon + \left(1 + \frac{B}{b}\right) \left|\alpha - \alpha_F^{(k+1)}\right|,$$

which implies that $|\Delta_{F,\alpha}(s)| \leqslant \epsilon + (1 + B/b)\left|\alpha_F^{(k+1)} - \alpha_F^{(k)}\right|/2$.

Finally, we have shown that:

$$\sup_{\alpha \in [0,1]} |\Delta_{F,\alpha}(s)| \leqslant \epsilon + \frac{B+b}{2b} \max_{k \in \{0,\dots,m\}} \left|\alpha_F^{(k+1)} - \alpha_F^{(k)}\right|.$$

□

10.4.2 Learning with Pointwise ROC-based Fairness Constraints

To impose stricter fairness conditions, the ideal goal is to enforce the equality of the score distributions of the positives (resp. negatives) between the two groups, *i.e.* $G_s^{(0)} = G_s^{(1)}$ (resp. $H_s^{(0)} = H_s^{(1)}$). This stronger functional criterion can be expressed in terms of ROC curves. For $\alpha \in [0,1]$, consider the deviations between the *positive* (resp. *negative*) *inter-group* ROC*s* and the identity function:

$$\Delta_{G,\alpha}(s) := \mathrm{ROC}_{G_s^{(0)},G_s^{(1)}}(\alpha) - \alpha \qquad \big(\text{resp. } \Delta_{H,\alpha}(s) := \mathrm{ROC}_{H_s^{(0)},H_s^{(1)}}(\alpha) - \alpha\big).$$

The aforementioned condition of equality between the distribution of the positives (resp. negatives) of the two groups are equivalent to satisfying $\Delta_{G,\alpha}(s) = 0$ (resp. $\Delta_{H,\alpha}(s) = 0$) for any $\alpha \in [0,1]$. When both of those conditions are satisfied, all of the AUC-based fairness constraints covered by Eq. (10.9) are verified, as it is easy to see that $C_l(s) = 0$ for all $l \in \{1,\dots,5\}$. Furthermore, guarantees on the fairness of classifiers $g_{s,t}$ induced by s hold for all possible thresholds t. While this strong property is desirable, it is challenging to enforce in practice due to its functional nature, and in many cases it may only be achievable by completely jeopardizing the ranking performance.

We thus propose to implement the satisfaction of a *finite* number of fairness constraints on $\Delta_{H,\alpha}(s)$ and $\Delta_{G,\alpha}(s)$ for specific values of α. Let $m_H, m_G \in \mathbb{N}$ be the number of constraints for the negatives and the positives respectively, as well as $\alpha_H = [\alpha_H^{(1)},\dots,\alpha_H^{(m_H)}] \in [0,1]^{m_H}$ and $\alpha_G = [\alpha_G^{(1)},\dots,\alpha_G^{(m_G)}] \in [0,1]^{m_G}$ the points at which they apply (sorted in strictly increasing order). With the notation $\Lambda := (\alpha, \lambda_H, \lambda_G)$, we can introduce the criterion $L_\Lambda(s)$, defined as:

$$L_\Lambda(s) = \mathrm{AUC}_{H_s,G_s} - \textstyle\sum_{k=1}^{m_H} \lambda_H^{(k)} |\Delta_{H,\alpha_H^{(k)}}(s)| - \sum_{k=1}^{m_G} \lambda_G^{(k)} |\Delta_{G,\alpha_G^{(k)}}(s)|,$$

$\lambda_H = [\lambda_H^{(1)},\dots,\lambda_H^{(m_H)}] \in \mathbb{R}_+^{m_H}$, $\lambda_G = [\lambda_G^{(1)},\dots,\lambda_G^{(m_G)}] \in \mathbb{R}_+^{m_G}$ being trade-off hyperparameters.

This criterion is flexible enough to address the limitations of AUC-based constraints outlined above. In particular, a practitioner can choose the points in α_H and α_G so as to guarantee the fairness of classifiers obtained by thresholding the scoring function at the desired trade-offs between false negative/false positive rates. Furthermore, in applications where the threshold used at deployment can vary in a whole interval, as in biometric verification (Grother and Ngan, 2019), we show in Proposition 10.8 under some regularity assumption on the ROC curve (see Assumption 10.9 in Section 10.4.3), if a small number of fairness constraints m_F are satisfied at discrete points of the interval for $F \in \{H, G\}$, then one obtains guarantees in sup norm on $\alpha \mapsto \Delta_{F,\alpha}$ (and therefore fair classifiers) in the entire interval.

10.4.3 Statistical Guarantees and Training Algorithm

We now prove statistical guarantees for the maximization of $\widehat{L}_\Lambda(s)$, the empirical counterpart of L_Λ:

$$\widehat{\mathrm{AUC}}_{H_s,G_s} - \textstyle\sum_{k=1}^{m_H} \lambda_H^{(k)} \big|\widehat{\Delta}_{H,\alpha_H^{(k)}}(s)\big| - \sum_{k=1}^{m_G} \lambda_G^{(k)} \big|\widehat{\Delta}_{G,\alpha_G^{(k)}}(s)\big|,$$

where $\widehat{\Delta}_{H,\alpha}(s) = \widehat{\mathrm{ROC}}_{G_s^{(0)},G_s^{(1)}}(\alpha) - \alpha$ and $\widehat{\Delta}_{G,\alpha}(s) = \widehat{\mathrm{ROC}}_{H_s^{(0)},H_s^{(1)}}(\alpha) - \alpha$ for any $\alpha \in [0,1]$. We denote by s_Λ^* the maximizer of L_Λ over $\mathcal{S}$, and $\widehat{s}_\Lambda$ the maximizer of $\widehat{L}_\Lambda$ over $\mathcal{S}$.

Our analysis relies on the following regularity assumption on the ROC curve.

Assumption 10.9. *The class $\mathcal{S}$ of scoring functions take values in $(0,T)$ for some $T > 0$, and the family of c.d.f.s $\mathcal{K} = \{G_s^{(z)}, H_s^{(z)} : s \in \mathcal{S}, z \in \{0,1\}\}$ satisfies: (a) any $K \in \mathcal{K}$ is continuously differentiable, and (b) there exists $b, B > 0$ s.t. $\forall (K,t) \in \mathcal{K} \times (0,T)$, $b \leqslant |K'(t)| \leqslant B$. The latter condition is satisfied when scoring functions do not have flat or steep parts, see Clémençon and Vayatis (2007) (Remark 7) for a discussion.*

Theorem 10.10. *Under Assumption 10.9 and those of Theorem 10.5, for any $\delta > 0$, $n > 1$, w.p. $\geqslant 1 - \delta$:*

$$\epsilon^2 \cdot [L_\Lambda(s_\Lambda^*) - L_\Lambda(\widehat{s}_\Lambda)] \leqslant C\,(1/2 + 2\epsilon C_{\Lambda,\mathcal{K}})\sqrt{\tfrac{V}{n}} + 2\epsilon\,(1 + 3C_{\Lambda,\mathcal{K}})\sqrt{\tfrac{\log(19/\delta)}{n-1}} + O(n^{-1}),$$

where $C_{\Lambda,\mathcal{K}} = (1 + B/b)(\bar{\lambda}_H + \bar{\lambda}_G)$, *with* $\bar{\lambda}_H = \sum_{k=1}^{m_H} \lambda_H^{(k)}$ *and* $\bar{\lambda}_G = \sum_{k=1}^{m_G} \lambda_G^{(k)}$.

Proof. Usual arguments imply that: $L_\Lambda(s_\Lambda^*) - L_\Lambda(\widehat{s}_\Lambda) \leqslant 2 \cdot \sup_{s\in\mathcal{S}} \left|\widehat{L}_\Lambda(s) - L_\Lambda(s)\right|$. As for Theorem 10.5, the triangle inequality implies that:

$$\begin{aligned}\left|\widehat{L}_\Lambda(s) - L_\Lambda(s)\right| &\leqslant \left|\widehat{\mathrm{AUC}}_{H_s,G_s} - \mathrm{AUC}_{H_s,G_s}\right| + \sum_{F\in\{F,G\}} \sum_{k=1}^{m_F} \lambda_F^{(k)} \left|\left|\widehat{\Delta}_{F,\alpha_k}(s)\right| - |\Delta_{F,\alpha_k}(s)|\right|, \\ &\leqslant \left|\widehat{\mathrm{AUC}}_{H_s,G_s} - \mathrm{AUC}_{H_s,G_s}\right| + \sum_{F\in\{F,G\}} \sum_{k=1}^{m_F} \lambda_F^{(k)} \left|\widehat{\Delta}_{F,\alpha_k}(s) - \Delta_{F,\alpha_k}(s)\right|.\end{aligned}$$

It follows that:

$$\begin{aligned}\sup_{s\in\mathcal{S}}\left|\widehat{L}_\Lambda(s) - L_\Lambda(s)\right| \leqslant \sup_{s\in\mathcal{S}} \left|\widehat{\mathrm{AUC}}_{H_s,G_s} - \mathrm{AUC}_{H_s,G_s}\right| + \bar{\lambda}_H \cdot \sup_{s,\alpha\in\mathcal{S}\times[0,1]} \left|\widehat{\Delta}_{H,\alpha}(s) - \Delta_{H,\alpha}(s)\right| \\ + \bar{\lambda}_G \cdot \sup_{s,\alpha\in\mathcal{S}\times[0,1]} \left|\widehat{\Delta}_{G,\alpha}(s) - \Delta_{G,\alpha}(s)\right|,\end{aligned}$$

and each of the terms is studied independently. The first term is already dealt with for Theorem 10.5, and the second and third terms have the same nature, hence we choose to focus on $\widehat{\Delta}_{G,\alpha}(s) - \Delta_{G,\alpha}(s)$. Note that:

$$\begin{aligned}&\widehat{\Delta}_{G,\alpha}(s) - \Delta_{G,\alpha}(s), \\ &= \widehat{\mathrm{ROC}}_{G_s^{(0)},G_s^{(1)}}(\alpha) - \mathrm{ROC}_{G_s^{(0)},G_s^{(1)}}(\alpha), \\ &= \underbrace{\left[G_s^{(1)} \circ \left(G_s^{(0)}\right)^{-1} - G_s^{(1)} \circ \left(\widehat{G}_s^{(0)}\right)^{-1}\right](1-\alpha)}_{T_1(s,\alpha)} + \underbrace{\left[G_s^{(1)} \circ \left(\widehat{G}_s^{(0)}\right)^{-1} - \widehat{G}_s^{(1)} \left(\widehat{G}_s^{(0)}\right)^{-1}\right](1-\alpha)}_{T_2(s,\alpha)}.\end{aligned}$$

Hence:

$$\sup_{s,\alpha\in\mathcal{S}\times[0,1]}\left|\widehat{\Delta}_{G,\alpha}(s) - \Delta_{G,\alpha}(s)\right| \leqslant \sup_{s,\alpha\in\mathcal{S}\times[0,1]} |T_1(s,\alpha)| + \sup_{s,\alpha\in\mathcal{S}\times[0,1]} |T_2(s,\alpha)|,$$

and we study each of these two terms independently.

Dealing with $\sup_{s,\alpha\in\mathcal{S}\times[0,1]}|T_1(s,\alpha)|$. Introduce the following functions, for any $z \in \{0,1\}$:

$$\widehat{U}_{n,s}^{(z)}(t) := \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}\{Y_i = +1, Z_i = z, s(X_i) \leqslant t\} \quad \text{and} \quad U_{n,s}^{(z)}(t) := \mathbb{E}\left[\widehat{U}_{n,s}^{(z)}(t)\right],$$

then $\widehat{G}_s^{(z)}(t) = (n/n_+^{(z)}) \cdot \widehat{U}_{n,s}^{(z)}(t)$ and $G_s^{(z)}(t) = (1/q_z p_z) \cdot U_{n,s}^{(z)}(t)$ for any $t \in (0,T)$.

The properties of the generalized inverse of a composition of functions (see van der Vaart (2000), Lemma 21.1, page 304 therein) give, for any $u \in [0,1]$:

$$\left(\widehat{G}_s^{(0)}\right)^{-1}(u) = \left(\widehat{U}_{n,s}^{(0)}\right)^{-1}\left(\frac{n_+^{(0)}u}{n}\right). \tag{10.23}$$

The assumption on $\mathcal{K}$ implies that $G_s^{(0)}$ is increasing. Define $k_s^{(0)} = G_s^{(0)} \circ s$, for any $t \in (0,T)$, we have:

$$\widehat{U}_{n,s}^{(0)}(t) = \widehat{U}_{n,k_s^{(0)}}\left(G_s^{(0)}(t)\right). \tag{10.24}$$

Combining Eq. (10.23) and Eq. (10.24), we have, for any $u \in [0,1]$:

$$\left(\widehat{G}_s^{(0)}\right)^{-1}(u) = \left(G_s^{(0)}\right)^{-1} \circ \left(\widehat{U}_{n,k_s^{(0)}}^{(0)}\right)^{-1}\left(\frac{n_+^{(0)}u}{n}\right).$$

Since $G_s^{(0)}$ is continuous and increasing, the inverse function theorem implies that $(G_s^{(0)})^{-1}$ is differentiable. It follows that:

$$\frac{d}{du}\left(G_s^{(1)} \circ (G_s^{(0)})^{-1}(u)\right) = \frac{\left(G_s^{(1)}\right)'\left((G_s^{(0)})^{-1}(u)\right)}{\left(G_s^{(0)}\right)'\left((G_s^{(0)})^{-1}(u)\right)} \leqslant \frac{B}{b},$$

and the mean value inequality implies:

$$\sup_{s,\alpha \in \mathcal{S}\times[0,1]} |T_1(s,\alpha)| \leqslant (B/b) \cdot \sup_{s,\alpha \in \mathcal{S}\times[0,1]} \left| \left(\widehat{U}^{(0)}_{n,k_s^{(0)}}\right)^{-1}\left(\frac{n_+^{(0)}\alpha}{n}\right) - \alpha \right|.$$

Conditioned upon the Z_i's and Y_i's, the quantity

$$\sqrt{n}\left(\left(\frac{n}{n_+^{(0)}}\right)\widehat{U}^{(0)}_{n,k_s^{(0)}}(\alpha) - \alpha\right),$$

is a standard empirical process, and it follows from Shorack and Wellner (1989) (page 86 therein), that:

$$\sup_{\alpha\in[0,1]} \left| \left(\widehat{U}^{(0)}_{n,k_s^{(0)}}\right)^{-1}\left(\frac{n_+^{(0)}\alpha}{n}\right) - \alpha \right| = \sup_{\alpha\in[0,1]} \left| \frac{n}{n_+^{(0)}}\widehat{U}^{(0)}_{n,k_s^{(0)}}(\alpha) - \alpha \right|.$$

Similar arguments as those seen in Theorem 10.5 imply:

$$\begin{aligned}\sup_{s,\alpha \in \mathcal{S}\times[0,1]} |T_1(s,\alpha)| &\leqslant (B/b) \cdot \sup_{s,\alpha \in \mathcal{S}\times[0,1]} \left| \frac{n}{n_+^{(0)}}\widehat{U}^{(0)}_{n,k_s^{(0)}}(\alpha) - \alpha \right|, \\ &\leqslant \frac{B}{bq_0p_0} \cdot \left| \frac{n_+^{(0)}}{n} - q_0p_0 \right| + \frac{B}{bq_0p_0} \cdot \sup_{s,\alpha \in \mathcal{S}\times[0,1]} \left| \widehat{U}^{(0)}_{n,k_s^{(0)}}(\alpha) - q_0p_0\alpha \right|,\end{aligned}$$

A standard learning bound (see Boucheron et al. (2005), Theorem 3.2 and 3.4 page 326-328 therein) implies that: for any $\delta > 0, n > 0$, *w.p.* $\geqslant 1-\delta$,

$$\sup_{s,\alpha \in \mathcal{S}\times[0,1]} \left| \widehat{U}^{(0)}_{n,k_s^{(0)}}(\alpha) - U^{(0)}_{n,k_s^{(0)}}(\alpha) \right| \leqslant C\sqrt{\frac{V}{n}} + \sqrt{\frac{2\log(2/\delta)}{n}}, \tag{10.25}$$

where C is an universal constant.

A union bound between Eq. (10.25) and a standard Hoeffding inequality for $n_+^{(0)}$ gives: for any $\delta > 0, n > 1$, *w.p.* $\geqslant 1-\delta$,

$$\sup_{s\in\mathcal{S}} |T_1(s,\alpha)| \leqslant \frac{BC}{bq_0p_0}\sqrt{\frac{V}{n}} + \frac{3B}{bq_0p_0}\sqrt{\frac{\log(4/\delta)}{2n}}. \tag{10.26}$$

Dealing with $\sup_{s,\alpha\in\mathcal{S}\times[0,1]} |T_2(s,\alpha)|$. We recall that $\widehat{G}_s^{(z)}(t) = (n/n_+^{(z)}) \cdot \widehat{U}_{n,s}^{(z)}(t)$ and $G_s^{(z)}(t) = (1/q_zp_z) \cdot U_{n,s}^{(z)}(t)$ for any $t \in (0,T)$.

First note that, using the same type of arguments as for Theorem 10.5:

$$\begin{aligned}\sup_{s,\alpha \in \mathcal{S}\times[0,1]} |T_2(s,\alpha)| &\leqslant \sup_{s,t \in \mathcal{S}\times(0,T)} \left| \widehat{G}_s^{(1)}(t) - G_s^{(1)}(t) \right|, \\ &\leqslant \frac{1}{q_1p_1}\left| \frac{n_+^{(1)}}{n} - q_1p_1 \right| + \frac{1}{q_1p_1} \cdot \sup_{s,t \in \mathcal{S}\times(0,T)} \left| \widehat{U}_{n,s}^{(1)}(t) - U_{n,s}^{(1)}(t) \right|.\end{aligned}$$

The same arguments as for Eq. (10.25) apply, which means that: for any $\delta > 0, n > 0$, *w.p.* $\geqslant 1-\delta$,

$$\sup_{s,t \in \mathcal{S}\times(0,T)} \left| \widehat{U}_{n,s}^{(1)}(t) - U_{n,s}^{(1)}(t) \right| \leqslant C\sqrt{\frac{V}{n}} + \sqrt{\frac{2\log(2/\delta)}{n}}, \tag{10.27}$$

where C is an universal constant.

A union bound of Eq. (10.27) and a standard Hoeffding inequality for $n_+^{(1)}$ finally imply that: for any $\delta > 0, n > 1$, *w.p.* $\geqslant 1-\delta$,

$$\sup_{s\in\mathcal{S}}|T_2(s,\alpha)| \leqslant \frac{C}{q_1 p_1}\sqrt{\frac{V}{n}} + \frac{3}{q_1 p_1}\sqrt{\frac{\log(4/\delta)}{2n}}. \tag{10.28}$$

Conclusion.

Combining Eq. (10.26) and Eq. (10.28), one obtains that: for any $\delta > 0, n > 1$, *w.p.* $\geqslant 1-\delta$,

$$\sup_{s,\alpha\in\mathcal{S}\times[0,1]}\left|\widehat{\Delta}_{G,\alpha}(s) - \Delta_{G,\alpha}(s)\right| \leqslant C\left(\frac{1}{q_1 p_1} + \frac{B}{bq_0 p_0}\right)\sqrt{\frac{V}{n}} + \left(\frac{3}{q_1 p_1} + \frac{3B}{bq_0 p_0}\right)\sqrt{\frac{\log(8/\delta)}{2n}}. \tag{10.29}$$

and a result with similar form can be shown for $\sup_{s,\alpha\in\mathcal{S}\times[0,1]}\left|\widehat{\Delta}_{H,\alpha}(s) - \Delta_{H,\alpha}(s)\right|$ by following the same steps.

Under the assumption $\min_{z\in\{0,1\}}\min_{y\in\{-1,1\}}\mathbb{P}\{Y=y, Z=z\} \geqslant \epsilon$, a union bound between Eq. (10.29), its equivalent for $\widehat{\Delta}_{H,\alpha}$ and Eq. (10.16) gives, with the upper-bound $1/(2n) \leqslant 1/(n-1)$: for any $\delta > 0, n > 1$, *w.p.* $\geqslant 1-\delta$,

$$\begin{aligned}\epsilon^2 \cdot (L_\Lambda(s^*_\Lambda) - L_\Lambda(\widehat{s}_\Lambda)) \leqslant 2\epsilon &\left(1 + 3(\bar{\lambda}_H + \bar{\lambda}_G)\left[1 + \frac{B}{b}\right]\right)\sqrt{\frac{\log(19/\delta)}{n-1}} \\ &+ C\left(\frac{1}{2} + 2\epsilon(\bar{\lambda}_H + \bar{\lambda}_G)\left[1 + \frac{B}{b}\right]\right)\sqrt{\frac{V}{n}} + O(n^{-1}),\end{aligned}$$

which concludes the proof. □

Theorem 10.10 generalizes the learning rate of $O(1/\sqrt{n})$ of Theorem 10.5 to ranking under ROC-based constraints. Like Theorem 10.5, its proof relies on results on U-processes, but further requires a study of the deviations of the empirical ROC curve seen as ratios of empirical processes indexed by $\mathcal{S}\times[0,1]$. In that regard, our analysis builds upon the decomposition proposed in Hsieh and Turnbull (1996), which enables the derivation of uniform bounds over $\mathcal{S}\times[0,1]$ from results on standard empirical processes (van der Vaart and Wellner, 1996).

10.4.4 Training Algorithm

The approach presented here follows the same broad principles as that of Section 10.3.3. First, we define an approximation of the quantities $\widehat{H}_s^{(z)}, \widehat{G}_s^{(z)}$ on $\mathcal{B}_N$, for any $z \in \{0,1\}$, as:

$$\begin{aligned}\widetilde{H}_s^{(z)}(t) &= \frac{1}{N_-^{(z)}}\sum_{i=1}^N \mathbb{I}\{y_i = -1, z_i = z\}\cdot\sigma(t - s(x_i)),\\ \widetilde{G}_s^{(z)}(t) &= \frac{1}{N_+^{(z)}}\sum_{i=1}^N \mathbb{I}\{y_i = +1, z_i = z\}\cdot\sigma(t - s(x_i)),\end{aligned}$$

which can be respectively seen as relaxations of the false positive rate (*i.e.* $\bar{H}_s^{(z)}(t) = 1 - H_s^{(z)}(t)$) and true positive rate (*i.e.* $\bar{G}_s^{(z)}(t) = 1 - G_s^{(z)}(t)$) at threshold t and conditioned upon $Z = z$.

For any $F \in \{H, G\}, k \in \{1, \ldots, m_F\}$, we introduce a loss ℓ_F^k which gradients are meant to enforce the constraint $|\widehat{\Delta}_{F,\alpha_F^{(k)}}(s)| = 0$. This constraint can be seen as one that imposes equality between the true positive rates and false positive rates for the problem of discriminating between the negatives (resp. positives) of sensitive group 1 against those of sensitive group 0 when $F = H$ (resp. $F = G$). An approximation of this problem's false positive rate (resp. true positive rate) at threshold t is $\widetilde{F}_s^{(0)}(t)$ (resp. $\widetilde{F}_s^{(1)}(t)$). Introduce $c_F^{(k)}$ as a constant in $[-1,+1]$ and $t_F^{(k)}$ as a threshold in $\mathbb{R}$, the following loss $\ell_F^{(k)}$ seeks to equalize these two quantities at threshold $t_F^{(k)}$:

$$\ell_F^{(k)}(s) = c_F^{(k)}\cdot\left(\widetilde{F}_s^{(0)}\left(t_F^{(k)}\right) - \widetilde{F}_s^{(1)}\left(t_F^{(k)}\right)\right).$$

If the gap between $\widehat{F}_s^{(0)}(t_F^{(k)})$ and $\widehat{F}_s^{(1)}(t_F^{(k)})$ — evaluated on the validation set $\mathcal{V}_m$ — is not too large, the threshold $t_F^{(k)}$ is modified slightly every few iterations so that $\widehat{F}_s^{(0)}(t_F^{(k)})$ and $\widehat{F}_s^{(1)}(t_F^{(k)})$ both approach the target value $\alpha_F^{(k)}$. Otherwise, the parameter $c_F^{(k)}$ is slightly modified. The precise strategy to modify $c_F^{(k)}$ and $t_F^{(k)}$ is detailed in Algorithm 3, and we introduce a step Δt to modify the thresholds $t_F^{(k)}$.

The final loss writes:

$$\widetilde{L}_{\Lambda,c,t}(s) := \left(1 - \widetilde{\mathrm{AUC}}_{H_s,G_s}\right) + \sum_{F\in\{F,G\}} \left(\frac{1}{m_F}\sum_{k=1}^{m_F} \lambda_F^{(k)} \cdot \ell_F^{(k)}(s)\right) + \frac{\lambda_{\mathrm{reg}}}{2} \cdot \|W\|_2^2,$$

and one can define $\widetilde{L}_{\Lambda,c,t}^{(B)}$ by approximating $\widetilde{\mathrm{AUC}}_{H_s,G_s}$ above by $\widetilde{\mathrm{AUC}}_{H_s,G_s}^{(B)}$. The full algorithm is given in Algorithm 3.

Algorithm 3 Practical algorithm for learning with ROC-based constraints.

Input: training set $\mathcal{D}_n$, validation set $\mathcal{V}_m$

$c_F^{(k)} \leftarrow 0$ for any $F \in \{H, G\}$, $k \in \{1, \ldots, m_F\}$

$t_F^{(k)} \leftarrow 0$ for any $F \in \{H, G\}$, $k \in \{1, \ldots, m_F\}$

for $i = 1$ **to** n_{iter} **do**

  $\mathcal{B}_N \leftarrow N$ observations sampled with replacement from $\mathcal{D}_n$

  $s \leftarrow$ updated scoring function using a gradient-based algorithm (*e.g.* ADAM), using the derivative of $\widetilde{L}_{\Lambda,c,t}^{(B)}(s)$ on $\mathcal{B}_N$

  if $(n_{\mathrm{iter}} \bmod n_{\mathrm{adapt}}) = 0$ **then**

    for any $F \in \{H, G\}$, $k \in \{1, \ldots, m_F\}$ **do**

      $\Delta_F^{(k)} \leftarrow \widehat{F}_s^{(0)}\left(t_F^{(k)}\right) - \widehat{F}_s^{(1)}\left(t_F^{(k)}\right)$ computed on $\mathcal{V}_m$

      $\Sigma_F^{(k)} \leftarrow \widehat{F}_s^{(0)}\left(t_F^{(k)}\right) + \widehat{F}_s^{(1)}\left(t_F^{(k)}\right) - 2\alpha_F^{(k)}$ computed on $\mathcal{V}_m$

      if $\left|\Sigma_F^{(k)}\right| > \left|\Delta_F^{(k)}\right|$ **then**

        $t_F^{(k)} \leftarrow t_F^{(k)} + \mathrm{sgn}\left(\Sigma_F^{(k)}\right) \cdot \Delta t$

      else

        $c_F^{(k)} \leftarrow c_F^{(k)} + \mathrm{sgn}\left(\Delta_F^{(k)}\right) \cdot \Delta c$

        $c_F^{(k)} \leftarrow \min\left(1, \max\left(-1, c_F^{(k)}\right)\right)$

      end if

    end for

  end if

end for

Output: scoring function s

10.5 Experiments

We have experimented with our algorithms on real and synthetic data, with various AUC and ROC-based fairness constraints. This section covers extensively all results and details about our experimental setup. In general, our empirical results show that our approaches consistently balance ranking performance and the chosen notion of fairness.

10.5.1 Experimental Details

Scoring function and optimization. We used a simple neural network of various depth D ($D = 0$ corresponds to linear scoring function, while $D = 2$ corresponds to a network of 2 hidden layers) where each layer has the same width d (the dimension of the input space), except for the output layer which outputs a real score. We used ReLU's as activation functions. To center and scale the output score we used *batch normalization* (BN) (see Goodfellow et al., 2016, Section 8.7.1 therein) with fixed values $\gamma = 1, \beta = 0$ for the output value of the network. Algorithm 4 gives a formal description of the network architecture. The intuition for normalizing the output score is that the ranking losses only depend on the relative value of the score between instances, and the more *classification-oriented* losses of ROC-based constraints only depend on a threshold on the score. Empirically, we observed the necessity of renormalization for the algorithm with ROC-based constraints, as the loss $\ell_F^{(k)}$ is zero when $\widehat{F}_s^{(0)}(t_F^{(k)}) = \widehat{F}_s^{(1)}(t_F^{(k)}) \in \{0, 1\}$, which leads to scores that drift away from zero during the learning process, as it seeks to satisfy the constraint imposed by $\ell_F^{(k)}$. All of the network weights were initialized using a simple centered normal random variable with standard deviation 0.01.

Algorithm 4 Network architecture.

Input: observation $x = h_0'' \in \mathbb{R}^d$,

for $k = 1$ **to** D **do**

Linear layer: $h_k = W_k^\top h_{k-1}'' + b_k$ with $W_k \in \mathbb{R}^{d,d}, b_k \in \mathbb{R}^{d,1}$ learned by GD,

ReLu layer: $h_k'' = \max(0, h_k')$ where max is an element-wise maximum,

end for

Linear layer: $h_{D+1} = w_{D+1}^\top h_D'' + b_{D+1}$ with $w_{D+1} \in \mathbb{R}^{d,1}, b_{D+1} \in \mathbb{R}$ learned by GD,

BN layer: $h_{D+1}' = (h_{D+1} - \mu_{D+1})/\sigma_{D+1}$, with $\mu_{D+1} \in \mathbb{R}, \sigma_{D+1} \in \mathbb{R}$ running averages,

Output: score $s(x)$ of x, with $s(x) = h_{D+1}' \in \mathbb{R}$.

For both AUC-based and ROC-based constraints, optimization was done with the ADAM algorithm. It has an adaptative step size, so we did not modify its default parameters. Refer to Ruder (2016) for more details on gradient descent optimization algorithms.

Other details. For all experiments, we set aside 40% of the data for validation, *i.e.* $m = \lfloor 0.40(m+n) \rfloor$ with $\lfloor \cdot \rfloor$ the floor function, the batch size to $N = 100$ and the parameters of the loss changed every $n_{\text{adapt}} = 50$ iterations. For any sampling-based approximation computed on a batch $\mathcal{B}_N$, we set $B = 100$, and $B_{\text{v}} = 10^5$ for those on a validation set $\mathcal{V}_m$. The value Δc was always fixed to 0.01 and Δt to 0.001. We used linear scoring functions, *i.e.* $D = 0$, for the synthetic data experiments, and networks with $D = 2$ for real data.

The experiments were implemented in Python, and relied extensively on the libraries `numpy`, `TensorFlow` (Abadi et al., 2016), `scikit-learn` (Pedregosa et al., 2011) and `matplotlib` for plots.

10.5.2 Synthetic Data Experiments

The following examples introduce data distributions that we use to illustrate the relevance of our approach.

Example 10.11. *Let $\mathcal{X} = [0, 1]^2$. For any $x = (x_1\ x_2)^\top \in \mathcal{X}$, let $F^{(0)}(x) = F^{(1)}(x) = 1$, as well as $\eta^{(0)}(x) = x_1$ and $\eta^{(1)}(x) = x_2$. We have $F = 1$ (F is the uniform distribution) and $\eta(x) = q_0 x_1 + q_1 x_2$. Consider linear scoring functions of the form $s_c(x) = cx_1 + (1 - c)x_2$ parameterized by $c \in [0, 1]$. Fig. 10.1 plots AUC_{H_s, G_s} and $\text{AUC}_{H_s^{(z)}, G_s^{(z)}}$ for any $z \in \{0, 1\}$ as a function of c, illustrating the trade-off between fairness and ranking performance.*

Example 10.12. *Set* $\mathcal{X} = [0,1]^2$. *For any* $x \in \mathcal{X}$ *with* $x = (x_1\ x_2)^\top$, *set:*

$$F^{(0)}(x) = (16/\pi) \cdot \mathbb{I}\{x^2 + y^2 \leqslant 1/2\},$$
$$F^{(1)}(x) = (16/3\pi) \cdot \mathbb{I}\{1/2 \leqslant x^2 + y^2 \leqslant 1\},$$

and $\eta^{(0)}(x) = \eta^{(1)}(x) = (2/\pi) \cdot \arctan(x_2/x_1)$.

For all of the synthetic data experiments, our objective is to show that the learning procedure recovers the optimal scoring function when the dataset is large enough. Each of the 100 runs that we perform uses independently generated train, validation and test datasets. The variation that we report on 100 runs hence includes that of the data generation process, which is small since we use large samples. Precisely, for each run, we chose a total of $n + m = 10,000$ points for the train and validation sets and a test dataset of size $n_{\text{test}} = 20,000$. Both algorithms ran for $n_{\text{iter}} = 10,000$ iterations, and with the same regularization strength $\lambda_{\text{reg}} = 0.01$.

Example 10.11. First, we illustrate learning with the AUC constraint in Eq. (10.4) on the simple problem in Example 10.11. Our experiment shows that we can effectively find trade-offs between ranking accuracy and satisfying Eq. (10.4) using the procedure described in Algorithm 2.

The final solutions of Algorithm 2 with two different values of λ, parameterized by c, are shown in Fig. 10.2. A representation of the value of the corresponding scoring functions on $[0,1] \times [0,1]$ is provided in Fig. 10.3. The median ROC curves for two values of λ over 100 independent runs are shown in Fig. 10.4, with pointwise 95% confidence intervals.

Example 10.12. Example 10.12 allows to compare ROC-based and ROC-based approaches. The former uses Eq. (10.4) as constraint and the latter penalizes $\Delta_{H,3/4}(s) \neq 0$. The goal of our experiment with Example 10.12 is to show that Algorithm 3 can effectively learn a scoring function s for which the α corresponding to a classifier g_{s,t_α} that is fair in FPR is specified in advance, and that the solution can be significantly different from those obtained with AUC-based constraints, *i.e.* Algorithm 2.

In practice, we compare the solutions of optimizing the AUC without constraint, *i.e.* Algorithm 2 with $\lambda = 0$ with those of Algorithm 2 with $\lambda = 1$ and Algorithm 3 where we impose $\Delta_{H,3/4}(s) = 0$ with strength $\lambda_H = 1$. To illustrate the results, we introduce the following family of scoring functions $s_c(x) = -c \cdot x_1 + (1-c) \cdot x_2$, parameterized by $c \in [0,1]$.

In practice, we observe that the different constraints lead to scoring functions with specific trade-offs between fairness and performance, as summarized in Table 10.2. Results with AUC-based fairness are the same for $\lambda = 0$ and $\lambda = 1$ because the optimal scoring function for ranking satisfies Eq. (10.4).

Fig. 10.5 shows that the AUC-based constraint has no effect on the solution, unlike the ROC-based constraint which is successfully enforced by Algorithm 3. Fig. 10.6 gives two possible scoring functions with Algorithm 3. The median ROC curves for two values of λ_H over 100 independent runs are shown in Fig. 10.7, with pointwise 95% confidence intervals.

10.5.3 Real Data Experiments

Datasets. We evaluate our algorithms on four datasets that have been commonly used in the fair machine learning literature. Those are the following:

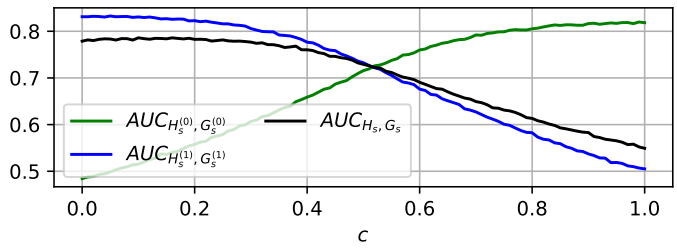


Figure 10.1: Plotting Example 10.11 for $q_1 = 17/20$. Under the fairness definition Eq. (10.4), a fair solution exists for $c = 1/2$, but the ranking performance for $c < 1/2$ is significantly higher.

Table 10.2: Results on the test set, averaged over 100 runs (std. dev. are all smaller than 0.02).

Method	AUC-based fairness					ROC-based fairness		
Value of λ	$\lambda = 0$		$\lambda > 0$			$\lambda_H^{(k)} = \lambda_H > 0$		
	AUC	ΔAUC	AUC	ΔAUC	$\lvert\Delta_{H,3/4}\rvert$	AUC	ΔAUC	$\lvert\Delta_{H,3/4}\rvert$
Example 10.11	**0.79**	0.28	0.73	**0.00**	–	–	–	–
Example 10.12	**0.80**	**0.02**	**0.80**	**0.02**	0.38	0.75	0.06	**0.00**

- The *German Credit Dataset* (German), featured in Zafar et al. (2019); Zehlike et al. (2017); Singh and Joachims (2019); Donini et al. (2018), consists in classifying people described by a set of attributes as good or bad credit risks. The sensitive variable is the gender of the individual, *i.e.* male ($Z = 0$) or female ($Z = 1$). It contains 1,000 instances and we retain 30% of those for testing, and the rest for training/validation.

- The *Adult Income Dataset* (Adult), featured in Zafar et al. (2019); Donini et al. (2018), is based on US census data and consists in predicting whether income exceeds \$50K a year. The sensitive variable is the gender of the individual, *i.e.* male ($Z = 1$) or female ($Z = 0$). It contains 32.5K observations for training and validation, as well as 16.3K observations for testing. For simplicity, we removed the weights associated to each instance of the dataset.

- The *Compas Dataset* (Compas), featured in Zehlike et al. (2017); Donini et al. (2018), consists in predicting recidivism of convicts in the US. The sensitive variable is the race of the individual, precisely $Z = 1$ if the individual is categorized as African-American and $Z = 0$ otherwise. It contains 9.4K observations, and we retain 20% of those for testing, and the rest for training/validation.

- The *Bank Marketing Dataset* (Bank), featured in Zafar et al. (2019), consists in predicting whether a client will subscribe to a term deposit. The sensitive variable is the age of the individual: $Z = 1$ when the age is between 25 and 60 (which we refer to as "working age population") and $Z = 0$ otherwise. It contains 45K observations, of which we retain 20% for testing, and the rest for training/validation.

For all of the datasets, we used one-hot encoding for any categorical variables. The number of training instances $n + m$, test instances n_{test} and features d for each dataset is summarized in Table 10.3.

Table 10.3: Number of observations and feat d per dataset.

Dataset	German	Adult	Compas	Bank
$n + m$	700	32.5K	7.5K	36K
n_{test}	300	16.3K	1.9K	9K
d	61	107	16	59

Parameters. For Algorithm 2, we select different AUC-based fairness constraints depending on the dataset. In the case of *Compas* (recidivism prediction), being labeled positive is a disadvantage so the approach with AUC-based fairness uses the constraint in Eq. (10.6) to balance FPRs (by forcing the probabilities that a negative from a given group is mistakenly ranked higher than a positive to be the same across groups). Conversely for *German* (credit scoring), a positive label is an advantage, so we choose Eq. (10.5) to balance FNRs. For *Bank* and *Adult*, the problem has no clear connotation so we select Eq. (10.8) to force the same ranking accuracy when comparing the positives of a group with the negatives of another.

Inspired by the consideration that many operational settings focus on learning a good score for small FPR rates, the ROC-based approach is configured to simultaneously align the distribution of FPR and TPR for low FPRs between both groups by penalizing solutions with high $|\Delta_{H,1/8}(s)|$, $|\Delta_{H,1/4}(s)|$, $|\Delta_{G,1/8}(s)|$ and $|\Delta_{G,1/4}(s)|$.

Precisely, for every run of Algorithm 3, we set:

$$m_G = m_H = 2, \quad \alpha_G^{(1)} = \alpha_H^{(1)} = \frac{1}{8}, \quad \alpha_G^{(1)} = \alpha_H^{(2)} = \frac{1}{4},$$
$$\lambda_G^{(1)} = \lambda_G^{(2)} = \lambda \quad \text{and} \quad \lambda_H^{(1)} = \lambda_H^{(2)} = \lambda.$$

For all algorithms, we chose the parameter λ from the candidate set $\in \{0, 0.25, 0.5, 1, 5, 10\}$, where $\lambda = 0$ corresponds to the case without constraint. Denoting by $\widetilde{s}$ the output of Algorithm 2 or Algorithm 3, we selected the parameter λ_{reg} of the L2 regularization that maximizes the criterion $L_\lambda(\widetilde{s})$ (resp. $L_\Lambda(\widetilde{s})$) on the validation dataset over the following candidate regularization strength set:

$$\lambda_{\text{reg}} \in \{1 \times 10^{-3}, 5 \times 10^{-3}, 1 \times 10^{-2}, 5 \times 10^{-2}, 1 \times 10^{-1}, 5 \times 10^{-1}, 1\}.$$

The selected parameters are summarized in Table 10.4. Results are summarized in Table 10.5, where AUC denotes the ranking accuracy AUC_{H_s,G_s}, and ΔAUC denotes the absolute difference of the terms in the AUC-based fairness constraint of interest. We also report on the values of $\left|\Delta_{F,1/8}\right|$ and $\left|\Delta_{F,1/4}\right|$ for any $F \in \{H, G\}$ and refer the reader to the ROC curves in Fig. 10.8 and Fig. 10.8 for a visual summary of the other values of $\Delta_{F,\alpha}$ with $F \in \{H, G\}$ and $\alpha \in [0, 1]$. We highlight in bold the best ranking accuracy, and the fairest algorithm for the relevant constraint. All of the numerical evaluations reported below are evaluations on the held-out test set.

Table 10.4: Parameters selected using the validation set for the runs on real data.

Parameters		Constraint		
Dataset	Variable	None	AUC	ROC
German	λ	0	0.25	0.25
	λ_{reg}	0.5	0.5	0.5
Adult	λ	0	0.25	0.25
	λ_{reg}	0.05	0.05	0.05
Compas	λ	0	0.5	0.25
	λ_{reg}	0.05	0.05	0.05
Bank	λ	0	0.25	0.25
	λ_{reg}	0.05	0.05	0.05

Results for the dataset *Compas*. *Compas* is a recidivism prediction dataset, where the sensitive variable is $Z = 1$ if the individual is categorized as African-American and 0 otherwise. As being labeled positive (i.e., recidivist) is a disadvantage, we consider the AUC-based constraint in Eq. (10.6) to force the probabilities that a negative from a given group is mistakenly ranked higher than a positive to be the same across groups. While the scoring function learned without fairness constraint systematically makes more ranking errors for non-recidivist African-Americans, we can see that the AUC-constraint achieves its goal as it makes the area under $\text{ROC}_{H_s^{(1)},G_s}$ and $\text{ROC}_{H_s^{(0)},G_s}$ very similar. We can however see that slightly more of such errors are still made in the first quartile of the scores. As an alternative to AUC-based fairness, we thus configure our ROC-based fairness constraints to align the distributions of positives and negatives across both groups by penalizing solutions with high $|\Delta_{G,1/8}(s)|$, $|\Delta_{G,1/4}(s)|$, $|\Delta_{H,1/8}(s)|$ and $|\Delta_{H,1/4}(s)|$. In line with our theoretical analysis (see the discussion in Section 10.4.2), we can see from $\text{ROC}_{G_s^{(0)},G_s^{(1)}}$ and $\text{ROC}_{H_s^{(0)},H_s^{(1)}}$ that it suffices to impose equality of the positive and negative distributions in the entire interval $[0, 1/4]$ of interest. In turn, $\text{ROC}_{H_s^{(1)},G_s}$ and $\text{ROC}_{H_s^{(0)},G_s}$ become essentially equal in this region as desired. Note that on this dataset, both the AUC and ROC constraints are achieved with minor impact on the ranking performance, as seen from the AUC scores.

Results for the dataset *Adult*. We now turn to *Adult*, an income prediction dataset where Z denotes the gender ($Z = 0$ for woman) and a positive label indicates that the person makes over \$50K/year. For this dataset, we plot $\text{ROC}_{H_s^{(1)},G_s^{(0)}}$ and $\text{ROC}_{H_s^{(0)},G_s^{(1)}}$ and observe that without fairness constraint, men who make less than \$50K are much more likely to be mistakenly ranked

Table 10.5: Results on test set. The strength of fairness constraints and regularization is chosen based on a validation set to obtain interesting trade-offs, as detailed in Section 10.5.3.

Measure		Dataset			
Constraint	Value	German	Adult	Compas	Bank
None	AUC	**0.76**	**0.91**	**0.72**	**0.94**
	ΔAUC	0.07	0.16	0.20	0.13
	$\lvert\Delta_{H,1/8}\rvert$	**0.01**	0.31	0.26	0.09
	$\lvert\Delta_{H,1/4}\rvert$	0.20	0.36	0.32	0.18
	$\lvert\Delta_{G,1/8}\rvert$	0.13	0.02	0.29	**0.00**
	$\lvert\Delta_{G,1/4}\rvert$	0.20	0.06	0.29	**0.04**
AUC-based	AUC	0.75	0.89	0.71	0.93
	ΔAUC	**0.05**	**0.02**	**0.00**	**0.05**
	$\lvert\Delta_{H,1/8}\rvert$	0.05	0.09	0.06	**0.03**
	$\lvert\Delta_{H,1/4}\rvert$	0.08	0.17	0.03	0.11
	$\lvert\Delta_{G,1/8}\rvert$	**0.01**	0.06	0.02	0.27
	$\lvert\Delta_{G,1/4}\rvert$	0.02	0.14	0.06	0.37
ROC-based	AUC	0.75	0.87	0.70	0.91
	ΔAUC	0.07	0.07	0.05	0.14
	$\lvert\Delta_{H,1/8}\rvert$	0.03	**0.06**	**0.01**	**0.03**
	$\lvert\Delta_{H,1/4}\rvert$	**0.07**	**0.01**	**0.02**	**0.05**
	$\lvert\Delta_{G,1/8}\rvert$	0.04	**0.00**	**0.00**	0.06
	$\lvert\Delta_{G,1/4}\rvert$	**0.01**	**0.02**	**0.00**	0.21

above a woman who actually makes more, than the other way around. The learned score thus reproduces a common gender bias. To fix this, the appropriate notion of AUC-based fairness is Eq. (10.8): we can see that it successfully equates the area under $\mathrm{ROC}_{H_s^{(1)},G_s^{(0)}}$ and $\mathrm{ROC}_{H_s^{(0)},G_s^{(1)}}$. Note however that this comes at the cost of introducing a small bias against men in the top scores. As can be seen from $\mathrm{ROC}_{H_s^{(0)},H_s^{(1)}}$ and $\mathrm{ROC}_{G_s^{(0)},G_s^{(1)}}$, positive women now have higher scores overall than positive men, while negative men have higher scores than negative women. These observations illustrate the limitations of AUC-based fairness discussed in Section 10.4. To address them, we use the same ROC constraints as we did in *Compas* so as to align the distributions of positives and negatives of each group in $[0, 1/4]$, which is again achieved almost perfectly in the entire interval. While the degradation in ranking performance is more noticeable on this dataset, a clear advantage from ROC-based fairness in both datasets is that the obtained scoring function can be thresholded to obtain fair classifiers at a wide range of thresholds.

Results for the dataset *Bank*. Recall that for this dataset we consider the AUC constraint Eq. (10.8) to force the same ranking accuracy when comparing the positives of a group with the negatives of another. Fig. 10.8 shows that the score learned without constraint implies a stochastic order between the distributions of the problem that writes $H_s^{(1)} \preceq H_s^{(0)} \preceq G_s^{(0)} \preceq G_s^{(1)}$, where $h \preceq g$ means that g is stochastically larger than h. This suggests that the task of distinguishing positives from negatives is much harder for observations of the group $Z = 0$ than for those of the working age population ($Z = 1$), which could be a consequence of the heterogeneity of the group $Z = 0$. On the other hand, the left plot representing $\mathrm{ROC}_{H_s^{(1)},G_s^{(0)}}$ and $\mathrm{ROC}_{H_s^{(0)},G_s^{(1)}}$ for the setting without constraint gives an appreciation of the magnitude of those differences. Precisely, it implies that it is much harder to distinguish working age positives ($Y = +1, Z = 1$) from negatives of group $Z = 0$ than working age negatives from positives of group $Z = 0$. The correction induced by the AUC constraint suggests that it was due to the fact that scores for positives of the group ($Y = +1, Z = 0$) were too small compared to the positives of the working age population ($Y = +1, Z = 1$). Indeed, learning with the AUC constraint roughly equalizes the scores of the positives across both groups $Z = 0$ and $Z = 1$. Additionally, in the left plot for learning with AUC constraints, we can see that $\mathrm{ROC}_{H_s^{(1)},G_s^{(0)}}$ and $\mathrm{ROC}_{H_s^{(0)},G_s^{(1)}}$ intersect and have similar AUC's as expected, which is more visible for the dashed lines (*i.e.* on training data). Finally, the ROC-based constraint induces as expected the equality of $G_s^{(0)}$ and $G_s^{(1)}$ as

well as that of $H_s^{(0)}$ and $H_s^{(1)}$ in the high score regime, as seen on the right plot. It implies that $\mathrm{ROC}_{H_s^{(1)},G_s^{(0)}}$ and $\mathrm{ROC}_{H_s^{(0)},G_s^{(1)}}$ are much closer for simultaneously small TPR's and FPR's, which implies that thresholding top scores will yield fair classifiers in FPR and TPR again for a whole range of high thresholds.

Results for the dataset *German*. Recall that for this credit scoring dataset we consider the AUC-based constraint in Eq. (10.5) to force the probabilities that a positive from a given group is mistakenly ranked higher than a negative to be the same across groups. Despite the blatant issues of generalization due to the very small size of the dataset (see Table 10.3), we see in Fig. 10.8 that the learned score without fairness constraints systematically makes more errors for women with good ground truth credit risk, as can be seen from comparing $\mathrm{ROC}_{H_s^{(1)},G_s^{(0)}}$ and $\mathrm{ROC}_{H_s^{(0)},G_s^{(1)}}$. Additionally, the credit score of men with good or bad credit risk is in both cases stochastically larger than that of women of the same credit risk assessment (see $\mathrm{ROC}_{G_s^{(0)},G_s^{(1)}}$ and $\mathrm{ROC}_{H_s^{(0)},H_s^{(1)}}$). On the other hand, the score learned with an AUC constraint makes a similar amount of mistakes for both genders, with only slightly more mistakes made on men than women, and the scores $s(X)$ conditioned on the events $(Y = y, Z = z)$ with $z = 0$ and $z = 1$ are more aligned when considering both $y = -1$ and $y = +1$. Finally, while the score learned with a ROC constraint has a slightly higher discrepancy between the AUC's involved in Eq. (10.5) than the one learned with an AUC constraint, one observes that both pairs of distributions $(G_s^{(0)}, G_s^{(1)})$ and $(H_s^{(0)}, H_s^{(1)})$ are equal for high thresholds. Consistently with the results on other datasets, this suggests that our score leads to classifiers that are fair in FPR and TPR for a whole range of problems where one selects individuals with very good credit risks by thresholding top scores.

10.6 Conclusion

In this chapter, we considered the issue of fairness for scoring functions learned from binary labeled data. We proposed general notions of fairness based on the AUC criterion and the ROC curves, and provided statistical guarantees for scoring functions learned via empirical AUC maximization under such fairness constraints. From a practical perspective, we showed how to implement stochastic gradient descent algorithms to solve these problems and illustrated our concepts and methods via numerical experiments. We point out that our framework can be extended to precision-recall curves (as they are a function of the FPR and TPR Clémençon and Vayatis (2009)) and to *similarity ranking*, a variant of bipartite ranking covering applications like biometric identification Vogel et al. (2018). In future work, we plan to investigate how the unrelaxed versions of our fairness constraints can be incorporated to ROC curve optimization algorithms based on recursive partitioning, such as those developed in Clémençon et al. (2011); Clémençon and Vayatis (2010).

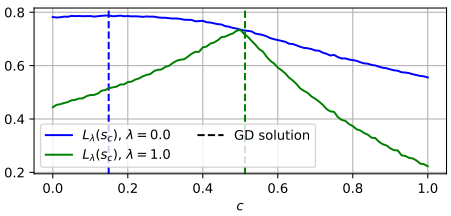
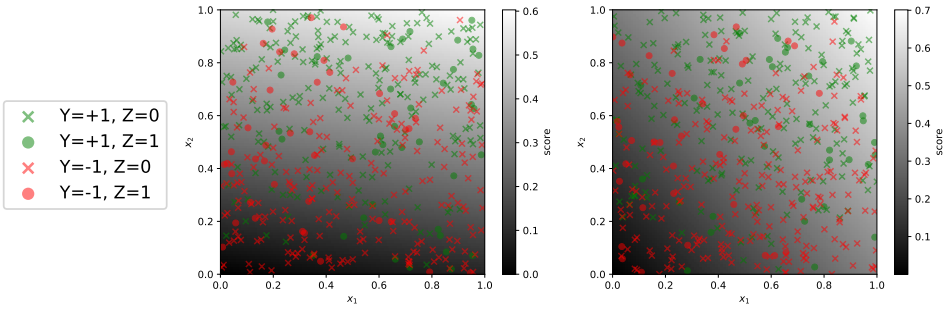
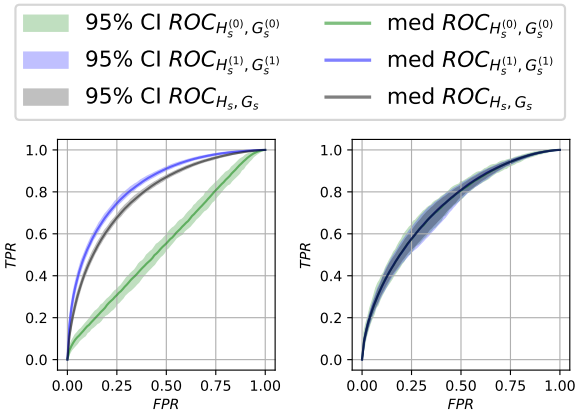


Figure 10.2: For Example 10.11, $L_\lambda(s_c)$ as a function of $c \in [0,1]$ for any $\lambda \in \{0,1\}$, with the parametrization $s_c(x) = cx_1 + (1-c)x_2$, and the values c for the scores obtained by gradient descent with Algorithm 2.



(a) $\lambda = 0$ (b) $\lambda = 1$

Figure 10.3: Values of the output scoring functions on $[0,1]^2$ for Algorithm 2 ran on Example 10.11.



(a) $\lambda = 0$ (b) $\lambda = 1$

Figure 10.4: Result of Example 10.11 with Algorithm 2.

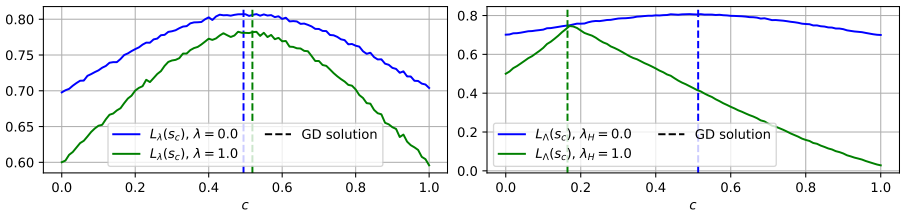
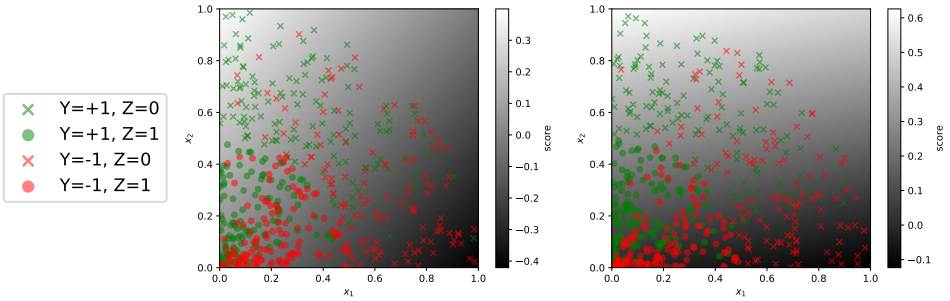
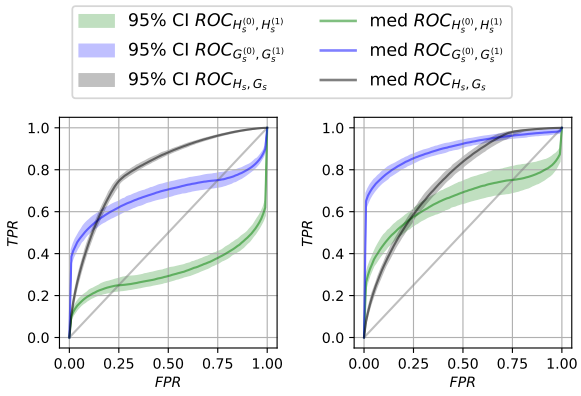


Figure 10.5: On the left (resp. right), for Example 10.12, $L_\lambda(s_c)$ (resp. $L_\Lambda(s_c)$) as a function of $c \in [0,1]$ for any $\lambda \in \{0,1\}$ (resp. $\lambda_H \in \{0,1\}$), with the parametrization $s_c(x) = -cx_1 + (1-c)x_2$, and the values c for the scores obtained by gradient descent with Algorithm 2 (resp. Algorithm 3).



(a) $\lambda_H = 0$ (b) $\lambda_H = 1$

Figure 10.6: Values of the output scoring functions on $[0,1]^2$ for Algorithm 3 ran on Example 10.12.



(a) $\lambda_H = 0$ (b) $\lambda_H = 1$

Figure 10.7: Result of Example 10.12 with Algorithm 3.

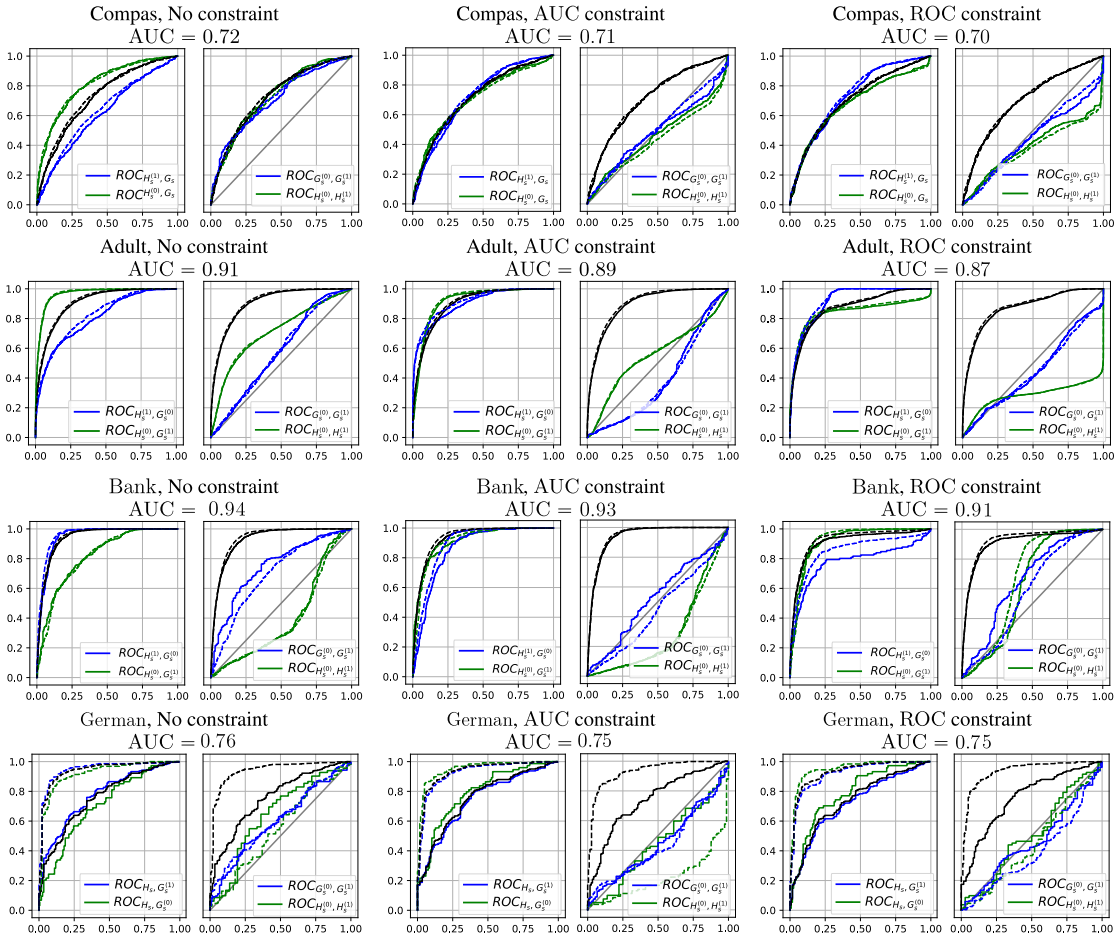


Figure 10.8: ROC curves for the databases Adult, Compas, Bank and German for a score learned without and with fairness constraints. On all plots, dashed and solid lines represent respectively training and test sets. Black curves represent ROC_{H_s,G_s}, and above the curves we report the corresponding ranking performance AUC_{H_s,G_s}.

Chapter 11

Conclusion and Perspectives

This thesis proposed to study important problems in biometrics from the point of view of statistical learning theory. Our work provides new theoretical results, which suggest original approaches to these problems, and also provide security guarantees in the form of generalization results. In that regard, it is a much-needed answer to the rapidly increasing volume of experimental machine learning literature that biometrics researchers have to follow. Biometric identification, and facial recognition in particular, incarnate many machine learning topics simultaneously, such as pairwise learning, sample bias or ranking. For this reason, we considered stylized versions of those problems, as their simultaneous examination would obscure our discourse, and runs the risk of being disregarded as anecdotal by the machine learning community. The richness of the topics tackled by the thesis is a result of this imperative.

Similarity Ranking Theory (Chapter 5 - Part II). First, we proposed to view similarity learning from the perspective of bipartite ranking on a product space, a problem that we named similarity ranking. From this point of view, we proposed another approach to similarity learning that is better suited to the evaluation of biometric systems. In a context of rapid development of the deep metric learning literature (Wang and Deng, 2018), this is a much-needed contribution. We proposed generalization guarantees that extend known results for bipartite ranking. Our analysis immediately suggest other approaches to similarity ranking, for example with extensions of ranking the best criteria in bipartite ranking (Menon and Williamson, 2016, Section 9). Our empirical illustration of the fast generalization rates for the pointwise ROC optimization problem gives an intuitive interpretation of these rates. Since generalization bounds are formulated as orders of n, they sometimes appear impractical. In that regard, future works could build upon our illustration to convey the meaning of fast learning rates. Finally, our extension of the TreeRank algorithm proposes a first practical approach to similarity ranking, with strong theoretical guarantees. As far as we know, very few papers propose learning a piecewise constant similarity. Future work could derive new methods to learn those and study their practical performance, for example by building on top of the work of Clémençon and Vayatis (2010).

Distributed U-Statistics (Chapter 6 - Part II). While the similarity ranking approach seems natural, it comes with specific challenges, such as the usual computational complexity of pairwise learning. In that regard, we exploited recent results on incomplete U-statistics (Clémençon et al., 2016), to alleviate this problem. Additionally, we derived new estimators for estimating U-statistics in a distributed environment, and compared their variance. Our statistical analysis answers concerns regarding the lack of study of distributed computing frameworks from the perspective of statistical accuracy, deplored for example in Jordan (2013). Finally, we proposed a gradient-descent learning approach with our distributed estimators, which is shown to provide trade-offs between communication costs and expected performance of the final model. While our analyses are well suited to estimation, extending them for our distributed gradient-descent algorithm is an open question. This extension is also a technical challenge, since the optimized functional changes during the training process, which makes the analysis more complicated than that of Papa et al. (2015). Performing the analysis would provide interesting results, as it applies to the optimization of any pairwise loss optimized with batch gradient descent, a problem that arises in many practical settings.

Practical Similarity Ranking (Chapter 7 - Part II). Generalization guarantees justify con-

sidering similarity ranking, while sampling techniques alleviate the computational costs associated with similarity ranking. However, both contributions do not address the optimization challenges that arise in that setting. From a biometrics perspective, the most important perspective in this thesis is to realize the potential impact of the methods presented, by providing strong empirical evidence of their relevance in practical settings. Indeed, while the rapid adoption of machine learning techniques by private companies has boosted the growth of the field, it also directed most of the attention to papers that propose unequivocal solutions to specific industrial problems. A notorious example for face recognition is Schroff et al. (2015). In this direction, we proposed gradient descent-based approaches that solve the pointwise ROC optimization problem, as well as illustrations of the TreeRank algorithm for learning similarities, that we all ran on simple toy examples. In that context, the promotion of this work will require finding and presenting pedagogically large-scale experiments that address precisely practical use-cases, which is a promising direction for future work.

Ranking the Most Likely Labels (Chapter 8 - Part III). Similarity ranking is the theoretical formalization of the flagship problem in biometrics, *i.e.* 1:1 identification. However, it does not covers all of the realities addressed by biometric systems providers. For example, to identify potential suspects in forensics, the objective is usually to return the most likely labels from a database. More generally, hard classification problems — *e.g.* Russakovsky et al. (2014) — consider other objectives than simple (top-1) classification to evaluate the performance of the system. Notably, a popular alternative performance measure is top-5 classification. In that context, the optimization customarily optimizes the query system for returning solely the best element, since orderings over labels are derived from class probabilities learned by optimizing for classification. The discrepancy between the risk functional and the evaluated objective suggests that that approach is ill-suited. Our work proposes the first theoretically-guaranteed approach to the prediction of a list of most likely results from classification data, by exploiting recent results in ranking median regression (Clémençon et al., 2018). It relies on the famous One-Versus-One (OVO) strategy for multiclass classification, and a byproduct of our analysis is the first generalization guarantees for OVO. From the point of view of biometrics, we provide statistical guarantees for criteria that are tailored to its specific objectives. Another common problem in the field is the absence of a significant difference in matching score between positive and negative instances, as ROC curves only evaluate order relations between the score of observations. In that regard, future work could consider criteria that explicitly force the score function to discriminate significantly between the two types of instances, *i.e.* with a large gap in score value. Beyond that example, other settings could be addressed, for example by discussing the 1:N identification problem described in Jain et al. (2011).

Selection Bias Correction (Chapter 9 - Part III). Another topic of interest in biometrics is the representativeness of training databases with respect to the operating conditions of the systems. Grother and Ngan (2019) measured practically the effects of biased databases in the context of facial recognition. That report has shown that the predominance of white caucasian people in training datasets makes the end algorithms more accurate on this specific type of population. To correct for representativeness issues, we proposed a reweighting scheme inspired from the idea of importance sampling (Bucklew, 2010, chapter 4), that adjusts for differences between the training distribution and the testing distribution. Our reweighting scheme is practical, in all scenarios where additional information is available on the relationship between the training and testing data. Precisely, our work assumes the knowledge of characteristics of the test distribution that can be summarized by a few finite values, such as class probabilities or the probabilities of specific strata in the population. While additional information is available in many practical scenarios, *e.g.* for a facial recognition system deployed in an airport that contains known proportions of several nationalities, a few other approches were proposed to relate the training and testing distributions. For example, Sugiyama et al. (2007) proposes to estimate a likelihood ratio with a small sample of the test set as auxiliary information. Additionally, while the absolute continuity assumption is necessary for importance sampling, Laforgue and Clémençon (2019) loosened that assumption by considering several training samples. Therefore, a promising direction for future work is the exploration of new ways to incorporate other types of auxiliary information during training, and relate those to our contribution in a general framework.

Learning Fair Scoring Functions (Chapter 10 - Part III). Although accounting for the representativeness of the databases may correct some of the flaws of a model, the distribution of

the training data often contains other type of biases that should be explicitly addressed. On that subject, the fairness in machine learning literature proposes approaches to balance the expected responses of a system between two sensitive groups, *e.g.* men and women. Our work proposed to learn scoring functions that are fair with respect to criteria based on usual ranking measures, such as the AUC-based ones of Borkan et al. (2019), as well as a framework to unify all of the AUC-based fairness criteria. Additionally, our work discussed the limitations of AUC criteria. In reaction to those limitations, we introduce new fairness criteria, based directly on the ROC curve. The usual approach when dealing with fairness constraints is to relax them, and then integrate them with the performance indicator of the task as a penalty term. As we followed this rationale, we could not explicit a notion of optimal fair score function. In the context of fair regression, Chzhen et al. (2020) managed to derive an expression of the optimal fair regressor. In bipartite ranking, overcoming this hurdle paves the way for an extension of the partition-based algorithms algorithms of Clémençon and Vayatis (2009) under fairness constraints. Another possibility concerns the extension of the techniques presented here to the case of similarity ranking. Indeed, that extension gives a framework that matches operational considerations in biometrics very closely, and would be justified by the current interest in methods to explicitly correct biases for facial recognition. The experimental component of that work would be supported by the availability of well-suited face databases (Wang et al., 2019).

Conclusion. In general, each topic discussed in the thesis could be further developed in several natural ways, as presented in the manuscript. They could also be exploited to form at-scale experiments that showcase the effectiveness of the suggested approaches, whereby bolstering their adoption. Another possibility consists in using simultaneously the strategies of several topics, while ensuring that their overlap covers an application of significant importance in the machine learning community. Both of these last points are gateways to bringing more visibility to this work. In conclusion, the richness of the issues that arise in biometrics makes for fertile grounds for new theory and new practices in machine learning. This richness spurred the creation of this thesis and can inspire future research.

Chapter 12

Résumé (Summary in French)

12.1 Contexte

La thèse est issue d'une collaboration entre la grande école Télécom Paris et la société IDEMIA. Le projet s'appuie sur un contrat CIFRE (Convention Industrielle de Formation par la REcherche), un type de contrat introduit en 1981 par le gouvernement français pour renforcer les liens entre les institutions de recherche et les entreprises privées. Le travail de recherche est donc supervisé par les deux parties, ce qui est rendu possible dans notre cas par des interactions fréquentes.

Télécom Paris est l'un des principaux établissements publics français d'enseignement supérieur et de recherche en France, et est membre de l'Institut Mines-Télécom (IMT) et de l'Institut Polytechnique de Paris (IP Paris). L'IP Paris est un établissement public d'enseignement supérieur et de recherche qui regroupe cinq prestigieuses écoles d'ingénieur françaises: l'École Polytechnique, ENSTA Paris, ENSAE Paris, Télécom Paris et Télécom SudParis. Sous les auspices de l'Institut, ils partagent leur expertise pour développer des programmes de formation d'excellence et de recherche de pointe. La recherche impliquée dans cette thèse a été effectuée au sein de l'équipe Signal, Statistiques et Apprentissage (S2A) du Laboratoire de Traitement et Communication de l'Information (LTCI). L'équipe de supervision académique était composée de Stephan Clémençon et Anne Sabourin, tous deux membres de l'équipe S2A, ainsi que d'Aurélien Bellet, chercheur à l'INRIA (Institut National de Recherche en Informatique et en Automatique).

IDEMIA est une société leader en matière d'identification biométrique et de sécurité numérique. La société est une fusion des sociétés Morpho et Oberthur Technologies, effectuée en 2017. Oberthur technologies a été un acteur dominant dans les solutions de sécurité numérique pour le monde mobile, tandis que Morpho était considéré comme le leader mondial de l'identification biométrique. IDEMIA a pour objectif de faire converger les technologies développées pour le secteur public (par l'ancien Morpho) et pour le secteur privé (par Oberthur Technologies). Dans le secteur privé, les principaux clients de l'entreprise proviennent du secteur bancaire, des télécommunications et des objets connectés. La thèse a commencé en 2017 avec Safran Identité et Sécurité (anciennement Morpho) avant la fusion, lorsque Morpho était une filiale de la grande entreprise d'aéronautique et de défense Safran. Tout au long de la thèse, Stéphane Gentric a assumé la supervision continue de ce projet du côté industriel. Les responsables de l'équipe *Advanced Machine Learning* à IDEMIA: successivement Julien Bohné, Anouar Mellakh et Vincent Despiegel, ont contribué de manière significative à cette supervision.

12.2 Introduction

La *biométrie* est la discipline qui consiste à distinguer les individus sur la base de leurs attributs physiques ou comportementaux tels que les empreintes digitales, le visage, l'iris et la voix. Dans le monde moderne, la biométrie a de nombreuses applications essentielles, telles que la surveillance des frontières, le commerce électronique et le versement de prestations sociales. Bien que son usage courant soit récent, la discipline n'est pas nouvelle. En effet, à la fin du XIXe siècle, l'officier de police français Alphonse Bertillon a proposé un système d'identification personnelle basé sur

la mesure de parties osseuses du corps (Jain et al., 2011).

Aujourd'hui, la mesure biométrique la plus répandue est l'empreinte digitale, suivie par le visage et l'iris. Toutes reposent sur l'acquisition d'images de parties spécifiques du corps. Ainsi, alors que la biométrie est considérée par de nombreux auteurs comme un domaine distinct de la science, son histoire et son développement sont étroitement liés avec celui de la *vision par ordinateur*, le domaine scientifique interdisciplinaire qui vise à permettre aux ordinateurs d'acquérir une compréhension de haut niveau des images numériques.

Au début des années 2010, les performances des systèmes de vision par ordinateur ont commencé à s'améliorer (Goodfellow et al., 2016), en raison du développement des calculs génériques sur processeur graphique (GPGPU, *General-purpose Processing on Graphics Processing Units*). Ce développement a permis l'adoption généralisée des réseaux de neurones, des modèles statistiques composés de couches qui résument l'information, car leur entraînement bénéficie fortement d'une multiplication très rapide de matrices. L'entraînement des réseaux de neurones consiste à trouver les paramètres du réseau qui minimisent une fonction de perte avec des algorithmes de descente de gradient. Ceux-ci modifient itérativement les paramètres du réseau, en ajoutant une petite quantité qui est négativement proportionnelle au gradient à chaque étape. L'intérêt croissant pour les réseaux de neurones a engendré un énorme corpus de littérature scientifique. La plupart des articles proposent une meilleure architecture pour le modèle, suggèrent une amélioration de la méthode d'optimisation ou introduisent une meilleure fonction de perte pour un problème particulier.

La littérature récente a proposé de nombreuses fonctions de perte pour la biométrie, basées sur l'intuition qu'une séparation plus stricte des identités dans un espace de représentation entraîne une amélioration des performances (Wang and Deng, 2018). Le problème phare de la biométrie est la vérification 1:1, qui vise à vérifier l'identité d'une personne en comparant une mesure en direct avec des données de référence avec une fonction de similarité. Par exemple, l'entrée d'un individu dans une zone restreinte peut nécessiter la conformité de la mesure avec une carte d'identification personnelle. La performance de la vérification 1:1 est évaluée à l'aide de la courbe ROC, un critère fonctionnel qui résume la qualité d'une fonction de similarité. Pour un ensemble de tests, la courbe ROC donne tous les taux de fausses acceptances et de faux rejet pouvant être atteints en seuillant la fonction de similarité. Il s'agit de la mesure référence pour l'évaluation des fonctions de score. Dans la thèse, nous plaidons pour l'exploitation de la littérature sur l'ordonnancement/*scoring* bipartite pour concevoir des fonctions de perte pour la vérification 1:1, en la considérant comme l'évaluation d'un score sur des paires d'observations. Bien que la littérature traite indépendamment des problèmes de *scoring* et d'apprentissage sur paires, leur examen simultané est nouveau et pose des défis particuliers.

Les récentes améliorations spectaculaires de la performance pour de nombreuses applications de l'apprentissage automatique préfigurent l'émergence de nouveaux marchés, issus de la maturation de technologies autrefois très expérimentales. L'un de ces marchés est celui de la reconnaissance faciale, qui a enregistré, et devrait maintenir, une croissance exponentielle. Son développement a suscité une couverture médiatique des éventuelles utilisations abusives et biais systémiques de la technologie, qui s'ajoutent aux préoccupations usuelles en matière de protection de la vie privée. Dans ce contexte, les praticiens et les organismes gouvernementaux ont enregistré des différences de précision entre les ethnies dans la reconnaissance faciale (Grother and Ngan, 2019). Une explication courante est que les bases de données d'images de visages disponibles pour l'entraînement des systèmes de reconnaissance faciale ne sont pas représentatifs de la population générale. L'écart de performance soulève la question plus large de l'équité, une préoccupation commune dans l'automatisation de décisions, et qui a reçu une attention croissante dans la littérature récente en apprentissage automatique (Barocas et al., 2019). Certains observateurs ont même commenté que les algorithmes prédictifs risquent d'être simplement "des opinions ancrées dans les mathématiques". Selon cette idée, les praticiens ne devraient pas se concentrer uniquement sur la performance prédictive, mais aussi s'assurer de la conformité de leur système à un ensemble de valeurs morales. La biométrie est également concernée par l'équité. En effet, même lorsque la représentativité de la base de données en termes de genre est prise en compte, les femmes ont généralement un taux de faux positifs plus élevé, possiblement en raison de normes sociales concernant l'apparence. Cela peut conduire à une discrimination systémique, notamment lorsqu'on considère des systèmes qui signalent des personnes recherchées. Dans la thèse, nous proposons d'abord de corriger les biais en utilisant des poids d'importance, ce qui ajuste la

distribution des données d'entraînement. La correction couvre plusieurs cas où les données d'entraînement et de test ne correspondent pas, et suppose l'existence d'informations auxiliaires sur un lien entre ces données. Le niveau de généralité où nous nous plaçons est nouveau, et couvre de nombreuses applications importantes en biométrie. Ensuite, nous proposons de modifier les fonctions de perte afin d'intégrer explicitement des considérations d'équité lors de l'apprentissage d'un score pour l'ordonnancement bipartite. L'équité dans un contexte de classification est l'objet de beaucoup de travaux dans la littérature, mais ce n'est pas le cas pour l'équité en ordonnancement bipartite. Compte tenu de notre perspective de *scoring* sur le problème de vérification 1:1, ce travail est une étape intermédiaire dans l'intégration explicite de contraintes d'équité dans ce problème.

En général, de nombreuses questions abordées en apprentissage automatique se posent simultanément dans la conception de systèmes biométriques. L'objectif de cette thèse est d'identifier et d'en aborder plusieurs du point de vue de l'apprentissage statistique, afin de: fournir des garanties de sécurité basées sur des hypothèses probabilistes, et de proposer des solutions judicieuses aux fabricants de systèmes biométriques. À cet égard, la littérature habituelle sur l'apprentissage statistique traite de problèmes simples, tels que la classification ou la régression (Boucheron et al., 2005). Cependant, les problèmes abordés en biométrie impliquent à la fois un apprentissage sur paires et un critère fonctionnel, et nécessitent donc une analyse spécifique.

Plan du chapitre. Le chapitre présent résume les contributions des principales parties II et III de la thèse, et n'aborde pas la partie restante (Partie I). La partie I figure dans le manuscrit pour des raisons de clarté et ne contient que des préliminaires techniques aux résultats théoriques des autres parties. Le chapitre présent est organisé comme suit: premièrement, la section 12.3 couvre les idées présentées dans la brève introduction ci-dessus et donne un aperçu détaillé de la thèse. Deuxièmement, la section 12.4 se concentre sur la partie II et traite de l'ordonnancement par similarité. Troisièmement, la section 12.5 résume les contributions de la partie III sur le vaste sujet de la fiabilité des algorithmes d'apprentissage automatique. Enfin, la section 12.6 détaille les perspectives de la thèse.

La section 12.3 est une introduction développée qui présente les aspects importants de la biométrie qui sont abordés dans la littérature sur l'apprentissage automatique. Plus précisément, elle présente d'abord la relation entre la biométrie et l'apprentissage de métriques, ainsi que l'impact de l'apprentissage approfondi sur les deux domaines. Elle examine ensuite la nature des biais dans la reconnaissance faciale, et détaille les dangers potentiels des biais du point de vue de l'équité des algorithmes.

La section 12.4 est un bref résumé de la partie II. Elle se concentre sur l'idée de considérer l'apprentissage de similarité comme un problème de *scoring* sur un espace produit. Nous appelons ce point de vue *ordonnancement par similarité*. À cet égard, elle commence par des garanties théoriques pour ce problème. Ensuite, elle propose des stratégies accompagnées de garanties statistiques pour réduire la complexité de calcul en ordonnancement par similarité, et finit avec plusieurs approches pratiques basées sur le gradient.

La section 12.5 est un bref résumé de la partie III, qui aborde le sujet de la fiabilité en apprentissage automatique. À ce titre, elle propose d'abord une stratégie pour prévoir une liste ordonnée des classes probables à partir de données de classification multiclasse, au lieu de se concentrer uniquement sur la précision en classification. Ensuite, elle donne des stratégies pour faire face au manque de représentativité des bases de données. Elle se termine par une proposition visant à faire respecter des critères d'équité dans le problème de *scoring*.

La section 12.6 présente les perspectives de la thèse. Plus précisément, elle souligne l'importance d'illustrations pratiques de ce travail pour les praticiens de la biométrie, et discute plusieurs extensions possibles de nos analyses.

Les notations adoptées tout au long de la thèse sont résumées dans la table 12.1.

12.3 Défis récents en biométrie

Les progrès récents dans le domaine de l'apprentissage profond ont entraîné des changements rapides de l'état de l'art en biométrie. À cet égard, la littérature dite d'"apprentissage profond

Notation	description
w.r.t.	*with respect to*: par rapport à
s.t.	*Subject to*: tel que
r.h.s.	*Right-hand side*: partie droite
l.h.s.	*Left-hand side*: partie gauche
a.s.	*Almost surely*: presque partout
r.v.	*Random variable*: variable aléatoire
i.i.d.	*Independent and identically distributed*: indépendant et identiquement distribué
c.d.f.	*Cumulative distribution function*: fonction de répartition
p.d.f.	*Probability density function*: fonction de densité
$:=$	Définition d'une variable
$\forall$	Quantifieur universel
$\mathcal{X} \to \mathcal{Y}$	Application de $\mathcal{X}$ à $\mathcal{Y}$
$A^\top$	Transposée de la matrice A
$\varnothing$	Ensemble vide
$\binom{n}{k}$	Coefficient binomial
$\mathfrak{S}_n$	Permutation de $\{1, \dots, n\}$
$A \cup B$ (resp. $A \cap B$)	Union (resp. intersection) entre les ensembles A et B
$A \Delta B$	Différence symétrique entre les ensembles A et B
$\#A$	Cardinal de l'ensemble A
A^c	Complémentaire d'un ensemble A
$\mathcal{P}(A)$	Ensemble de toutes les parties d'un ensemble A
$\subset$	Inclusion d'ensembles
$\mathbb{I}\{\cdot\}$	Fonction indicatrice
$\mathrm{Im}(f)$	Image de la fonction f
$\mathrm{sgn}(\cdot)$	Fonction de signe, $\mathrm{sgn}(x) = 2\mathbb{I}\{x \geqslant 0\} - 1$
$\log(\cdot)$	Logarithme naturel
$O(\cdot)$	"Grand O": Ordre asymptotique d'une quantité
$\mathbb{P}[\cdot]$	Probabilité d'un événement
$\mathbb{E}[\cdot]$	Espérance d'une variable aléatoire
$\mathrm{supp}(\mu)$	Support de la distribution μ
$X \sim \mu$	La *r.v.* X suit la distribution μ
$\mu \otimes \nu$	Mesure produit entre μ et ν
δ_x	Masse de Dirac au point x
$\bar{F}$	Fonction de survie pour la *c.d.f.* F, $\bar{F} = 1 - F$
F^{-1}	Inverse généralisé d'une fonction càdlàg
$\mathbb{N}$ (resp. $\mathbb{R}$)	Nombres naturels (resp. rééls)
$\mathbb{N}^* = \mathbb{N}\{0\}$,	$\mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geqslant 0\}$, $\mathbb{R}_+^* = \mathbb{R}_+ \backslash \{0\}$

Table 12.1: Résumé des notations.

de métriques“ a proposé de nombreuses pertes dérivables, mais aucune d'entre elles n'est liée à l'évaluation en termes d'ordonnancement des systèmes biométriques. Simultanément, ces avancées ont suscité un débat public sur les technologies biométriques, et en particulier sur la reconnaissance faciale. Si la plupart des questions concernent l'utilisation des algorithmes de reconnaissance faciale, l'une d'entre elles concerne leur biais racial, mesuré récemment. La communauté de recherche en apprentissage automatique peut proposer des solutions techniques pour ce problème.

12.3.1 Introduction à la biométrie

Valeur sociale. La biométrie répond à la nécessité d'établir l'identité d'une personne avec une grande confiance. Elle est devenue cruciale dans le monde moderne, car nous interagissons avec un nombre toujours croissant de personnes. Toutefois, on peut retracer ses origines à la fin du XIXe siècle, avec les premiers enregistrements de l'utilisation d'empreintes digitales pour des fins d'identification (Jain et al., 2011, Section 1.8). Aujourd'hui, la biométrie est largement utilisée dans la police scientifique et dans d'autres applications gouvernementales, ainsi que par diverses industries telles que le secteur bancaire. Un exemple d'application de la biométrie à grande échelle est le projet Aadhaar, géré par l'*Unique IDentification Authority of India* (UIDAI), qui a assigné des numéros d'identification nationaux et a enregistré les biométries (iris, visages et empreintes digitales) de plus d'un milliard de personnes (Jain et al., 2011, Section 1.6).

Formalisation des objectifs. L'objectif des systèmes biométriques est de comparer deux mesures (x, x') dans un espace d'entrée $\mathcal{X}$, par exemple deux empreintes digitales ou deux visages, et de décider si les deux proviennent du même individu. Cela se fait généralement au moyen d'une fonction de similarité sur la paire $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$, qui quantifie la probabilité que x et x' proviennent de la même personne. La décision est prise en fixant un seuil de similarité, ce qui signifie que l'on considère que la paire (x, x') est acceptée si $s(x, x') > t$, où t est un seuil dans $\mathbb{R}_+$. Il existe deux problèmes phares en matière de biométrie: la vérification et l'identification (Jain et al., 2011, Section 1.3). Le problème de vérification est également appelé authentification 1:1. Il est illustré par le cas d'utilisation du franchissement des frontières, où un officiel compare un document x' avec une mesure en direct x. Ainsi, il consiste à prendre une décision sur une paire x, x'. L'identification est illustrée par le cas d'utilisation de la surveillance automatique, où une mesure en direct x est comparée à une base de données. Précisément, il s'agit de trouver l'existence d'un élément correspondant à x dans une base de données de $N \in \mathbb{N}$ observations $\mathcal{D}_N = \{x_i\}_{i=1}^N \subset \mathcal{X}$. Si une telle correspondance existe, l'identification doit renvoyer les éléments pertinents dans $\mathcal{D}_N$. L'identification est également appelée identification 1:N ou authentification 1:N. Le nombre de personnes enrôlées N peut être important, et peut par exemple se compter en millions.

Étapes opérationnelles. Pour comparer une observation x avec une grande base de données $\mathcal{D}_N$, il est nécessaire de comparer rapidement des représentations d'éléments dans $\mathcal{X}$ ayant des besoins faibles en termes de mémoire. Cela nécessite la dérivation de représentations intermédiaires efficaces des données d'entrée. Les systèmes biométriques peuvent généralement être divisés en trois processus distincts: 1) l'acquisition de données d'entrée, appelée *enrôlement*, 2) l'extraction de caractéristiques, parfois appelée *codage* des données, 3) et la comparaison des codages. Voir (Jain et al., 2011, Section 1.2) pour plus de détails. Dans le contexte de la reconnaissance des empreintes digitales, la phase d'enrôlement couvre l'acquisition de la donnée brute, des étapes de post-traitement, ainsi que des vérifications de qualité sur l'image finale. L'extraction de caractéristiques consiste à appliquer des techniques de vision par ordinateur, suivie de techniques spécifiques aux empreintes pour extraire des points précis dans le l'image des empreintes digitales. Par exemple, des *filtres de Gabor* (Szeliski, 2011, Section 3.4.1) sont utilisés pour obtenir des *ridge orientations maps* (cartes d'orientation des crêtes) (Jain et al., 2011, Chapter 2) à partir des images brutes d'empreintes digitales. Des points caractéristiques appelés *minuties* sont alors extraits de cette représentation intermédiaire. Enfin, la comparaison repose sur l'évaluation d'une distance entre les nuages de points associés aux deux images. Nous renvoyons à Jain et al. (2011) (Section 2) pour plus de détails.

Extraction de caractéristiques. Le module qui a reçu le plus d'attention dans la recherche en biométrie est le module d'extraction de caractéristiques. Par exemple, les recherches sur la recon-

naissance automatique des empreintes digitales ont consacré une quantité phénoménale de travail pour trouver le plus d'information discriminante possible dans les images. En reconnaissance faciale, l'extraction de caractéristiques a d'abord été basée sur les *eigenFaces* (Turk and Pentland, 1991), une application de l'Analyse en Composantes Principales (ACP) aux images de visage. Ensuite, cette approche a été remplacée par la combinaison de descripteurs usuels en vision par ordinateur — tels que les motifs binaires locaux (*local binary patterns*, LBP) et les descripteurs d'images SIFT (*Scale-Invariant Feature Transformer:* transformation de caractéristiques visuelles invariantes à l'échelle) — avec des techniques de réduction de la dimensionnalité. Enfin, l'extraction de caractéristiques s'appuie désormais sur une approche de bout en bout — qui effectuent la tâche finale en utilisant les données brutes — basée sur des réseaux neuronaux convolutifs profonds, un type de réseau de neurones adapté aux images (Wang and Deng, 2018).

Des algorithmes d'apprentissage de métriques servent à entraîner le module d'extraction de caractéristiques. Récemment, l'avènement des algorithmes d'apprentissage profond a poussé les chercheurs en biométrie à remplacer la combinaison de méthodes d'extraction de caractéristiques spécifiques à la tâche et de méthodes d'apprentissage de métriques linéaires, par un apprentissage profond de métrique de bout en bout. Les chercheurs en biométrie doivent donc suivre de près les récents développements en apprentissage profond pour rester compétitifs.

12.3.2 Apprentissage profond de métriques pour la biométrie

L'apprentissage de métriques ou l'apprentissage de similarités est un problème d'apprentissage automatique dont le but est d'apprendre à quel point deux objets sont similaires. Nous renvoyons à Bellet et al. (2015a) pour une revue. En biométrie, la supervision de ces algorithmes provient d'une base de données de n images $\{x_i\}_{i=1}^n \subset \mathcal{X}$, chaque image x_i ayant une identité $y_i \in \{1, \dots, K\}$ avec $K \leqslant n$ et $(n, K) \in \mathbb{N}^2$.

Apprentissage de métriques linéaires. Les premiers algorithmes d'apprentissage de métriques et une grande partie de la littérature se concentre sur l'apprentissage de métriques linéaires. L'expression se réfère aux distances ou aux fonctions de similarité $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ qui sont des fonctions linéaires de leurs entrées. Celles-ci reposent principalement sur l'utilisation de *distances de Mahalanobis* — une distance $\mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$ pour $d \in \mathbb{N}$, paramétrée par une matrice semidéfinie positive $M \in \mathbb{R}^{d \times d}$ — avec quelques exceptions reposant sur d'autres combinaisons de M et de l'entrée (x, x'). La distance de Mahalanobis d_M entre les points x et x' s'écrit:

$$d_M(x, x') = \sqrt{(x - x')^\top M (x - x')}.$$

La factorisation de Cholesky (Petersen and Pedersen, 2008, Section 5.5.1) implique que l'on peut écrire $M = LL^\top$, où L est une matrice triangulaire inférieure. Ce résultat justifie le fait de considérer les distances de Mahalanobis comme le calcul d'une simple distance euclidienne sur une transformation des entrées, puisque $d_M(x, x') = \|Lx - Lx'\|_2$, où $\|\cdot\|_2$ est la distance euclidienne standard. Parmi les approches notables pour l'apprentissage des distances de Mahalanobis, on peut citer l'algorithme *Mahalanobis Metric for Clustering* (MMC, métrique de Mahalanobis pour le partitionnement) en 2002 (Xing et al., 2002), *Neighborhood Component Analysis* (NCA, analyse des composantes du voisinage) en 2004 (Goldberger et al., 2004), et *Large Margin Nearest Neighbor* (LMNN, grande marge pour les plus proches voisins) en 2009 (Weinberger and Saul, 2009). Plusieurs auteurs ont envisagé diverses extensions des algorithmes d'apprentissage de métriques linéaires. Par exemple, la kernelisation des méthodes d'apprentissage de la métrique linéaire a été proposée, ainsi que l'utilisation de plusieurs métriques linéaires locales, voir Bellet et al. (2015a) (Section 5). Ces extensions se sont avérées utiles pour les praticiens de la reconnaissance faciale. Par exemple, Bohné et al. (2014) considère l'application de l'algorithme MM-LMNN (Weinberger and Saul, 2009), qui apprend une métrique linéaire locale, à la reconnaissance faciale.

Apprentissage profond de métriques. En raison du développement du calcul générique sur processeur graphique (GPGPU), l'apprentissage et l'utilisation de réseaux neuronaux très profonds sont devenus raisonnables. Ces réseaux ont eu pour effet d'améliorer considérablement les performances en vision par ordinateur, notamment pour la tâche de classification à grande échelle sur le défi ILSVRC (*ImageNet Large Scale Visual Recognition Challenge*, challenge à grande échelle de reconnaissance visuelle sur ImageNet). L'avancée la plus marquante sur ce challenge a eu lieu en 2012 et est présentée dans Krizhevsky et al. (2012). Les résultats obtenus

avec les réseaux profonds ont conduit à des avancées importantes en reconnaissance faciale, avec notamment Taigman et al. (2014) en 2014. En apprentissage profond pour la reconnaissance faciale, les données brutes x sont encodées dans un vecteur $e(x)$, où $e : \mathcal{X} \to \mathbb{R}^d$ est une fonction non linéaire qui correspond à la sortie d'un réseau de neurones. Ensuite, une simple distance est calculée entre $e(x)$ et $e(x')$ pour décider si la correspondance entre x et x' est acceptée. L'encodage e est optimisé par une descente de gradient qui minimise une fonction de perte. Contrairement à d'autres biométries populaires telles que les empreintes digitales et l'iris, trouver manuellement les traits distinctifs importants d'un visage est difficile. Par conséquent, une approche de bout en bout basée sur l'apprentissage par descente de gradient est très bien adaptée à la reconnaissance faciale. Néanmoins, certains auteurs ont déjà proposé des approches basées sur l'apprentissage profond de métriques pour d'autres biométries (Minaee et al., 2019).

Fonctions de perte pour l'apprentissage profond de métriques. L'apprentissage profond de métriques, avec ses modèles de bout en bout, a remplacé le processus composé séquentiellement: d'une sélection par le praticien de caractéristiques importantes suivie d'une étape de réduction de la dimensionnalité, comme l'illustre le très influent article de Schroff et al. (2015). Pour cette raison, de nombreux auteurs se concentrent sur la recherche de meilleures architectures de réseaux de neurones. Simultanément, l'avènement de l'apprentissage par descente de gradient a ouvert la voie à de nombreux articles les fonctions de perte. Les premiers systèmes de reconnaissance faciale utilisaient l'habituelle perte d'entropie croisée softmax (SCE, *softmax cross-entropy*), une perte de classification usuelle dans l'apprentissage profond qui cherche à séparer les identités (Goodfellow et al., 2016, Section 3.13). Depuis lors, de nombreuses autres fonctions de perte ont été proposées, telles que ArcFace (Deng et al., 2019). Nous renvoyons à Wang and Deng (2018) (Figure 5) pour un aperçu des pertes pour la reconnaissance faciale. Leur objectif est soit: d'augmenter la marge entre les identités pour diminuer la variance inter-classe, regrouper toutes les observations de chaque identité pour diminuer la variance intra-classe, ou combiner les deux approches comme le fait la perte par triplets (*triplet loss*) (Schroff et al., 2015). Les praticiens ont observé que le fait d'additionner différentes pertes, tout en ajustant la proportion de chaque perte était nécessaire pour optimiser les performances (Parkhi et al., 2015).

Évaluation basée sur l'ordonnancement. La performance des systèmes de reconnaissance faciale est mesurée sur la courbe ROC, comme le montre les évaluations de systèmes commerciaux de reconnaissance faciale par Grother and Ngan (2019). Ces évaluations ont été menées par le *National Institute of Standards and Technology* (NIST, Institut national des normes et de la technologie) une agence du département du Commerce des États-Unis. La courbe ROC est le standard dans l'évaluation des fonctions de score en ordonnancement bipartite, un problème qui vise à attribuer des scores plus élevés aux éléments associés à un label positif $+1$ qu'aux éléments avec un label négatif -1. Nous renvoyons à Menon and Williamson (2016) pour une revue sur l'ordonnancement bipartite. Dans ce contexte, les fonctions de similarité peuvent être considérées comme des fonctions de score sur un espace produit. Cette observation suggère que l'utilisation du vaste corpus de recherche sur l'ordonnancement bipartite est une approche justifiée pour proposer de meilleures fonctions de perte pour la reconnaissance faciale, ce que nous faisons à la fois dans la partie II, et dans le chapitre 8 contenu dans la partie III.

Récemment, des améliorations rapides en termes de précision ont été observées en reconnaissance faciale, qui ne requiert pas que l'individu soit coopératif. Ainsi, la technologie a récemment attiré l'attention des médias et de l'opinion publique. En plus des préoccupations usuelles en matière de protection de la vie privée, les observateurs ont exprimé des inquiétudes croissantes concernant le manque de fiabilité ou l'injustice éventuelle induite par la reconnaissance faciale.

12.3.3 Fiabilité en biométrie

Les récents progrès en matière de reconnaissance faciale ont confirmé la maturation de la technologie, ce qui préfigure son déploiement et a provoqué un large débat à ce sujet. Gates (2011) a mis en garde contre la tendance du public à extrapoler sur l'omniprésence de ces systèmes, ce qui crée une l'illusion d'une surveillance qui modifie les comportements. Cependant, les obstacles techniques énoncés par Gates (2011) semblent beaucoup plus faibles aujourd'hui. Précisément, des travaux tels que Schroff et al. (2015) montrent la capacité des systèmes de reconnaissance faciale dans des conditions d'acquisition très peu contrôlées. De plus, certains observateurs ont

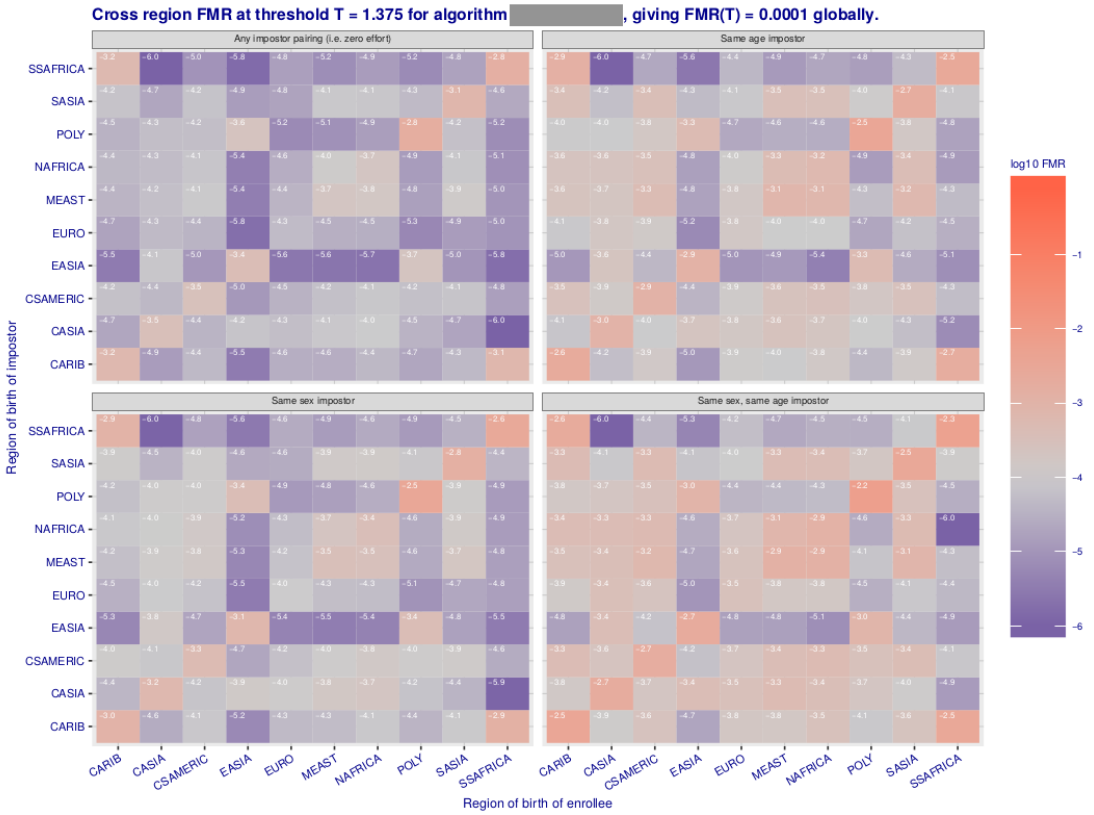


Figure 12.1: Ce graphique montre que la fixation d'un seuil pour les faux positifs (appelé taux de fausse acceptance (FAR) en reconnaissance faciale) de $\alpha = 10^{-5}$ pour la population générale peut donner un taux de faux positif bien plus élevé lorsqu'il est utilisé sur un autre type de population. Plus précisément, il donne $\alpha = 10^{-2,8}$ pour une population originaire de Polynésie.

prévu un taux de croissance annuel composé (TCAC) — c'est-à-dire une progression géométrique moyenne — de 14,5 % par an entre 2020 et 2027 pour le marché mondial de la reconnaissance faciale. Dans ce contexte, les questions relatives au déploiement de la reconnaissance faciale et des autres technologies d'apprentissage automatique appartiennent au domaine du législateur, mais les décisions des modèles sont de la responsabilité du praticien en apprentissage automatique.

Biais dans la reconnaissance faciale. Dans le cas spécifique de la reconnaissance faciale, le NIST a quantifié avec précision une moindre performance pour la reconnaissance de personnes de couleur, avec les caucasiens comme classe de référence. La figure 12.1 illustre leurs travaux et provient de Grother and Ngan (2019). Cette observation a été reprise par de nombreux médias et a été qualifiée de "biais racial" en 2019. La principale justification pour cet écart fut que les bases de données utilisées pour l'entraînement de ces systèmes sont généralement composés de personnalités européennes ou d'Amérique du Nord, qui sont pour la plupart des caucasiens et ne représentent pas la population générale. Les auteurs ont interprété cette observation comme un "*other race effect*" (littéralement, effet d'autre ethnicité) pour la reconnaissance faciale automatique, une idée introduite dans Furl et al. (2002) et Phillips et al. (2011), qui dit que les humains ont généralement du mal à distinguer les individus d'une ethnie différente de la leur. Certains auteurs ont proposé des stratégies pour corriger ce problème explicitement (Wang et al., 2019). Plus généralement, une large littérature sur l'équité en apprentissage automatique peut être invoquée pour aborder ce problème (Bolukbasi et al., 2016; Zhao et al., 2017; Hendricks et al., 2018; Liu et al., 2016; Huang et al., 2006). Le chapitre 9 dans la partie III contribue à cet effort, en proposant une méthode générale de repondération qui s'applique aux problèmes de représentativité en biométrie.

Limites de la représentativité des bases de données. Bien qu'il soit important de disposer d'une base de données représentative de la population cible, on ne peut pas s'attendre à ce que cela corrige les biais inhérents aux données d'entraînement. Précisément, même si un groupe

social identifié par des attributs protégés tels que la race ou la religion est en moyenne beaucoup plus pauvre qu'un autre groupe social, il peut être considéré comme immoral de refuser un prêt au motif qu'un demandeur appartient au premier groupe. Dans ce contexte, certains observateurs ont brutalement qualifié les algorithmes prédictifs d'"opinions intégrées dans des mathématiques" (O'Neil, 2016).

Equité algorithmique. Un grand nombre de travaux (Agarwal et al., 2018; Woodworth et al., 2017; Zafar et al., 2017a,b, 2019; Menon and Williamson, 2018; Bechavod and Ligett, 2017) sont apparus sur le thème de l'équité en l'apprentissage automatique — également appelée équité algorithmique — qui est un nouveau domaine d'étude. Ceux-ci cherchent à ajouter des contraintes explicites durant la phase d'entraînement afin que des optimisations brutales de la précision ne conduisent pas à la reproduction systémique d'injustices sociales. Les premiers travaux influents remontent à 2012 (Dwork et al., 2012). Plus récemment, des auteurs ont travaillé sur un manuel consacré à ce sujet (Barocas et al., 2019).

Equité en reconnaissance faciale. Dans la reconnaissance faciale, l'incorporation de contraintes d'équité lors de l'apprentissage des modèles peut corriger le fait que certains groupes sociaux sont plus difficiles à identifier que d'autres. C'est une nécessité dans de nombreux cas pratiques. Par exemple, si un système conçu pour signaler des personnes d'intérêt a un taux plus élevé de fausse acceptance pour une ethnicité spécifique, cela peut être interprété comme du profilage racial automatique.

Littérature scientifique sur l'équité. La littérature scientifique sur l'équité pour le problème de classification est vaste, mais il existe peu de travaux pour des contextes spécifiques, tels que l'ordonnancement ou l'apprentissage de similarités. Parmi les exceptions notables, citons Beutel et al. (2019) pour l'ordonnancement, qui utilise une étape de post-traitement pour modifier une fonction de score afin de satisfaire un critère d'équité. Cela laisse entrevoir la possibilité de nouvelles approches pour l'équité spécifiquement adaptées à ces problèmes importants, ce que nous proposons dans le chapitre 10 contenu dans la partie III.

Dans son ensemble, la thèse participe au dialogue entre la communauté de l'apprentissage automatique et celle de la biométrie. Elle propose une vision stylisée et théorique de challenges en biométrie, qui s'appuie sur la littérature récente pour étudier les critères spécifiques — c'est-à-dire fonctionnels et basés sur les paires — qui sont abordés en biométrie. Nous espérons que notre point de vue sur les problèmes biométriques aura des répercussions bénéfiques sur la pratique. De plus, la dérivation de garanties statistiques pour les systèmes biométriques constitue un outil important pour s'assurer de leur sécurité.

12.3.4 Plan de la thèse

La thèse est divisée en trois parties. La partie I de la thèse contient des préliminaires techniques, qui fournissent tous les résultats intermédiaires nécessaires pour prouver les contributions théoriques de la thèse. Elle est comprise dans la thèse pour des raisons de clarté. La partie II et III se concentrent sur nos contributions. La partie II se penche sur l'idée de considérer l'apprentissage de similarité comme un problème de *scoring* sur un espace produit. La partie III est centrée autour de l'idée générale de la fiabilité des algorithmes d'apprentissage automatique.

La partie I est divisé en trois chapitres. Le premier chapitre (Chapitre 2) est une introduction rapide à la théorie de l'apprentissage statistique. Il présente les résultats nécessaires pour obtenir des garanties de généralisation dans le cas facile de la classification binaire. Il détaille précisément la dérivation de bornes à nombre de données fini sur l'excès de risque du minimiseur empirique. La plupart de nos contributions théoriques peuvent être interprétées comme des extensions de ces résultats, mais nos contributions concernent des problèmes plus complexes. Le deuxième chapitre (Chapitre 3) traite de tous les résultats requis qui concernent l'idée d'ordonnancement en apprentissage automatique. Nos contributions s'appuie sur deux thèmes liés à l'ordonnancement: l'ordonnancement bipartite et l'agrégation d'ordonnancements. En effet, notre travail étend les garanties existantes en ordonnancement bipartite à l'ordonnancement par similarité présenté dans la partie II. L'agrégation d'ordonnancement est utilisée pour le *bagging* (*bootstrap aggregating*, une méthode pour l'aggrégation de modèles statistiques) d'arbres d'ordonnancement, et les garanties du premier chapitre de la partie III sont construites sur un modèle paramétrique pour les ordonnancements, qui peut être considéré comme un outil pour

l'agrégation d'ordonnancements. Enfin, le troisième et dernier chapitre (Chapitre 4) présente une brève introduction à d'importants résultats sur les U-statistiques, une type de statistique présente dans tous les problèmes d'apprentissage par paires. À ce titre, le chapitre 4 est un pré-requis à nos garanties pour l'ordonnancement par similarité, et intervient lors de l'estimation de fonctions de risque dans le cadre de l'ordonnancement bipartite standard.

La partie II explore l'idée de considérer le problème de vérification biométrique comme l'ordonnancement de paires d'instances. Il est divisé en trois chapitres. Le premier chapitre (Chapitre 5) présente formellement l'idée de voir l'apprentissage de similarité du point de vue de l'optimisation de la courbe ROC. Il propose de nouvelles garanties pour le problème de l'optimisation de la courbe ROC en un point, qui cherche à optimiser le taux de vrais positifs avec une limite supérieure sur le taux de faux positifs. Cette analyse ouvre la voie à une extension des garanties des garanties de l'algorithme TreeRank à l'apprentissage de similarité, que nous fournissons. À l'aide de simulations numériques, nous produisons la première illustration empirique des vitesses d'apprentissage rapide, adaptées ici au cas spécifique de l'optimisation de la courbe ROC en un point pour l'ordonnancement par similarité. En raison du nombre prohibitif de paires impliquées dans les calculs, les propositions du premier chapitre ne sont pas raisonnables pour une application à grande échelle. Pour y remédier, les statisticiens ont proposé des approximations par échantillonnage pour les U-statistiques, une approche utile pour l'ordonnancement par similarité. Le deuxième chapitre (Chapitre 6) étend cette proposition aux contextes où les données sont dans un environnement distribué. Enfin, le troisième chapitre (Chapitre 7) est plutôt prospectif et propose des expériences numériques simples sur l'ordonnancement par similarité, qui abordent la question de l'optimisation pour ce problème. Étendre ces expériences fera l'objet de travaux futurs.

La partie III est la dernière partie de la thèse. Elle s'articule autour de l'idée de fiabilité en apprentissage automatique et est également divisée en trois chapitres. Le premier chapitre (Chapitre 8) donne des garanties d'apprentissage pour la prédiction d'un ordonnancement sur les classes possibles en utilisant seulement des données de classification multiclasse, ce qui repose sur l'utilisation d'une stratégie OVO (*One-Versus-One*, un-contre-un). Ce problème se pose souvent dans des problèmes bruités ou incertains, c'est-à-dire pour lesquels la classe la plus probable a de bonnes chances d'être un faux positif. Dans ce contexte, on s'intéresse alors aux classes les plus probables, comme c'est souvent le cas dans les enquêtes criminelles. Le deuxième chapitre (Chapitre 9) propose des techniques pour corriger le biais observé entre un échantillon d'entraînement et celui de test, en utilisant des informations auxiliaires concernant la différence entre les deux. Il repose sur l'application du principe bien connu d'échantillonnage préférentiel en statistique. Enfin, le troisième chapitre (Chapitre 10) propose une unification d'une classe de contraintes d'équité, ainsi qu'une nouvelle contrainte d'équité, plus restrictive et plus adaptée aux situations concrètes. Il comporte également des garanties théoriques, ainsi que des approches pratiques basées sur le gradient pour apprendre sous les deux types de contraintes.

Le dernier chapitre de la thèse (Chapitre 11) contient un résumé des contributions de la partie II et III, ainsi qu'un compte rendu détaillé des orientations les plus prometteuses pour de futurs travaux. Il se termine par une conclusion générale sur la thèse.

Les deux sections suivantes de ce chapitre — les sections 12.4 et 12.5 — résument les contributions de la thèse. Chaque section se concentre respectivement sur la première et la deuxième partie de la thèse, et est divisée en sous-sections qui résument chaque chapitre de la partie. Enfin, la section 12.6 résume les perspectives de la thèse.

12.4 Ordonnancement par similarité

L'apprentissage de similarité joue un rôle clé dans de nombreux problèmes d'apprentissage automatique comme le partitionnement, la classification ou la réduction de la dimensionnalité. Cette technique est particulièrement importante lorsqu'on considère des problèmes dits "en monde ouvert" — c'est-à-dire pour lesquels un modèle rencontre des classes qui n'étaient pas disponibles pendant l'entraînement au moment du déploiement (Chen et al., 2018) — ce qui est le cas pour toute application biométrique. Dans cette section, nous considérons l'apprentissage de métrique du point de vue du *scoring* de paires d'instances, ce qui est cohérent avec l'évaluation

de nombreux systèmes basés sur des techniques d'apprentissage de métrique.

12.4.1 Théorie de l'ordonnancement par similarité

L'ordonnancement/*scoring* bipartite considère un ensemble d'éléments associés à un label binaire, et cherche à classer ceux qui ont le label $+1$ plus haut que ceux qui ont le label -1. Pour obtenir un ordre sur un espace d'entrée $\mathcal{X}$, l'ordonnancement bipartite repose généralement sur l'apprentissage d'une fonction de score $s : \mathcal{X} \to \mathbb{R}$ (Menon and Williamson, 2016). D'autre part, l'apprentissage de métrique/similarité (Bellet et al., 2015a) est la tâche d'apprendre une similarité — ou de manière équivalente, une distance — $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ sur l'espace produit $\mathcal{X} \times \mathcal{X}$. Alors que les algorithmes d'apprentissage de métrique ont été évalués à l'origine par rapport à leur pertinence pour une tâche de partitionnement (Xing et al., 2002), les praticiens utilisent aujourd'hui des indicateurs de performance dérivés de la courbe ROC, la mesure standard pour l'évaluation des fonctions de score en ordonnancement bipartite. Par conséquent, notre travail introduit sous le nom d'*ordonnancement par similarité* l'idée d'apprendre des similarités pour un objectif d'ordonnancement.

Un critère fonctionnel: la courbe ROC**.** Dans le cadre de la classification multiclasse, nous introduisons une paire aléatoire $(X, Y) \in \mathcal{X} \times \{1, \dots, K\}$, avec $K \in \mathbb{N}$ le nombre de classes, ainsi qu'une copie indépendante (X', Y') de (X, Y). Ensuite, nous pouvons définir une variable $Z = 2 \cdot \mathbb{I}\{Y = Y'\} - 1$ égale à 1 si les deux paires appartiennent à la même classe et à -1 sinon. La courbe ROC d'une fonction de similarité est alors égale au *PP-plot* (*Probability-Probability plot*, représentation utilisée pour comparer deux distributions réelles) $t \in \mathbb{R} \mapsto (\bar{H}_s(t), \bar{G}_s(t))$, où, pour tout $t \in \mathbb{R}$:

$$\bar{H}_s(t) := \mathbb{P}\{s(X, X') > t \mid Z = -1\} \qquad \text{et} \qquad \bar{G}_s(t) := \mathbb{P}\{s(X, X') > t \mid Z = +1\}.$$

$\bar{H}_s(t)$ et $\bar{G}_s(t)$ sont respectivement le taux de faux positifs et de vrais positifs associés à la similarité s. Sous des hypothèses de continuité, la courbe ROC s'écrit comme le tracé de la fonction $\alpha \in (0, 1) \mapsto \mathrm{ROC}_s(\alpha) = \bar{G}_s(t) \circ \bar{H}_s^{-1}(\alpha)$. Certaines approches en apprentissage de similarité optimisent une version empirique de l'aire sous la courbe ROC (AUC, *Area Under the Curve*) de la fonction de similarité s (McFee and Lanckriet, 2010; Huo et al., 2018).

Optimisation en un point de la courbe ROC **(p**ROC**).** L'AUC est un résumé global de la courbe ROC qui pénalise les erreurs d'ordonnancement, quelle que soit la position des instances concernées dans la liste (Clémençon et al., 2008, Proposition B.2). D'autres critères se concentrent sur le haut de la liste (Clémençon and Vayatis, 2007; Huo et al., 2018), et leur étude est l'objet de la littérature sur l'ordonnancement des meilleures instances (*ranking the best instances*) (Menon and Williamson, 2016, Section 9). Dans notre travail, nous envisageons d'optimiser le taux de vrais positifs atteint par une similarité sous contrainte d'une limite supérieure $\alpha \in (0, 1)$ sur son taux de faux positifs. Ce problème est pertinent dans les applications biométriques, car les garanties de sécurité sont généralement spécifiées par une borne supérieure sur un taux de faux positifs acceptable pour un système. Nous appelons ce problème *pointwise* ROC *optimization* (pROC). Nous considérons des risques:

$$R^-(s) := \mathbb{E}[s(X, X') \mid Z = -1] \qquad \text{et} \qquad R^+(s) := \mathbb{E}[s(X, X') \mid Z = +1],$$

avec $\mathcal{S}$ une famille de fonctions de similarités candidates, le problème pROC s'écrit:

$$\max_{s \in \mathcal{S}} R^+(s) \qquad \text{avec} \qquad R^-(s) \leqslant \alpha. \tag{12.1}$$

Nous définissons s^* comme la solution de Eq. (12.1). Notons que Clémençon and Vayatis (2010) ont étudié l'équivalent de Eq. (12.1) dans le cas l'ordonnancement bipartite. Ce problème est analogue à la classification de Neyman-Pearson (Scott and Nowak, 2005), et ressemble beaucoup au problème d'apprentissage d'ensembles de volume minimum (Scott and Nowak, 2006). Lorsque $\mathcal{S}$ est la classe de toutes les fonctions mesurables, la solution de Eq. (12.1) s'écrit comme un ensemble de sur-niveau de la probabilité a posteriori $\eta : x, x' \mapsto \mathbb{P}\{(X, X') = (x, x') \mid Y = Y'\}$, ce qui est une conséquence du lemme fondamental de Neyman-Pearson (Lehmann and Romano, 2005, Théorème 3.2.1).

Estimateurs sur paires. L'analyse de Clémençon and Vayatis (2010) repose sur le fait que les estimateurs naturels de $R^-(s)$ et de $R^+(s)$ sont des moyennes empiriques usuelles dans le

cas de l'ordonnancement bipartite. Cependant, ce n'est pas le cas pour l'ordonnancement par similarité. Considérons un échantillon $\mathcal{D}_n = \{(X_i, Y_i)\}_{i=1}^n$ composé de n copies *i.i.d.* de la paire (X, Y), ainsi que les estimateurs naturels de $R^-(s)$ et $R^+(s)$ basés sur l'échantillon $\mathcal{D}_n$:

$$R_n^-(s) := \frac{1}{n_-} \sum_{i<j} \mathbb{I}\{Y_i \neq Y_j\} \cdot s(X_i, X_j), \tag{12.2}$$

$$R_n^+(s) := \frac{1}{n_+} \sum_{i<j} \mathbb{I}\{Y_i = Y_j\} \cdot s(X_i, X_j), \tag{12.3}$$

où $n_+ := \sum_{i<j} \mathbb{I}\{Y_i = Y_j\}$ et $n_- =: n(n-1)/2 - n_+$. Les quantités de Eq. (12.2) et Eq. (12.3) ne sont pas des sommes de variables aléatoires indépendantes, et l'analyse de Clémençon and Vayatis (2010) ne s'applique alors pas à ce cas. Cependant, il s'agit de ratios de statistiques bien connues, les U-statistiques (Lee, 1990; de la Pena and Giné, 1999).

Garanties de généralisation pour pROC**.** La version empirique de pROC (Eq. (12.1)) s'écrit:

$$\max_{s \in \mathcal{S}} \widehat{R}_n^+(s) \qquad \text{avec} \qquad R_n^-(s) \leqslant \alpha + \phi, \tag{12.4}$$

où $\phi \geqslant 0$ est un terme qui tolère les variations de $R_n^-(s)$ autour de son espérance $R^-(s)$. Nous écrivons s_n la solution de Eq. (12.4). La généralisation d'inégalités de concentration standard aux U-statistiques nous permet d'étendre directement les garanties uniformes de Clémençon and Vayatis (2010). Précisément, nous garantissons simultanément qu'avec une forte probabilité: $R^+(s^*) - R^+(s_n)$ est limité par une quantité d'ordre $n^{-1/2}$ et $R^-(s_n) \leqslant \alpha + \phi_n$ avec $\phi_n = O(n^{-1/2})$. En résumé, nous montrons que l'excès de risque est borné par la vitesse d'apprentissage usuelle en $n^{-1/2}$ en l'absence d'hypothèses sur la distribution des données.

Vitesses de généralisation rapide pour pROC**.** Dans le cas de la classification binaire et sous une hypothèse de bruit paramétrée par $a \in (0, 1)$ sur la distribution des données, Mammen and Tsybakov (1995) ont montré qu'on obtient une vitesse d'apprentissage rapide en $O(n^{-1/(2-a)})$. La vitesse rapide est la conséquence d'une limite supérieure sur la variance de l'excès de risque, elle-même dérivée de l'hypothèse du bruit. L'analyse de Clémençon and Vayatis (2010) s'est appuyée sur ces idées pour proposer une limite supérieure sur $R^+(s^*) - R^+(s_n)$ en $O(n^{-(2+a)/4})$ avec des garanties en $O(n^{-1/2})$ pour R^-, pour pROC en ordonnancement bipartite. Par rapport à la classification binaire, le problème pROC a une vitesse d'apprentissage plus faible, issue de la nature bilatérale de pROC. Notre travail étend les vitesses rapides de Clémençon and Vayatis (2010) pour l'ordonnancement bipartite au cas de l'ordonnancement par similarité. Incidemment, le résultat est vrai sous des hypothèses beaucoup plus faibles. Précisément, il repose sur la seconde décomposition de Hoeffding pour les U-statistiques (Hoeffding, 1948), qui implique que la variance de l'excès de risque consiste essentiellement en la variance de sa projection de Hájek. Comme la projection de Hájek est une transformation qui réduit la variance d'une U-statistique (van der Vaart, 2000, Section 11), des hypothèses plus faibles impliquent la limite supérieure de la variance nécessaire pour des vitesses d'apprentissage rapides. Clémençon et al. (2008) contient cette utilisation des propriétés des U-statistiques dans l'objectif de dériver des vitesses de convergence rapides, mais ne l'a jamais appliqué à des problèmes comportant une contrainte aléatoire dans le programme d'optimisation.

Illustration empirique des vitesses rapides. Notre travail contient également la première illustration expérimentale des vitesses rapides d'apprentissage, sur notre problème spécifique d'ordonnancement par similarité. Cette illustration repose sur la génération de données satisfaisant l'hypothèse de bruit pour différents paramètres de bruit $a \in (0, 1)$, et d'une comparaison de leurs vitesses d'apprentissage empiriques. Nous avons choisi la distribution des données et la famille de propositions $\mathcal{S}$ afin que la similarité optimale s^* soit connue et telle que le minimiseur empirique s_n puisse être trouvé exactement. Pour ces raisons, nous avons défini $\mathcal{S}$ comme étant un *decision stump* (arbre de décision de profondeur 1) sur une transformation fixe des données.

Limites de pROC**.** Alors que le problème de pROC fait écho à des considérations pratiques en biométrie, où les systèmes sont déployés pour fonctionner à un taux fixe de faux positifs α, sa résolution empirique est difficile en pratique. Les rares exceptions reposent sur un partitionnement fixé de l'espace d'entrée (Scott and Nowak, 2006). Dans de nombreuses situations, le taux de faux positifs α pour un système est inconnu à l'avance, et l'optimisation pour un mauvais α peut ne pas donner de résultats satisfaisants au déploiement.

TreeRank pour l'ordonnancement bipartite. L'algorithme TreeRank pour l'ordonnancement bipartite a été introduit dans Clémençon and Vayatis (2009). TreeRank apprend une fonction de score constante par morceaux s_{D_n}, construite pour fournir une estimation adaptative et linéaire par morceaux de la courbe ROC optimale. Comme le suggèrent les solutions optimales de Eq. (12.1), la courbe ROC optimale ROC* est celle de la probabilité a posteriori η. TreeRank sépare récursivement l'espace d'entrée $\mathcal{X}$ et optimise de façon gloutonne l'AUC à chaque séparation, et forme ainsi un arbre binaire de profondeur D_n basé sur des partitions imbriquées de l'espace d'entrée $\mathcal{X}$. Sous des hypothèses spécifiques, Clémençon and Vayatis (2009) ont prouvé des bornes uniformes sur la norme sup entre la courbe ROC optimale et celle de s_{D_n} lorsque $D_n \sim \sqrt{\log(n)}$, c'est-à-dire qu'avec une grande probabilité:

$$\sup_{\alpha \in [0,1]} \left| \mathrm{ROC}_{s_{D_n}}(\alpha) - \mathrm{ROC}^*(\alpha) \right| \leqslant \exp(-\lambda \sqrt{\log(n)}), \tag{12.5}$$

où λ est une constante spécifiée par l'utilisateur.

TreeRank pour l'ordonnancement par similarité. Notre travail propose une extension de l'algorithme TreeRank pour l'apprentissage de similarité, en considérant des séparations récursives de l'espace produit $\mathcal{X} \times \mathcal{X}$. Afin de s'assurer que la similarité s est symétrique, nous considérons seulement des séparations symétriques par rapport aux deux arguments de l'espace d'entrée $\mathcal{X} \subset \mathbb{R}^d$, en séparant l'espace sur la simple reparamétrisation suivante de $\mathcal{X} \times \mathcal{X}$:

$$f : (x, x') \mapsto \begin{pmatrix} |x - x'| \\ x + x. \end{pmatrix}.$$

En utilisant les mêmes extensions des inégalités de concentration classiques aux cas des U-statistiques qu'auparavant, nous avons étendu la preuve de Eq. (12.5) à l'ordonnancement par similarité. Notre analyse fournit une approche fondée théoriquement pour l'apprentissage de similarités qui s'approchent de la courbe ROC optimale en norme sup.

Nous ayons prouvé des garanties théoriques pour nos approches d'ordonnancement par similarité, mais les estimateurs des fonctions de risque nécessitent le calcul de sommes comprenant un très grand nombres de termes. Ce coût de calcul rend l'application pratique de ces approches inabordable. Par exemple, le calcul de $R_n^-(s)$ nécessite la somme de n_- termes, une quantité quadratique en n quand K est constant. Dans une application biométrique typique, le nombre d'échantillons par classe est fixe. Ainsi, la proportion de paires négatives n_- sur l'ensemble des paires n^2 est encore plus élevée que dans le cas K constant. La section suivante exploite des analyses récentes concernant l'approximation de U-statistiques afin d'atténuer ce problème.

12.4.2 U-statistiques distribuées

La plupart des applications biométriques nécessitent un apprentissage sur de très grands volumes de données. En reconnaissance faciale, la plus grande base de données mise à la disposition du public contient 8,2 millions d'images (Guo et al., 2016) et certaines bases de données privées sont bien plus grandes. Cela illustre les problèmes d'échelle présentés à la Section 12.4.1, car le nombre de paires négatives est supérieur à 50 trillions (10^{12}) pour Guo et al. (2016). Outre une limite en termes de nombre d'opérations, ces ensembles de données ne peuvent souvent pas être contenus dans la mémoire vive (RAM) d'une seule machine. Dans notre travail, nous avons proposé une approche pour l'estimation de U-statistiques dans un environnement distribué, qui concerne ces deux limites pratiques simultanément.

***U*-statistiques incomplètes.** L'idée d'alléger la complexité de calcul des statistiques U n'est pas nouvelle, étant donné que Blom (1976) proposait déjà l'utilisation un petit échantillon de B paires sélectionnées par tirage aléatoire avec remise dans l'ensemble de toutes les paires pour former des *U-statistiques incomplètes* en 1976. Clémençon et al. (2016) propose une borne supérieure, vraie avec une grande probabilité, sur l'écart entre une U-statistique incomplète U_B et la statistique complète U_n. Dans le cas d'une U-statistique de degré deux à un échantillon — c'est-à-dire une moyenne sur toutes les paires pouvant être formées avec un échantillon — la borne implique qu'un estimateur U_B utilisant seulement $B = n$ paires suffit pour retrouver la vitesse d'apprentissage usuelle en $n^{-1/2}$, plutôt que d'additionner toutes les $n(n-1)/2$ paires pour calculer U_n. Ce résultat implique que l'on peut étendre les résultats présentés ci-dessus

pour l'ordonnancement par similarité, afin de travailler avec des U-statistiques incomplètes, ce qui rend le problème raisonnable pour tout contexte où l'on a de grands volumes de données.

Environnements distribués. Dans les cas où les données ne tiennent pas sur une seule machine, les récents progrès technologiques en matière de bases de données distribuées et le calcul parallèle ont rendu accessible le déploiement d'algorithmes d'apprentissage automatique dans un environnement distribué, en grande partie grâce au développement de *frameworks* pour le calcul sur grappes de serveur, comme Apache Spark (Zaharia et al., 2010) ou Petuum (Xing et al., 2015). Ces *frameworks* ont permis d'abstraire les aspects réseau et communication lors du développement d'algorithmes distribués. Ils ont ainsi facilité le déploiement des algorithmes distribués, mais limité le type d'opérations qui peuvent être réalisées efficacement, généralement afin de respecter certaines propriétés algorithmiques. Simultanément, Jordan (2013) a exhorté les statisticiens de guider les praticiens de l'apprentissage automatique à grande échelle, par l'étude des implications de la distribution sur l'estimation, notamment pour mettre en perspective les gains en temps de calcul avec les pertes potentielles en matière de précision statistique. Nos travaux abordent cette question, en proposant plusieurs estimateurs pour U-statistiques dans un environnement distribué et en comparant leurs variances. Dans ce contexte, nous proposons des compromis entre temps de calcul et variance.

Cadre probabiliste. Soit deux échantillons indépendants et *i.i.d.* $\mathcal{D}_n = \{X_1, \dots, X_n\} \subset \mathcal{X}$ et $\mathcal{Q}_m = \{Z_1, \dots, Z_m\} \subset \mathcal{Z}$ contenant respectivement $n \in \mathbb{N}$ et $m \in \mathbb{N}$ éléments, et tels que $\mathcal{D}_n$ et $\mathcal{Q}_m$ peuvent avoir des distributions différentes. La U-statistique complète à deux échantillons de noyau $h : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ associée à ces données s'écrit:

$$U_{\mathbf{n}}(h) := \frac{1}{nm} \sum_{k=1}^{n} \sum_{l=1}^{m} h(X_k, Z_l), \tag{12.6}$$

avec $\mathbf{n} = (n, m)$. En revanche, la version incomplète de $U_n(h)$ basée sur B paires s'écrit:

$$U_B(h) := \frac{1}{B} \sum_{(k,l) \in \mathcal{D}_B} h(X_k, Z_l), \tag{12.7}$$

où $\mathcal{D}_B$ est un ensemble de B éléments sélectionnés aléatoirement dans l'ensemble de toutes les paires $\{(k, l) \mid (k, l) \in \{1, \dots, n\} \times \{1, \dots, m\}\}$. Pour les données à grande échelle, les ensembles de données complets $\mathcal{D}_n$ et $\mathcal{Q}_m$ ne peuvent pas être stockés sur une seule machine, ce qui rend le calcul direct de Eq. (12.6) et Eq. (12.7) impossible. Dans ce contexte, l'approche standard consiste à distribuer les données sur un ensemble de $N \in \mathbb{N}$ serveurs différents. Pour une moyenne standard, le simple calcul d'une moyenne des moyennes locales de chaque serveur donne le même estimateur que dans un contexte centralisé. Ce n'est pas aussi simple pour les U-statistiques, étant donné que sans communication réseau chaque serveur ne peut former des paires qu'avec l'échantillon local.

Estimateurs distribués pour les U-statistiques. Nous introduisons tout d'abord deux estimateurs simples dans un contexte distribué qui ne nécessitent pas de communication sur le réseau: la moyenne $U_{\mathbf{n},N}$ de N U-statistiques complètes sur chaque échantillon local, et la moyenne $U_{\mathbf{n},N,B}$ de N U-statistiques incomplètes formées de B paires choisies aléatoirement dans chaque échantillon local. En utilisant la deuxième décomposition de Hoeffding, nous obtenons une expression analytique pour les variances de ces estimateurs, en suivant les mêmes étapes que Hoeffding (1948) le fait pour $U_{\mathbf{n}}$. Leur expression montre que les statistiques $U_{\mathbf{n},N}$ et $U_{\mathbf{n},N,B}$ ont une précision limitée, c'est-à-dire une variance minimale, qui peut être largement supérieure à celle de $U_{\mathbf{n}}$ pour un certain h et des distributions spécifiques de X_1 et Z_1.

Repartitionnement des données pour les estimateurs distribués. Cette différence de variance entre les estimateurs distribués et U_n vient du fait que de nombreuses paires impliquées dans le calcul du second ne sont pas impliquées dans les calculs des premiers. Pour contrebalancer cet effet, nous proposons de faire la moyenne d'estimateurs calculés entre des procédures de repartitionnement des données, qui réattribuent les observations à chacun des groupes aléatoirement, afin que chaque paire impliquée dans $U_{\mathbf{n}}$ ait une chance d'être vue. Nous proposons deux estimateurs avec repartitionnement des données: $U_{\mathbf{n},N,T}$ (resp. $U_{\mathbf{n},N,B,T}$) qui fait la moyenne de T estimateurs $U_{\mathbf{n},N}$ (resp. $U_{\mathbf{n},N,B}$) calculé sur la base de T partitionnements différents des données. À mesure que T augmente, la variance de ces estimateurs se rapproche de la variance de $U_{\mathbf{n}}$ par le haut.

Variance relative de nos estimations. Nous fournissons des expressions analytiques pour la variance des estimateurs $U_{\mathbf{n}}, U_B, U_{\mathbf{n},N}, U_{\mathbf{n},N,B}, U_{\mathbf{n},N,T}$ et $U_{\mathbf{n},N,B,T}$, pour plusieurs façons de répartir les données sur les serveurs. À cet égard, nous considérons tout d'abord le tirage sans remplacement (SWOR), qui est pertinent lorsque l'on divise toutes les données sur plusieurs machines pour des raisons de contraintes d'espace. Ensuite, nous considérons avec le tirage avec remplacement (SWR), qui est pertinent lors de la sélection d'ensembles de données sur lesquels calculer plusieurs estimations en parallèle, par exemple lors d'une descente de gradient sur des *batchs* (petits sous-échantillons) de données. Nous supposons pour ces deux types de tirage que chaque serveur contient n/N éléments de l'échantillon $\mathcal{D}_n$ et m/N éléments de $\mathcal{Q}_m$, un réglage que nous appelons *prop*. L'assouplissement de cette hypothèse implique qu'il y a une probabilité non nulle de ne pas avoir d'éléments de $\mathcal{D}_n$ ou $\mathcal{Q}_m$ sur un serveur. Dans ce cas particulier, il faut fournir une valeur par défaut pour l'estimateur. Cela implique que la variance n'a pas de forme analytique simple et interprétable, donc nous fournissons des preuves empiriques que les variances observées sont du même ordre. De plus, nous caractérisons les paramètres h, n, m et les distributions de X_1, Z_1 pour lesquelles la procédure de répartition est importante.

Apprentissage avec des U-statistiques distribuées. Papa et al. (2015) a étudié la descente de gradient stochastique pour les U-statistiques. Bien que nous n'étendons pas leur analyse à nos estimateurs distribués, notre travail considère la minimisation de U-statistiques avec des méthodes de gradient stochastique dans un environnement distribué, en estimant un gradient avec $U_{\mathbf{n},N,B}$ et en repartitionnant les données tous les n_r itérations. Nous fournissons des preuves empiriques que la réduction de n_r donne une meilleure solution au processus d'optimisation en moyenne. En outre, la variance de la perte de la solution finale est beaucoup plus faible, ce qui montre l'augmentation la robustesse obtenue en repartitionnant les données.

Alors que la section 12.4.1 et cette section rendent compte respectivement de l'aspect de généralisation et de scalabilité pour l'ordonnancement par similarité, la section suivante se concentre sur les questions d'optimisation. Ainsi, la section suivante est plus prospective et donne des stratégies d'optimisation pour le problème d'ordonnancement par similarité sur des exemples-jouets.

12.4.3 Ordonnancement par similarité en pratique

Le développement de la littérature sur l'apprentissage profond de métriques est motivée par un besoin de critères qui correspondent aux exigences de l'identification biométrique, et peuvent être optimisés par descente de gradient. La plupart de ces critères sont basés sur une meilleure séparation des identités avec une heuristique, telle que le fait que toutes les instances associées à une même identité doivent correspondre au même point dans un espace de représentation, voir la *center loss* (Wen et al., 2016). Bien que ces critères soient raisonnables, ils sont mal reliés à l'évaluation des systèmes biométriques, qui est basée sur la courbe ROC. La courbe ROC est une fonction conçue pour évaluer la capacité d'une fonction de score à distinguer les éléments positifs des négatifs, ou, en biométrie, la capacité d'une fonction de similarité à distinguer les paires correspondantes des paires non correspondantes. Le lien entre l'ordonnancement bipartite et la biométrie nous motive à proposer des approches pratiques pour l'apprentissage de fonctions de similarité qui optimisent une mesure dérivée de la courbe ROC.

Analyses préliminaires de l'ordonnancement par similarité. Considérer l'apprentissage de similarité comme l'ordonnancement bipartite sur les paires est discuté en profondeur dans d'autres parties de la thèse sous le nom d'ordonnancement par similarité. En effet, nous avons fourni des résultats concernant la généralisation de l'optimisation en un point de la courbe ROC (pROC), parfois désigné sous le nom de classification de Neyman-Pearson (Scott and Nowak, 2006). Le problème pROC consiste à trouver un score s et un seuil t, qui donnent le taux de vrais positif le plus élevé possible tout en satisfaisant une limite supérieure sur le taux de faux positifs. De plus, nous avons fourni une extension de l'analyse et de la méthodologie associées à l'algorithme TreeRank à l'apprentissage de fonctions de similarité, pour lesquelles (Clémençon and Vayatis, 2009) avait prouvé initialement qu'il permettait d'apprendre une fonction de score s qui se rapproche de la courbe ROC optimale en norme sup. Enfin, nous avons motivé théoriquement l'utilisation de telles méthodes, car nous prouvons que les estimateurs gourmands en calcul impliqués peuvent être remplacés par des U-statistiques incomplètes, ce qui corrige les limites

computationnelles associées à l'ordonnancement par similarité.

Limites des analyses précédentes. Nos résultats théoriques justifient d'envisager l'optimisation du problème pROC ou l'utilisation de l'algorithme TreeRank. Cependant, elles n'impliquent pas d'approches évidentes pour optimiser le problème pROC, ni ne garantissent la performance empirique de notre extension de l'algorithme TreeRank. Ce chapitre de la thèse fournit des illustrations d'approches possibles, dont nous montrons le fonctionnement sur des données synthétiques simples. L'évaluation empirique approfondie de ces approches est une direction intéressante pour de futurs travaux.

Résoudre pROC pour des similarités linéaires. Bien que de nombreux articles traitent du problème pROC (Scott and Nowak, 2006; Clémençon and Vayatis, 2010; Rigollet and Tong, 2011), il subsiste un manque d'approches pratiques pour celui-ci, à l'exception de quelques propositions basées sur du partitionnement récursif, voir par exemple Scott and Nowak (2006). Soit un échantillon $\mathcal{D}_n := \{(x_i, y_i)\}_{i=1}^n \subset \mathcal{X} \times \{1, \dots, K\}$ de n points de données, où y_i est l'identité d'une observation, $x_i \in \mathcal{X}$ et $\mathcal{X} \subset \mathbb{R}^d$. Dans le cas de l'ordonnancement par similarité, pROC au niveau $\alpha \in (0, 1)$ pour une classe de fonctions candidates $\mathcal{S}$, avec $s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ pour tout $s \in \mathcal{S}$, s'écrit:

$$\max_{s \in \mathcal{S}} \frac{1}{n_+} \sum_{i<j} \mathbb{I}\{y_i = y_j\} \cdot s(x_i, x_j) \quad \text{avec} \quad \frac{1}{n_-} \sum_{i<j} \mathbb{I}\{y_i \neq y_j\} \cdot s(x_i, x_j) \leqslant \alpha. \tag{12.8}$$

Dans le cas général, Eq. (12.8) peut être très difficile à résoudre. Par exemple, si $\mathcal{S}$ est composé d'indicateurs d'ensembles dans $\mathcal{X} \times \mathcal{X}$, Eq. (12.8) peut ne pas être différentiable, ni continu. Cependant, si la famille $\mathcal{S}$ a une forme particulière, la résolution d'Eq. (12.8) peut être beaucoup plus facile. Par exemple, nous proposons une solution analytique lorsque $\mathcal{S}$ est l'ensemble de toutes les similarités bilinéaires bornées $(x, x') \mapsto x^\top A x$ avec $A \in \mathbb{R}^{d \times d}$ et $\|A\|_F \leqslant 1$, où $\|\cdot\|_F$ est la norme de Frobenius. Bien que nous montrions que cette approche donne des solutions raisonnables pour l'optimisation de la courbe ROC en un point pour des distributions très spécifiques, le cas général nécessite des familles de fonctions plus flexibles.

Résoudre pROC par descente de gradient. Pour traiter pROC avec des familles de fonctions candidates plus complexes, nous proposons une approche basée sur la descente de gradient pour minimiser un analogue de l'excès d'erreur pour l'estimation de l'ensemble de volume minimum de Scott and Nowak (2006), adapté à cet effet pour l'ordonnancement bipartite. Bien que nous démontrions son efficacité avec un simple classifieur linéaire et un exemple-jouet, l'approche est suffisamment souple pour s'adapter à des modèles plus complexes, ainsi qu'à une extension à l'ordonnancement par similarité. Toutefois, ses performances devront être démontrées empiriquement sur des exemples plus éloquents. Soit H et G les distributions de $X \mid Y = -1$ et $X \mid Y = +1$ respectivement, ainsi que R_α^* la région de rejet optimale pour l'optimisation de la courbe ROC en point, alors notre relaxation de l'excès d'erreur écrit:

$$\max_{(w,b) \in \mathbb{R}^d \times \mathbb{R}} \left(G(R_\alpha^*) - \widehat{G}(w, b)\right)_+ + \left(\widehat{H}(w, b) - \alpha\right)_+ \qquad \text{tel que} \qquad \|w\|_2^2 + b^2 \leqslant 1. \tag{12.9}$$

où $\widehat{G}$ et $\widehat{H}$ sont des versions empiriques relaxées de G et H respectivement, et $(x)_+ := \max(x, 0)$ pour tout $x \in \mathbb{R}$. En minimisant la fonction objectif dans Eq. (12.9) et en projetant les poids sur la balle l'unité, notre méthode retrouve les bonnes régions de rejet par descente de gradient, pour notre exemple-jouet. Bien que la quantité $G(R_\alpha^*)$ soit inconnue, nous supposons que les résultats ne sont pas extrêmement sensibles à cette valeur, et que des approximations raisonnables peuvent être proposées dans la plupart des applications. Le problème pROC est une approche raisonnable pour apprendre une similarité, mais il n'aborde pas la question de l'ordonnancement comme un problème global sur $\mathcal{X}$. Précisément, il se concentre sur la récupération d'un seul ensemble de niveau de la fonction de score, comme le démontre le lemme fondamental de Neyman-Pearson, alors que l'ordonnancement bipartite consiste à récupérer une relation d'ordre entre deux points quelconques de l'espace d'entrée.

TreeRank pour l'apprentissage de similarité. L'algorithme TreeRank de Clémençon and Vayatis (2009) traite l'ordonnancement bipartite comme un problème global. Précisément, il aborde celui-ci par une procédure de séparation récursive, qui résout des problèmes de classification binaire avec des erreurs pondérées asymétriquement entre les observations positives et négatives. Comme présenté précédemment, nos travaux ont étendu l'algorithme à l'apprentissage de similarité, ainsi que les garanties en norme sup pour la distance entre la courbe ROC du score appris et

la courbe ROC optimale. Pour illustrer notre variante de TreeRank pour l'ordonnancement par similarité, nous représentons visuellement la forme de nos régions candidates symétriques utilisées pour séparer l'espace.

Considérations pratiques pour TreeRank. Les deux inconvénients de TreeRank sont: 1) sa dépendance aux divisions initiales de l'espace, qui compromettent les performances si la famille de régions candidates est trop limitée, 2) et son caractère discontinu, qui est incompatible avec les hypothèses naturelles de nombreux cas pratiques. Une première réponse à ces limites est d'étendre l'idée de forêts aléatoires proposé dans Breiman (2001) à TreeRank, ce qui est proposé dans Clémençon et al. (2013). Nous montrons sur des données-jouets avec un rapport de vraisemblance continu dG/dH que les arbres d'ordonnancement moyennés peuvent corriger à la fois la mauvaise spécification initiale des régions candidates et la nature discrète des arbres d'ordonnancement. Précisément, la fonction de score moyennée que nous obtenons donne une courbe ROC presque indiscernable de la courbe ROC optimale, malgré son ensemble de valeurs limité — mais nombreux — et l'inadéquation de chaque arbre relativement au vrai rapport de vraisemblance.

Ces propositions esquissent une voie pour de nouveaux algorithmes résolvant l'ordonnancement par similarité, qui est une approche rationnelle pour traiter le problème d'identification biométrique. Cependant, il ne s'agit que d'illustrations sur des problèmes très simples. Trouver de nouvelles approches pour l'ordonnancement par similarité en pratique, idéalement sur des expériences à grande échelle qui correspondent à des scénarios pratiques en matière de biométrie, est une piste prometteuse pour de futurs travaux.

12.5 Fiabilité en apprentissage automatique

Outre l'apprentissage de similarité, la biométrie et en particulier la reconnaissance faciale incarne de nombreuses questions importantes en apprentissage automatique, comme le montrent les rapports du NIST (National Institute of Standards and Technology) sur les évaluations des entreprises de reconnaissance faciale (Grother and Ngan, 2019). Précisément, la reconnaissance faciale est confrontée à des questions concernant la robustesse des prédictions, les biais dans les données d'entraînement, ainsi que sur l'équité algorithmique. Les sous-sections de cette section abordent toutes ces questions séquentiellement. Bien qu'il existe une littérature abondante sur ces sujets en général, notre travail porte sur des situations difficiles et peu étudiées qui s'appliquent directement aux problèmes biométriques. Précisément: 1) concernant la robustesse des prédictions, nous considérons l'apprentissage d'une liste ordonnée d'identités candidates, comme le font certains problèmes d'identification biométrique, 2) pour le biais des données d'entraînement, nous rééquilibrons les instances d'entraînement en utilisant des informations de haut niveau, telle que la nationalité dans le cadre du contrôle aux frontières, 3) pour la question d'équité, nous nous concentrons sur les fonctions de score en tant que passerelle vers les fonctions de similarité, comme dans la partie II.

12.5.1 Ordonner les labels par probabilité

Dans les systèmes biométriques destinés aux enquêtes criminelles, un expert humain considère souvent les suspects les plus probables proposés par un système. En général, les problèmes de classification difficiles se concentrent sur les labels les plus probables, comme le fait par exemple le challenge ILSVRC (*ImageNet Large Scale Visual Recognition Challenge*), où Krizhevsky et al. (2012) et les articles suivants considèrent la précision en top-5 à côté de la précision en classification habituelle (en top-1). Dans notre travail, nous proposons une approche pour la prédiction d'une liste ordonnée de labels à partir de données de classification. Nous désignons ce problème par le nom de *label ranking* (LR, ordonnancement de labels). Précisément, nous proposons d'utiliser la fameuse technique *One-versus-One* (un contre un) pour la classification multiclasse, et nous dérivons des garanties pour cette approche.

Cadre probabiliste pour l'ordonnancement de labels (LR). Dans le cadre probabiliste de la classification multiclasse, nous considérons une paire aléatoire $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ avec $\mathcal{Y} = \{1, \dots, K\}$, ainsi que le risque $L(g) = \mathbb{P}\{g(X) \neq Y\}$ associé à un classifieur $g : \mathcal{X} \to Y$.

Le classifieur optimal pour $L(g)$ dans la classe de toutes les fonctions mesurables est le célèbre classifieur de Bayes g^*, défini comme suit:

$$g^*(x) := \underset{k\in\{1,\dots,K\}}{\arg\max}\ \eta_k(x),$$

où $\eta_k : x \mapsto \mathbb{P}\{Y = k \mid X = x\}$ est la probabilité a posteriori de la classe k pour tout $k \in \{1, \dots, K\}$. Nous appelons *label ranking* (LR) la tâche consistant à trouver une liste ordonnée des labels les plus probables. Cela revient à associer à tout $x \in \mathcal{X}$ une permutation $\sigma_x^* \in \mathfrak{S}_K$, telle que:

$$\eta_{\sigma_x^{-1}(1)} > \eta_{\sigma_x^{-1}(2)} > \cdots > \eta_{\sigma_x^{-1}(K)}. \tag{12.10}$$

Nous notons σ_X la permutation aléatoire associée.

Sur la régression de médiane d'ordonnancements (RMR). Un autre problème bien connu dans la littérature statistique concerne la prédiction d'un ordonnancement sur un ensemble de labels est la *régression de médiane d'ordonnancements* (RMR, *Ranking Median Regression*) (Tsoumakas et al., 2009; Vembu and Gärtner, 2010; Clémençon et al., 2018). La RMR considère une paire $(X, \Sigma) \in \mathcal{X} \times \mathfrak{S}_K$ et apprend à partir de données une règle d'ordonnancement $s : \mathcal{X} \to \mathfrak{S}_K$ qui minimise le risque:

$$R(s) := \mathbb{E}[d(\Sigma, s(X))], \tag{12.11}$$

où $d : \mathfrak{S}_K \times \mathfrak{S}_K \to \mathbb{R}_+$ est une fonction de perte symétrique. La distance d quantifie une distance entre les ordonnancements. La distance la plus connue est le τ *de Kendall* d_τ, qui est égal au nombre de désaccords sur les paires entre les deux permutations.

Solution optimale de la RMR. Des travaux antérieurs sur la RMR (Clémençon et al., 2018) ont montré que le minimiseur optimal de Eq. (12.11) pour des règles d'ordonnancement mesurables a une formulation analytique simple pour la distance d_τ, sous une hypothèse appelée la Transitivité Stochastique Stricte (SST). La SST suppose que les probabilités sur paires $p_{k,l}(x) := \mathbb{P}\{\Sigma(k) < \Sigma(l) \mid X = x\} =: 1 - p_{l,k}(x)$ pour tout $1 \leqslant k < l \leqslant K$ satisfont: pour tous $x \in \mathcal{X}$ et $(i, j, k) \in \{1, \dots, K\}^2$ avec $i \neq j$, nous avons $p_{i,j}(x) \neq 1/2$ et:

$$p_{i,j}(x) > 1/2 \qquad et \qquad p_{j,k}(x) > 1/2 \qquad \Rightarrow \qquad p_{i,k}(x) > 1/2.$$

Sous l'hypothèse du SST, la règle d'ordonnancement optimal pour d_τ s'écrit:

$$s_X^*(k) = 1 + \sum_{l\neq k} \mathbb{I}\{p_{k,l}(X) < 1/2\}. \tag{12.12}$$

Le LR en tant que RMR avec information partielle. Si la caractérisation de l'élément optimal est une extension bienvenue de la théorie de l'apprentissage usuelle (Devroye et al., 1996) au problème de RMR, l'ordonnancement sur les labels Σ n'est pas disponible en entier lorsqu'il s'agit de données de classification. Cependant, nos travaux montrent que si l'on considère la permutation aléatoire Σ comme générée par un modèle BLTP conditionnel (Korba, 2018) avec vecteur de préférence $\eta(X) = (\eta_1(X), \dots, \eta_K(X))$, alors il est possible de construire un Σ qui satisfasse $Y = \Sigma^{-1}(1)$ presque sûrement. Sur la base de cette observation, nous proposons de considérer le LR comme un problème de RMR avec l'information partielle $\Sigma^{-1}(1)$ sur la permutation aléatoire complète Σ.

Solutions optimales du LR avec la méthode One-versus-One (OVO). Nous pouvons calculer les expressions des probabilités sur paires $p_{k,l}(x)$ sous un modèle BTLP conditionnel avec vecteur de préférence $\eta(x)$. Précisément, nous avons $p_{k,l}(x) = \eta_k(x)/(\eta_k(x) + \eta_l(x))$ pour tout $x \in \mathcal{X}$ et $k < l$. Remarquons que $p_{k,l}(x)$ correspond à la probabilité de prédire k contre l pour le problème de classification *"One-Versus-One"* (OVO, un-contre-un). L'approche OVO a été étudiée en détail (Hastie and Tibshirani, 1997; Moreira and Mayoraz, 1998; Allwein et al., 2000; Fürnkranz, 2002) pour la résolution de la classification multiclasse à l'aide d'algorithmes de classification binaire. L'approche OVO consiste à apprendre $K(K-1)/2$ fonctions de décision, précisément un classifieur pour chaque classe k contre l avec $k < l$, et de prendre le vote majoritaire des $K(K-1)/2$ classifieurs. Le classifieur de Bayes pour le problème OVO (k, l) est $g_{k,l}^* : x \mapsto 2 \cdot \mathbb{I}\{p_{k,l}(x) \geqslant 1/2\} - 1$. Ainsi, Eq. (12.12) se réduit à:

$$s_X^*(k) = 1 + \sum_{l\neq k} \mathbb{I}\{g_{k,l}^*(X) = -1\}. \tag{12.13}$$

Notons que s^*_X correspond à σ^*_X dans Eq. (12.10), dès que tous les $\eta_k(X)$ sont distincts. Nous avons montré qu'une combinaison des solutions optimales de tous les $K(K-1)/2$ problèmes OVO implique une règle d'ordonnancement optimale s^*_X. Par conséquent, nous pouvons probablement dériver une bonne solution du problème LR à partir de bonnes solutions de tous les problèmes OVO.

Garanties pour le LR avec la méthode OVO. Nous proposons une solution au LR, qui utilise une combinaison des solutions de tous les problèmes empiriques de classification OVO. Ensuite, nous déduisons des garanties théoriques pour cette solution. Soit un échantillon $\mathcal{D}_n = \{(X_i, Y_i)\}_{i=1}^n$ de n copies *i.i.d.* de la paire aléatoire (X, Y), ainsi que la notation $Y_{k,l,i} = \mathbb{I}\{Y_i = l\} - \mathbb{I}\{Y_i = k\}$ pour tout $k < l$ et n'importe quel $i \in \{1, \dots, n\}$. Le risque empirique $\widehat{L}_{k,l}$ de $g : \mathcal{X} \to \{-1, +1\}$ pour la classification OVO de k contre l s'écrit:

$$\widehat{L}_{k,l}(g) := \frac{1}{n_k + n_l} \sum_{i:Y_i \in \{k,l\}} \mathbb{I}\{g(X_i) \neq Y_{k,l,i}\},$$

où $n_k = \sum_{i=1}^n \mathbb{I}\{Y_i = k\}$ pour tout $k \in \{1, \dots, K\}$. Nous écrivons $\widehat{g}_{k,l}$ le minimiseur de $\widehat{L}_{k,l}$ sur la classe de proposition fixe $\mathcal{G}$ de classifieurs binaires. Sur le modèle de Eq. (12.13), une solution empirique du LR s'écrit:

$$\widehat{s}_X(k) := 1 + \sum_{k \neq l} \mathbb{I}\{\widehat{g}_{k,l}(X) = -1\}.$$

Une simple inégalité de Boole implique:

$$\mathbb{P}\{\widehat{s}_X \neq s^*_X\} \leqslant \sum_{k<l} \mathbb{P}\{\widehat{g}_{k,l}(X) \neq g^*_{k,l}(X)\}. \tag{12.14}$$

Eq. (12.14) montre que la somme des probabilités de ne pas prédire la classe optimale pour chaque problème d'OVO borne la probabilité de ne pas prévoir l'ordonnancement optimal de labels en LR. De plus, une conséquence des hypothèses habituelles pour la dérivation de vitesses de généralisation rapide — présentée pour la première fois dans Mammen and Tsybakov (1995) et aussi détaillée dans Boucheron et al. (2005) — donne une borne supérieure sur la quantité à droite de Eq. (12.14) par les excès de risque des problèmes de classification k contre l. Sous une hypothèse de bruit standard, nous utilisons les vitesses d'apprentissage rapide usuelles en $O(n^{-1/(2-a)})$ pour l'excès d'erreur de chaque problème de classification k contre l, où $a \in (0, 1)$ est un paramètre de bruit. Combinés avec Eq. (12.14), ces résultats impliquent une borne de convergence en $O(n^{-a/(2-a)})$ pour la quantité $\mathbb{P}\{\widehat{s}_X \neq s^*_X\}$, ce qui est plus lent que la vitesse d'apprentissage usuelle en classification, en raison de la complexité inhérente dans le problème d'ordonnancement des labels.

Implications de l'analyse. Un analogue à l'erreur RMR de l'Eq. (12.11) pour le LR serait le risque suivant:

$$\mathcal{R}(s) := \mathbb{E}[d(s(X), \sigma^*_X)], \tag{12.15}$$

pour une règle d'ordonnancement $s : \mathcal{X} \to \mathfrak{S}_K$. Notons que, pour toute distance d bornée:

$$d(\sigma, \sigma') \leqslant \mathbb{I}\{\sigma \neq \sigma'\} \times \max_{\sigma_0, \sigma_1 \in \mathfrak{S}_K} d(\sigma_0, \sigma_1), \tag{12.16}$$

pour tout $\sigma, \sigma' \in \mathfrak{S}_K$. Eq. (12.16) implique une extension de nos garanties pour $\mathbb{P}\{\widehat{s}_X \neq s^*_X\}$ au risque Eq. (12.15) du minimiseur empirique. Incidemment, notre analyse du LR fournit la première garantie de généralisation à nombre d'instances fini pour l'approche OVO en classification multiclasse, en considérant le cas spécifique $k = 1$ pour les garanties sur la précision en top-k que nous fournissons.

En conclusion, nous avons proposé le nouveau mais naturel problème d'ordonnancement de labels (LR), qui consiste à apprendre la prédiction d'une liste ordonnée des labels les plus probables à partir de données de classification multiclasse. Bien que Korba et al. (2018) et Brinker and Hüllermeier (2019) donnent des approches pratiques à la régression de médiane d'ordonnancements (RMR) avec une information partielle, nos garanties théoriques sont nouvelles. Notre analyse s'inscrit parfaitement dans le cadre usuel de minimisation du risque empirique et exploite des résultats récents sur la RMR. Un sous-produit de notre analyse est la première borne de généralisation à échantillon fini sur l'approche OVO pour la classification multiclasse.

12.5.2 Correction des biais de sélection

Dans certains problèmes d'apprentissage statistique, la distribution P' des données d'entraînement $Z'_1, \ldots, Z'_n$ peut être différente de celle des données de test P. Cette configuration constitue un cas particulier d'apprentissage par transfert (Pan and Yang, 2010; Ben-David et al., 2010; Storkey, 2009; Redko et al., 2019). Notamment dans les problèmes de reconnaissance faciale, la population utilisée pour l'entraînement n'est souvent pas représentative de la population test, comme souligné dans Wang et al. (2019). Des informations auxiliaires sous forme de caractéristiques de haut niveau sont cependant souvent disponibles, comme la nationalité associée à un portrait en reconnaissance faciale. Notre travail porte sur l'apprentissage avec des données biaisées du point de vue de la minimisation du risque empirique (ERM). À cet égard, nous proposons une approche basée sur l'échantillonnage préférentiel qui traite: 1) de problèmes de classification où les probabilités de chaque classe diffèrent entre les phases d'entraînement et de test, 2) de situations où les données proviennent de populations stratifiées représentées différemment entre l'entraînement et le test, 3) de l'apprentissage PU, qui consiste à apprendre avec un échantillon de données positives et non labellisées (du Plessis et al., 2014), 4) et de l'apprentissage avec des données censurées (Fleming and Harrington, 2011). Notre analyse est soutenue par des résultats empiriques solides pour la classification sur la base de données ImageNet (Russakovsky et al., 2014), pour laquelle nous avons créé une information de strate à partir de concepts de plus haut niveau que les classes prédites.

Minimisation du risque empirique pondéré (WERM). Le but des algorithmes d'apprentissage est généralement de trouver un paramètre $\theta \in \Theta$, qui minimise l'espérance de risque $\mathcal{R}(\theta) = \mathbb{E}_P[\ell(\theta, Z)]$ sur les données de test, avec $\ell : \Theta \times \mathcal{Z} \to \mathbb{R}_+$ une fonction de perte mesurable. Pour approximer $\mathcal{R}(\theta)$, nous proposons un estimateur pondéré $\widetilde{\mathcal{R}}_{w,n}$ sur les données d'entraînement:

$$\widetilde{\mathcal{R}}_{w,n}(\theta) := \frac{1}{n} \sum_{i=1}^{n} w_i \cdot \ell(\theta, Z'_i),$$

où $w = (w_1, \ldots, w_n)$ est un vecteur de poids. Lorsque $P' = P$ et $w = (1, \ldots, 1)$, alors $\widetilde{\mathcal{R}}_{w,n}(\theta)$ est le risque empirique usuel, ainsi qu'un estimateur non biaisé de l'espérance du risque $\mathcal{R}(\theta)$ sur les données de test. Lorsque $P' \neq P$ et P est absolument continu par rapport à P', la méthode d'échantillonnage préférentiel (Wasserman, 2010, Section 25.3) suggère de fixer $w_i := \Phi(Z_i) =: w_i^*$ pour tout $i \in \{1, \ldots, n\}$, où $\Phi(z) := (dP/dP')(z)$ pour tout $z \in \mathcal{Z}$. Φ est le rapport de vraisemblance entre P' et P, donc $\widetilde{\mathcal{R}}_{w,n}$ est alors un estimateur non biaisé de $\mathcal{R}$.

Garanties de généralisation pour WERM. Nous dérivons ensuite des garanties de généralisation habituelles en $O(n^{-1/2})$ qui dépendent de la norme sup $\|\Phi\|_\infty$ du rapport de vraisemblance sur l'espace d'entrée $\mathcal{Z}$. Nos garanties montrent que la généralisation est meilleure lorsque les deux distributions P' et P sont similaires. Dans le cas général, le rapport de vraisemblance Φ est inconnu, ce qui limite l'applicabilité de cette technique. De plus, Φ est une fonction sur l'espace d'entrée $\mathcal{Z}$, ce qui rend son estimation peu pratique. Notre travail présente des situations pour lesquelles la fonction de vraisemblance Φ a une formulation simple, sous l'hypothèse qu'une information auxiliaire sur une relation entre les distributions P' et P est disponible.

WERM pour les probabilités de strates. Dans le contexte de la classification multiclasse, c'est-à-dire quand $Z = (X, Y) \in \mathcal{X} \times \mathcal{Y}$ avec $\mathcal{Y} = \{1, \ldots, K\}$ et $K \in \mathbb{N}$ est le nombre de classes, Φ a une forme simple lorsque la proportion en espérance $p_k = \mathbb{P}_{Z \sim P}\{Y = k\}$ de chaque classe k dans l'ensemble de données du test est connu. Dans ce contexte, les poids optimaux $w^* = (w_1^*, \ldots, w_n^*)$ satisfont $w_i^* = p_{Y_i}/p'_{Y_i}$ pour tout $i \in \{1, \ldots, n\}$, où $p'_k = \mathbb{P}_{P'}\{Y = k\}$ est la proportion en espérance de la classe k dans les données d'entraînement pour tout $k \in \{1, \ldots, K\}$. Notons que les p'_k peuvent être estimés à partir des données d'entraînement. Cependant cette stratégie de repondération ne dépend pas du fait que la prédiction des classes soit l'objectif, mais s'applique dès l'on connaît: les proportions de chaque strate pour une stratification quelconque sur la distribution test, et la strate associée à chaque instance de l'ensemble d'entraînement.

WERM pour l'apprentissage avec des données positives et non labellisées (PU). L'apprentissage avec des données positives et non labellisées (PU) a fait l'objet d'une attention croissante dans la récente littérature en apprentissage statistique (du Plessis and Sugiyama, 2014; du Plessis et al., 2014, 2015; Kiryo et al., 2017; Bekker et al., 2019). L'apprentissage PU considère

un problème de classification binaire, c'est-à-dire $Z = (X, Y) \in \mathcal{X} \times \{-1, +1\}$, et apprend un classifieur avec un échantillon de positifs et d'instances non labellisées. L'échantillon non labellisé est un mélange de données négatives et positives dans des proportions fixes. Notre travail montre que, en repondérant les instances de ces deux échantillons, nous obtenons un estimateur non biaisé du risque $\mathcal{R}$. Nous fournissons des garanties statistiques pour le minimiseur de notre risque estimé.

Expériences sur la base de données ImageNet. Finalement, nous fournissons une illustration numérique convaincante de l'efficacité de WERM sur la base de données ImageNet, une base utilisée pour évaluer les algorithmes de classification à grande échelle en vision par ordinateur (Russakovsky et al., 2014). Les classes d'ImageNet sont construites à partir de la base de données lexicale WordNet pour l'anglais (Fellbaum, 1998). En tant que telles, ces classes peuvent être regroupées en plusieurs concepts de haut niveau, qui constituent les strates de notre expérience de classification pondérée. Par exemple, la classe "flamant rose" est une sous-classe de la strate "oiseau". L'utilisation d'informations de haut niveau pour re-pondérer les données d'entraînement améliore considérablement les performances sur les données de test, mesuré en termes de précision en classification au top-1 et au top-5.

Bien qu'assurer la représentativité d'une base de données puisse aider à obtenir des prédictions satisfaisant des critères d'équité, il y a un intérêt croissant pour des mécanismes qui corrigent explicitement les biais inhérents aux données d'entraînement.

12.5.3 Équité dans l'apprentissage de fonctions de *scoring*

L'apprentissage d'un classifieur sous des contraintes d'équité a reçu beaucoup d'attention dans la littérature (Dwork et al., 2012; Zafar et al., 2017a; Donini et al., 2018; Barocas et al., 2019; Williamson and Menon, 2019; McNamara et al., 2019). Toutefois, les approches proposées pour assurer une équité en ordonnancement soit corrigent seulement la fonction de score avec une étape de post-traitement (Borkan et al., 2019; Beutel et al., 2019; Zehlike et al., 2017; Celis et al., 2018), soit s'attaquent à des notions originales d'équité, comme une équité d'exposition dans la présentation séquentielle d'ordonnancements (Singh and Joachims, 2018, 2019). De nombreux articles sur l'équité pour l'ordonnancement ont proposé différentes contraintes basées sur l'aire sous la courbe ROC (AUC), une mesure standard de la performance en ordonnancement. Notre travail propose tout d'abord: un cadre unifié pour des contraintes d'équité basées sur l'AUC, des garanties de généralisation pour la minimisation d'une perte qui intègre n'importe laquelle de ces contraintes, ainsi qu'une procédure pratique d'optimisation de cette perte basée sur la descente de gradient. Ensuite, nous montrons les limites des contraintes d'équité basées sur l'AUC, et proposons des contraintes plus fortes basées sur la courbe ROC. Pour finir, nous prouvons des garanties de généralisation, ainsi qu'une extension de notre procédure d'optimisation pour l'apprentissage avec un critère d'équité à nos nouvelles contraintes basées sur la ROC.

Cadre probabiliste pour l'équité en ordonnancement bipartite. Le cadre standard d'équité pour la classification binaire considère un triplet de variables aléatoires $(X, Y, Z) \in \mathcal{X} \times \{-1, 1\} \times \{0, 1\}$, où X est la variable aléatoire d'entrée, Y est la variable aléatoire binaire de sortie, et Z encode l'appartenance à un groupe protégé. Dans l'ordonnancement bipartite, nous apprenons une fonction de score $s : \mathcal{X} \to \mathbb{R}$ et nous l'évaluons par rapport à la manière dont elle projette les négatifs $Y = -1$ relativement aux positifs $Y = +1$ sur la droite réelle. Dans le contexte de l'équité, l'influence de Z sur la distribution des scores est important. Pour cela, nous introduisons les distributions conditionnelles suivantes d'un score s donné pour tout $z \in \{0, 1\}$:

$$H_s(t) := \mathbb{P}\{s(X) \leqslant t \mid Y = -1\} \qquad \text{et} \qquad H_s^{(z)}(t) := \mathbb{P}\{s(X) \leqslant t \mid Y = -1, Z = z\},$$
$$G_s(t) := \mathbb{P}\{s(X) \leqslant t \mid Y = +1\} \qquad \text{et} \qquad G_s^{(z)}(t) := \mathbb{P}\{s(X) \leqslant t \mid Y = +1, Z = z\}.$$

Bien que la courbe ROC sert usuellement à évaluer la performance en ordonnancement d'une fonction de score, c'est aussi un outil général pour évaluer les différences entre deux fonctions de répartition h et g sur $\mathbb{R}$. Dans ce contexte, la courbe ROC est connue sous le nom de courbe probability-probability (PP *plot*) de h et g. L'aire sous la courbe ROC (AUC) est un résumé scalaire de la courbe ROC, et est omniprésent dans la littérature sur l'ordonnancement. L'AUC sert généralement à évaluer les performances des algorithmes d'ordonnancement bipartite (Clémençon et al., 2008). Formellement, les ROC et AUC entre les deux fonctions de répartitions

h et g, s'écrivent:

$$\mathrm{ROC}_{h,g} : \alpha \in [0,1] \mapsto 1 - g \circ h^{-1}(1-\alpha) \qquad \text{et} \qquad \mathrm{AUC}_{h,g} := \int_0^1 \mathrm{ROC}_{h,g}(\alpha)\, d\alpha.$$

Des différences dans la répartition des positifs (ou des négatifs) entre les groupes protégés entraîne des écarts dans les taux d'erreur de ces groupes, observés pour la reconnaissance faciale dans Grother and Ngan (2019).

Critère unifié d'équité basé sur l'AUC. Pour corriger les écarts de taux d'erreur entre les groupes protégés, de nombreux auteurs — principalement issus de la communauté des systèmes de recommandation — ont proposé des contraintes d'équité basées sur l'AUC (Beutel et al., 2019; Borkan et al., 2019; Kallus and Zhou, 2019). Avec $D(s) := (H_s^{(0)}, H_s^{(1)}, G_s^{(0)}, G_s^{(1)})^\top$, ces contraintes s'écrivent:

$$\mathrm{AUC}_{\alpha^\top D(s), \beta^\top D(s)} = \mathrm{AUC}_{\alpha'^\top D(s), \beta'^\top D(s)}, \tag{12.17}$$

pour différentes valeurs de $(\alpha, \beta, \alpha', \beta') \in (\mathcal{P})^4$, où $\mathcal{P} = \{v \mid v \in \mathbb{R}_+^4, \mathbf{1}^\top v = 1\}$ désigne le 4-simplexe. Nos travaux montrent que: si la contrainte d'équité Eq. (12.17) est satisfaite lorsque la distribution $X|Y = y, Z = z$ ne dépend pas de $z \in \{0,1\}$ pour à la fois $y = -1$ et $y = +1$, alors Eq. (12.17) s'écrit comme la combinaison linéaire $\Gamma^\top C(s) = 0$ de cinq contraintes d'équité élémentaires $C(s) = (C_1(s), \ldots, C_5(s))$ avec $\Gamma \in \mathbb{R}^5$. Notre définition générale des contraintes d'équité basées sur l'AUC englobe toutes les mesures d'équité basées sur l'AUC proposées, et peut servir à en dériver de nouvelles. Plus important encore, elle ouvre la voie à des approches flexibles pour l'apprentissage de score sous une contrainte d'équité basée sur l'AUC.

Apprentissage sous contraintes d'équité basée sur l'AUC**.** Nous intégrons la contrainte d'équité basée sur l'AUC dans une pénalisation ajoutée à la fonction objectif, que nous maximisons:

$$\max_{s \in \mathcal{S}} \mathrm{AUC}_{H_s, G_s} - \lambda \left| \Gamma^\top C(s) \right|, \tag{12.18}$$

sur une famille de fonction de score $\mathcal{S}$. Le paramètre λ règle un compromis entre la précision en ordonnancement et le critère d'équité. Nous fournissons des garanties théoriques sur l'erreur de généralisation de notre critère Eq. (12.18), prouvées avec des inégalités de concentration sur les déviations des U-statistiques. Nous proposons un algorithme basé sur la descente de gradient qui optimise simplement une version relaxée d' Eq. (12.18). Avec la notation $\widetilde{\cdot}$ pour indiquer la relaxation d'une estimation empirique de l'AUC par la fonction logistique $\sigma : x \mapsto 1/(1+e^{-x})$, notre relaxation de la perte avec une contrainte particulière basée sur l'AUC s'écrit:

$$\widetilde{L}_\lambda(s) := \widetilde{\mathrm{AUC}}_{H_s, G_s} - \lambda \cdot c \left(\widetilde{\mathrm{AUC}}_{H_s^{(0)}, G_s^{(0)}} - \widetilde{\mathrm{AUC}}_{H_s^{(1)}, G_s^{(1)}} \right), \tag{12.19}$$

où $c \in [-1, 1]$ est un paramètre qui change au cours du processus d'apprentissage. Le paramètre c est modifié après un nombre fixé de n_{adapt} itérations de l'algorithme du gradient, selon que la différence dans la contrainte d'AUC de l'Eq. (12.19) est évaluée comme positive ou négative sur un ensemble de données de validation.

Limites des contraintes basées sur l'AUC**.** L'égalité entre deux AUC contraint les distributions concernées. Plus précisément, le théorème des accroissements finis montre qu'elle impose un point d'égalité pour les courbes ROC concernées. Cependant, ce point est inconnu a priori. De nombreuses applications — et en particulier la biométrie — se concentrent sur les performances du système pour de petits taux de faux positifs, c'est-à-dire des régions spécifiques de la courbe ROC. Imposer l'égalité des courbes ROC dans ces régions implique que des classifieurs obtenus par le seuillage du score satisfont un critère d'équité dans un contexte de classification. Pour faire en sorte que les courbes ROC soient égales dans une région spécifique, nous pouvons nous concentrer sur quelques points précis, comme usuellement en approximation numérique. Pour ces raisons, nous introduisons des contraintes basées sur les courbes ROC, qui imposent l'égalité de deux courbes ROC à des points précis.

Apprentissage sous contrainte d'équité basée sur la ROC**.** Considérons les courbes ROC entre les négatifs et positifs de chaque groupe sensible, c'est-à-dire $\mathrm{ROC}_{H_s^{(0)}, H_s^{(1)}}$ et $\mathrm{ROC}_{G_s^{(0)}, G_s^{(1)}}$, leur écart par rapport à la diagonale s'écrit:

$$\Delta_{F,\alpha}(s) := \mathrm{ROC}_{F_s^{(0)}, F_s^{(1)}}(\alpha) - \alpha,$$

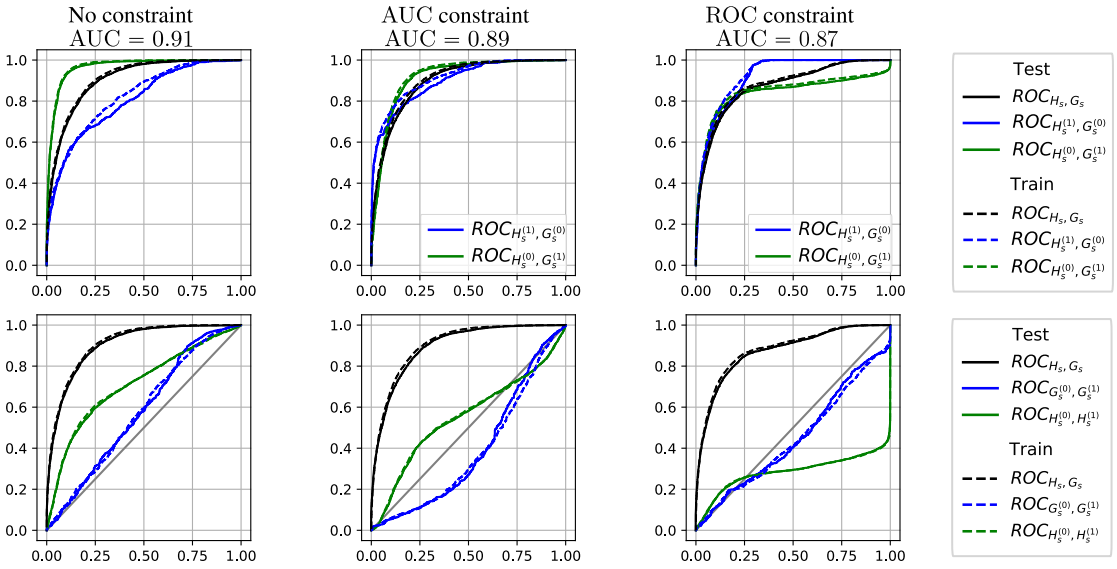


Figure 12.2: Courbes ROC obtenues respectivement: en apprenant un score sans contrainte, avec une contrainte basée sur l'AUC, et avec une contrainte basée sur la ROC. Nous avons choisi les paramètres de la contrainte basée sur la ROC dans le but d'obtenir $\mathrm{ROC}_{F_s^{(0)},F_s^{(1)}}(\alpha) = \alpha$ pour tout $\alpha \in [0, 1/4]$ et $F \in \{H, G\}$.

pour $F \in \{H, G\}$. Au lieu de contraintes d'équité basées sur les AUC, nous imposons $\Delta_{F,\alpha}(s) = 0$ avec des valeurs spécifiques $\alpha_F = [\alpha_F^{(1)}, \ldots, \alpha_F^{(m_F)}]$ de α pour tout $F \in \{H, G\}$. Pour cela, nous introduisons une perte L_Λ qui intègre ces contraintes avec des coefficients $\lambda_F = \left[\lambda_F^{(1)}, \ldots, \lambda_F^{(m_F)}\right]$ pour $F \in \{H, G\}$, qui s'écrit:

$$L_\Lambda(s) := \mathrm{AUC}_{H_s,G_s} - \sum_{k=1}^{m_H} \lambda_H^{(k)} \left|\Delta_{H,\alpha_H^{(k)}}(s)\right| - \sum_{k=1}^{m_G} \lambda_G^{(k)} \left|\Delta_{G,\alpha_G^{(k)}}(s)\right|,$$

où $\Lambda := (\alpha, \lambda_H, \lambda_G)$. Nous étendons nos garanties de généralisation aux contraintes d'équité basées sur la courbe ROC, à l'aide de bornes uniformes sur les courbes ROC. Pour prouver ce résultat, nous considérons un processus empirique indexé par la famille de points $\alpha \in [0, 1]$. Nous proposons également une stratégie d'optimisation analogue à celle utilisée pour Eq. (12.19). Notre stratégie comporte des paramètres de seuil, qui sont modifiés d'une manière similaire à c.

Résultats expérimentaux. Nous fournissons des preuves empiriques solides de la pertinence notre approche. Précisément, nous présentons dans la figure 12.2 les compromis entre équité et précision obtenus avec nos méthodes.

En conclusion, nous avons proposé de nouvelles approches pour aborder la question de l'équité algorithmique pour des problèmes d'ordonnancement. Tout d'abord, nous avons regroupé les contraintes basées sur l'AUC sous une seule définition générale. Ensuite, nous avons proposé des garanties théoriques, et une méthode pratique pour apprendre avec n'importe laquelle de ces contraintes à l'aide d'une descente de gradient. Nous avons souligné les limites des contraintes basées sur l'AUC, et avons proposé une approche basée sur la ROC qui correspond mieux aux conditions opérationnelles. Finalement, nous avons étendu nos garanties de généralisation et notre méthode pratique d'optimisation à notre nouvelle contrainte plus restrictive et plus flexible basée sur la courbe ROC.

12.6 Perspectives

En conclusion, la thèse aborde des problèmes importants en biométrie du point de vue de la théorie de l'apprentissage statistique. Notre travail propose des idées originales pour ces

problèmes, et soutient ces idées avec des résultats théoriques. Ces résultats peuvent être interprétés comme des garanties de sécurité qui sont vraies sous des hypothèses probabilistes. Notre travail est une réponse indispensable à l'augmentation rapide du volume de littérature expérimentale en apprentissage automatique que les chercheurs en biométrie doivent suivre. L'identification biométrique, et la reconnaissance faciale en particulier, incarnent simultanément de nombreux sujets en apprentissage automatique, comme l'apprentissage par paires, les biais d'échantillonnage ou l'ordonnancement. Pour cette raison, nous avons envisagé des versions stylisées de ces problèmes, car leur examen simultané obscurcirait notre discours, et court le risque d'être considéré comme anecdotique par la communauté de l'apprentissage automatique. La richesse des sujets abordés par la thèse résulte de cet impératif. L'élargissement de ce spectre pourrait être envisagé, par exemple en considérant l'extension des critères d'ordonnancement des meilleurs (Menon and Williamson, 2016, Section 9) au problème d'ordonnancement par similarité présenté ci-dessus.

Du point de vue de la biométrie, la perspective la plus importante pour cette thèse est de réaliser l'impact potentiel des méthodes présentées, en fournissant des preuves empiriques solides de leur pertinence dans des contextes pratiques. En effet, si l'adoption rapide des techniques d'apprentissage automatique par les entreprises privées ont stimulé la croissance du domaine, elle a également dirigé l'essentiel de l'attention sur les articles qui proposent des solutions sans équivoque à des problèmes industriels spécifiques. Un exemple notoire en reconnaissance faciale est Schroff et al. (2015). Dans ce contexte, la promotion de nos travaux nécessitera de trouver et de présenter pédagogiquement des expériences à grande échelle qui traitent de cas d'utilisation pratiques précis, ce qui est une orientation prometteuse pour les travaux futurs.

Enfin, nous pourrions étendre les différents sujets abordés dans la thèse. Dans le contexte de l'équité pour l'ordonnancement, une limite de notre analyse provient de l'absence d'une expression analytique pour la meilleure fonction de score sous une condition d'équité. Toutefois, elle est fournie dans le cas de l'équité pour la régression par Chzhen et al. (2020) par exemple. Pour l'ordonnancement bipartite, le fait de surmonter cet obstacle ouvre la voie à une extension des algorithmes de Clémençon and Vayatis (2009) — basés sur le partionnement de l'espace d'entrée — à l'apprentissage avec des contraintes d'équité. Une autre possibilité concerne l'extension des techniques présentées ici au cas de l'ordonnancement par similarité. En effet, cette extension donne un cadre qui correspond très étroitement aux considérations opérationnelles en matière de biométrie, et serait justifiée par l'intérêt actuel pour la correction explicite des biais en reconnaissance faciale. Le volet expérimental de ces travaux serait soutenu par la disponibilité de bases de données de visage adaptées (Wang et al., 2019). Une autre possibilité consiste à adresser les limites de nos travaux sur la minimisation des risques empiriques pondérés, en considérant les cas où la nature de la différence entre les données d'entraînement et de test n'est pas couverte par nos travaux. Par exemple, dans Sugiyama et al. (2007), les auteurs proposent d'estimer le rapport de vraisemblance à l'aide d'un petit échantillon qui suit la distribution de test comme information auxiliaire. Outre les exemples ci-dessus, d'autres extensions de chaque sujet abordé dans la thèse pourraient être envisagées.

En conclusion, la richesse des questions qui se posent en biométrie est un terrain fertile pour le développement à la fois de la théorie et de la pratique en apprentissage automatique. Cette richesse a engendré cette thèse et peut inspirer de futures recherches.

Bibliography

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. J. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Józefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. G. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. A. Tucker, V. Vanhoucke, V. Vasudevan, F. B. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467, 2016.

A. Agarwal, A. Beygelzimer, M. Dudík, J. Langford, and H. M. Wallach. A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 60–69. PMLR, 2018.

S. Agarwal, T. Graepel, R. Herbrich, S. Har-Peled, and D. Roth. Generalization bounds for the area under the ROC curve. *Journal of Machine Learning Research*, 6:393–425, 2005.

E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *Journal of Machine Learning Research*, 1:113–141, 2000.

P. K. Andersen, O. Borgan, R. D. Gill, and N. Keiding. *Statistical models based on counting processes*. Springer Science & Business Media, 2012.

M. A. Arcones and E. Giné. U-processes indexed by Vapnik-Červonenkis classes of functions with applications to asymptotics and bootstrap of U-statistics with estimated parameters. *Stochastic Processes and their Applications*, 52(1):17 – 38, 1994.

Y. Arjevani and O. Shamir. Communication complexity of distributed convex learning and optimization. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, pages 1756–1764, 2015.

G. Ausset, S. Clémençon, and F. Portier. Empirical risk minimization under random censorship: Theory and practice. *CoRR*, abs/1906.01908, 2019.

F. R. Bach, D. Heckerman, and E. Horvitz. Considering cost asymmetry in learning classifiers. *Journal of Machine Learning Research*, 7:1713–1741, 2006.

M. Balcan, A. Blum, S. Fine, and Y. Mansour. Distributed learning, communication complexity and privacy. In *COLT 2012 - The 25th Annual Conference on Learning Theory*, volume 23 of *JMLR Proceedings*, pages 26.1–26.22, 2012.

S. Barocas, M. Hardt, and A. Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2019.

G. D. Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall PTR, 1998.

Y. Bechavod and K. Ligett. Learning fair classifiers: A regularization-inspired approach. *CoRR*, abs/1707.00044, 2017.

J. Bekker, P. Robberechts, and J. Davis. Beyond the selected completely at random assumption for learning from positive and unlabeled data. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2019*, volume 11907 of *Lecture Notes in Computer Science*, pages 71–85. Springer, 2019.

R. Bekkerman, M. Bilenko, and J. Langford. *Scaling up Machine Learning: Parallel and Distributed Approaches*. Cambridge University Press, 2011.

A. Bellet and A. Habrard. Robustness and generalization for metric learning. *Neurocomputing*, 151:259–267, 2015.

A. Bellet, A. Habrard, and M. Sebban. *Metric Learning*. Morgan & Claypool Publishers, 2015a.

A. Bellet, Y. Liang, A. B. Garakani, M. Balcan, and F. Sha. A distributed frank-wolfe algorithm for communication-efficient sparse learning. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 478–486. SIAM, 2015b.

S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010.

A. Bendale and T. E. Boult. Towards open world recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pages 1893–1902. IEEE Computer Society, 2015.

P. Bertail and J. Tressou. Incomplete generalized U-statistics for food risk assessment. *Biometrics*, 62(1):66–74, 2006.

P. Bertail, S. Clémençon, and N. Vayatis. On bootstrapping the ROC curve. In *Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems*, pages 137–144. Curran Associates, Inc., 2008.

A. Beutel, J. Chen, T. Doshi, H. Qian, L. Wei, Y. Wu, L. Heldt, Z. Zhao, L. Hong, E. H. Chi, and C. Goodrow. Fairness in recommendation ranking through pairwise comparisons. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019*, pages 2212–2220. ACM, 2019.

A. J. Biega, K. P. Gummadi, and G. Weikum. Equity of attention: Amortizing individual fairness in rankings. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018*, pages 405–414. ACM, 2018.

G. Blom. Some properties of incomplete U-statistics. *Biometrika*, 63(3):573–580, 1976.

J. Bohné, Y. Ying, S. Gentric, and M. Pontil. Large margin local metric learning. In *Computer Vision - ECCV 2014 - 13th European Conference, Proceedings, Part II*, volume 8690 of *Lecture Notes in Computer Science*, pages 679–694. Springer, 2014.

T. Bolukbasi, K. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016*, pages 4349–4357, 2016.

D. Borkan, L. Dixon, J. Sorensen, N. Thain, and L. Vasserman. Nuanced metrics for measuring unintended bias with real data for text classification. In *Companion of The 2019 World Wide Web Conference, WWW 2019*, pages 491–500. ACM, 2019.

L. Bottou and O. Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems*, pages 161–168, 2007.

S. Boucheron, O. Bousquet, and G. Lugosi. Theory of classification : a survey of some recent advances. *ESAIM: Probability and Statistics*, 9:323–375, 2005.

S. Boucheron, G. Lugosi, and P. Massart. *Concentration Inequalities - A Nonasymptotic Theory of Independence*. Oxford University Press, 2013.

O. Bousquet, S. Boucheron, and G. Lugosi. Introduction to statistical learning theory. In *Advanced Lectures on Machine Learning, ML Summer Schools 2003*, volume 3176 of *Lecture Notes in Computer Science*, pages 169–207. Springer, 2003.

S. P. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

S. P. Boyd, C. Cortes, M. Mohri, and A. Radovanovic. Accuracy at the top. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012*, pages 962–970, 2012.

R. A. Bradley and M. E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, 1984.

K. Brinker and E. Hüllermeier. A reduction of label ranking to multiclass classification. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2019, Proceedings, Part III*, volume 11908 of *Lecture Notes in Computer Science*, pages 204–219. Springer, 2019.

S. Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3-4):231–357, 2015.

J. Bucklew. *Introduction to Rare Event Simulation*. Springer, 2010.

C. J. C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. N. Hullender. Learning to rank using gradient descent. In *Proceedings of the Twenty-Second International Conference on Machine Learning (ICML 2005)*, volume 119 of *ACM International Conference Proceeding Series*, pages 89–96. ACM, 2005.

Q. Cao, Z. Guo, and Y. Ying. Generalization bounds for metric and similarity learning. *Machine Learning*, 102(1):115–132, 2016.

P. Carbone, A. Katsifodimos, S. Ewen, V. Markl, S. Haridi, and K. Tzoumas. Apache flink™: Stream and batch processing in a single engine. *IEEE Data Engineering Bulletin*, 38(4):28–38, 2015.

L. E. Celis, D. Straszak, and N. K. Vishnoi. Ranking with fairness constraints. In *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018*, volume 107 of *LIPIcs*, pages 28:1–28:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 11:1109–1135, 2010.

J. Chen. Fair lending needs explainable models for responsible recommendation. *CoRR*, abs/1809.04684, 2018.

Z. Chen, B. Liu, R. Brachman, P. Stone, and F. Rossi. *Lifelong Machine Learning*. Morgan & Claypool Publishers, 2018.

F. Chollet et al. Keras, 2015.

E. Chzhen, C. Denis, M. Hebiri, L. Oneto, and M. Pontil. Fair Regression via Plug-in Estimator and Recalibration With Statistical Guarantees. HAL, archives ouvertes, Mar. 2020.

S. Clémençon and N. Vayatis. Tree-based ranking methods. *IEEE Transactions on Information Theory*, 55(9):4316–4336, 2009.

S. Clémençon and N. Vayatis. The RankOver algorithm: overlaid classification rules for optimal ranking. *Constructive Approximation*, 32:619–648, 2010.

S. Clémençon, G. Lugosi, and N. Vayatis. Ranking and empirical risk minimization of U-statistics. *The Annals of Statistics*, 36(2):844–874, 2008.

S. Clémençon, M. Depecker, and N. Vayatis. Adaptive partitioning schemes for bipartite ranking. *Machine Learning*, 83(1):31–69, 2011.

S. Clémençon, I. Colin, and A. Bellet. Scaling-up Empirical Risk Minimization: Optimization of Incomplete U-statistics. *Journal of Machine Learning Research*, 17(76):1–36, 2016.

S. Clémençon. A statistical view of clustering performance through the theory of U-processes. *Journal of Multivariate Analysis*, 124:42–56, 2014.

S. Clémençon and J. Jakubowicz. Scoring anomalies: a m-estimation formulation. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2013*, volume 31 of *JMLR Workshop and Conference Proceedings*, pages 659–667, 2013.

S. Clémençon and S. Robbiano. Building confidence regions for the ROC surface. *Pattern Recognitition Letters*, 46:67–74, 2014.

S. Clémençon and N. Vayatis. Ranking the best instances. *Journal of Machine Learning Research*, 8:2671–2699, 2007.

S. Clémençon and N. Vayatis. Empirical performance maximization for linear rank statistics. In *Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems*, pages 305–312. Curran Associates, Inc., 2008.

S. Clémençon and N. Vayatis. Tree-Based Ranking Methods. *IEEE Transactions on Information Theory*, 5(9):4136–4156, 2009.

S. Clémençon and N. Vayatis. Nonparametric estimation of the precision-recall curve. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009*, volume 382 of *ACM International Conference Proceeding Series*, pages 185–192. ACM, 2009.

S. Clémençon and N. Vayatis. Overlaying classifiers: a practical approach for optimal ranking. In *Constructive Approximation*, number 32, pages 313–320, 2010.

S. Clémençon, G. Lugosi, and N. Vayatis. Ranking and Empirical Minimization of U-Statistics. *The Annals of Statistics*, 36(2):844–874, 2008.

S. Clémençon, N. Vayatis, and M. Depecker. AUC optimization and the two-sample problem. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009*, pages 360–368. Curran Associates, Inc., 2009.

S. Clémençon, M. Depecker, and N. Vayatis. Ranking forests. *Journal of Machine Learning Research*, 14(1):39–73, 2013.

S. Clémençon, A. Korba, and E. Sibony. Ranking median regression: Learning to order through local consensus. In *Algorithmic Learning Theory, ALT 2018*, volume 83 of *Proceedings of Machine Learning Research*, pages 212–245. PMLR, 2018.

S. Clémençon and A. Thomas. Mass volume curves and anomaly ranking. *Electronic Journal of Statistics*, 12(2):2806–2872, 2018.

A. H. Copeland. A reasonable social welfare function. In *Seminar on applications of mathematics to social sciences, University of Michigan*, 1951.

T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition, 2009.

C. Cortes, M. Mohri, M. Riley, and A. Rostamizadeh. Sample selection bias correction theory. In *Algorithmic Learning Theory, 19th International Conference, ALT 2008. Proceedings*, volume 5254 of *Lecture Notes in Computer Science*, pages 38–53. Springer, 2008.

C. Cortes, Y. Mansour, and M. Mohri. Learning bounds for importance weighting. In *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010*, pages 442–450. Curran Associates, Inc., 2010.

F. Cucker and S. Smale. Best choices for regularization parameters in learning theory: On the bias-variance problem. *Foundations of Computational Mathematics*, 2:413–428, 01 2002.

A. Das, A. Dantcheva, and F. Brémond. Mitigating bias in gender, age and ethnicity classification: A multi-task convolution neural network approach. In *Computer Vision - ECCV 2018 Workshops - Proceedings, Part I*, volume 11129 of *Lecture Notes in Computer Science*, pages 573–585. Springer, 2018.

P. Davis. *Interpolation and approximation*. Dover Publications, 1975.

V. de la Pena and E. Giné. *Decoupling: from Dependence to Independence*. Springer, 1999.

J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019*, pages 4690–4699. Computer Vision Foundation / IEEE, 2019.

R. Deo. Machine learning in medicine. *Circulation*, 2015.

L. Devroye, L. Györfi, and G. Lugosi. *A probabilistic theory of pattern recognition*. Springer, 1996.

M. Deza and T. Huang. Metrics on permutations, a survey. 23, 1998.

J.-P. Doignon, A. Pekeč, and M. Regenwetter. The repeated insertion model for rankings: Missing link between two subset choice models. *Psychometrika*, 69:33–54, 03 2004.

P. Domingos. A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78–87, 2012.

M. Donini, L. Oneto, S. Ben-David, J. Shawe-Taylor, and M. Pontil. Empirical risk minimization under fairness constraints. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, pages 2796–2806, 2018.

M. C. du Plessis and M. Sugiyama. Class prior estimation from positive and unlabeled data. *IEICE Transactions on Information and Systems*, 97(5):1358–1362, 2014.

M. C. du Plessis, G. Niu, and M. Sugiyama. Analysis of learning from positive and unlabeled data. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, pages 703–711, 2014.

M. C. du Plessis, G. Niu, and M. Sugiyama. Convex formulation for learning from positive and unlabeled data. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1386–1394, 2015.

R. M. Dudley. *Uniform Central Limit Theorems*. Cambridge University Press, 1999.

C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. S. Zemel. Fairness through awareness. In *Innovations in Theoretical Computer Science 2012*, pages 214–226. ACM, 2012.

M. Egozcue and L. F. García. The variance upper bound for a mixed random variable. *Communications in Statistics - Theory and Methods*, 47(22):5391–9395, 2018.

C. Fellbaum, editor. *WordNet: an electronic lexical database*. MIT Press, 1998.

P. Festa, P. Pardalos, and M. C. Resende. *Feedback Set Problems*, pages 209–258. Springer US, Boston, MA, 1999.

T. R. Fleming and D. P. Harrington. *Counting processes and survival analysis*. John Wiley & Sons, 2011.

Y. Freund, R. D. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.

N. Furl, P. J. Phillips, and A. O'Toole. Face recognition algorithms and the other-race effect: Computational mechanisms for a developmental contact hypothesis. *Cognitive Science*, 26: 797–815, 11 2002.

J. Fürnkranz. Round robin classification. *Journal of Machine Learning Research*, 2:721–747, 2002.

J. Garcke and T. Vanck. Importance weighted inductive transfer learning for regression. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2014. Proceedings, Part I*, volume 8724 of *Lecture Notes in Computer Science*, pages 466–481. Springer, 2014.

K. Gates. *Our Biometric Future: Facial Recognition Technology and the Culture of Surveillance*. NYU Press, 2011.

J. Goldberger, S. T. Roweis, G. E. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *Advances in Neural Information Processing Systems 17 [Neural Information Processing Systems, NIPS 2004]*, pages 513–520, 2004.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

P. Grother and M. Ngan. Face Recognition Vendor Test (FRVT) — Performance of Automated Gender Classification Algorithms. Technical Report NISTIR 8052, National Institute of Standards and Technology (NIST), 2019.

M. Guillaumin, J. J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *IEEE 12th International Conference on Computer Vision, ICCV 2009*, pages 498–505. IEEE Computer Society, 2009.

Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *Computer Vision - ECCV 2016 - 14th European Conference, Proceedings, Part III*, volume 9907 of *Lecture Notes in Computer Science*, pages 87–102. Springer, 2016.

L. Györfi. *Principles of Nonparametric Learning*. Springer, 2002.

M. Hardt, E. Price, and N. Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016*, pages 3315–3323, 2016.

M. A. Hasnat, J. Bohné, J. Milgram, S. Gentric, and L. Chen. von Mises-Fisher Mixture Model-based Deep learning: Application to Face Verification. *CoRR*, abs/1706.04264, 2017.

T. Hastie and R. Tibshirani. Classification by pairwise coupling. In *Advances in Neural Information Processing Systems 10, [NIPS Conference, 1997]*, pages 507–513. The MIT Press, 1997.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. pages 770–778, 2016.

L. A. Hendricks, K. Burns, K. Saenko, T. Darrell, and A. Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *Computer Vision - ECCV 2018 - 15th European Conference, Proceedings, Part III*, volume 11207 of *Lecture Notes in Computer Science*, pages 793–811. Springer, 2018.

W. Hoeffding. A class of statistics with asymptotically normal distribution. *The Annals of Mathematical Statistics*, 19:293–325, 1948.

W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

F. Hsieh and B. W. Turnbull. Nonparametric and semiparametric estimation of the receiver operating characteristic curve. *The Annals of Statistics*, 24(1):25–40, 1996.

J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems*, pages 601–608. MIT Press, 2006.

J. Huo, Y. Gao, Y. Shi, and H. Yin. Cross-modal metric learning for auc optimization. *IEEE Transactions on Neural Networks and Learning Systems*, PP(99):1–13, 2018.

H. D. III, J. M. Phillips, A. Saha, and S. Venkatasubramanian. Protocols for learning classifiers on distributed data. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2012*, volume 22 of *JMLR Proceedings*, pages 282–290, 2012.

A. Jain, L. Hong, and S. Pankanti. Biometric identification. *Communications of the ACM*, 43(2): 90–98, 2000.

A. K. Jain, A. Ross, and S. Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.

A. K. Jain, A. A. Ross, and K. Nandakumar. *Introduction to Biometrics*. Springer, 2011.

R. Jin, S. Wang, and Y. Zhou. Regularized distance metric learning: Theory and algorithm. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009*, pages 862–870. Curran Associates, Inc., 2009.

T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142. ACM, 2002.

M. Jordan. On statistics, computation and scalability. *Bernoulli*, 19(4):1378–1390, 2013.

J. Jost. *Riemannian Geometry and Geometric Analysis*. Springer, 2011.

N. Kallus and A. Zhou. The fairness of risk scores beyond classification: Bipartite ranking and the XAUC metric. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, pages 3433–3443. 2019.

E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481, 1958.

J. G. Kemeny. Mathematics without numbers. *Daedalus*, (88):571–591, 1959.

R. Kiryo, G. Niu, M. C. du Plessis, and M. Sugiyama. Positive-unlabeled learning with non-negative risk estimator. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 1675–1685, 2017.

J. M. Kleinberg, S. Mullainathan, and M. Raghavan. Inherent trade-offs in the fair determination of risk scores. In *8th Innovations in Theoretical Computer Science Conference, ITCS 2017*, volume 67 of *LIPIcs*, pages 43:1–43:23. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

A. Korba. *Learning from ranking data : theory and methods*. PhD thesis, 2018. Thèse de doctorat dirigée par Stephan Clémençon - Mathématiques appliquées - Université Paris-Saclay (ComUE) 2018.

A. Korba, S. Clémençon, and E. Sibony. A learning theory of ranking aggregation. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017*, volume 54 of *Proceedings of Machine Learning Research*, pages 1001–1010. PMLR, 2017.

A. Korba, A. Garcia, and F. d'Alché-Buc. A structured prediction approach for label ranking. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, pages 9008–9018. 2018.

M. Köstinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2288–2295. IEEE Computer Society, 2012.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012*, pages 1106–1114. 2012.

B. Kulis. Metric Learning: A Survey. *Foundations and Trends in Machine Learning*, 5(4): 287–364, 2012.

P. Laforgue and S. Clémençon. Statistical learning from biased training samples. *CoRR*, abs/1906.12304, 2019.

A. J. Lee. *U-statistics: Theory and practice*. Marcel Dekker, Inc., New York, 1990.

E. Lehmann and J. P. Romano. *Testing Statistical Hypotheses*. Springer Texts in Statistics. Springer-Verlag New York, LLC, New York, third edition, 2005.

T.-Y. Liu. *Learning to Rank for Information Retrieval*. Springer, 2011.

W. Liu, X. Tian, D. Tao, and J. Liu. Constrained metric learning via distance gap maximization. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010*. AAAI Press, 2010.

Z. Liu, J. Yang, H. Liu, and W. Wang. Transfer learning by sample selection bias correction and its application in communication specific emitter identification. *Journal of Communication*, 11 (4):417–427, 2016.

R. D. Luce. *Individual Choice Behavior*. Wiley, 1959.

E. Mammen and A. Tsybakov. Smooth discrimination analysis. *The Annals of Statistics*, 27(6): 1808–1829, 1999.

E. Mammen and A. B. Tsybakov. Asympotical minimax recovery of the sets with smooth boundaries. *The Annals of Statistics*, 23(2):502–524, 1995.

B. Mason, L. Jain, and R. D. Nowak. Learning low-dimensional metrics. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 4139–4147, 2017.

P. Massart and E. Nédélec. Risk bounds for statistical learning. *Annals of Statistics*, 34(5), 2006.

B. McFee and G. R. G. Lanckriet. Metric learning to rank. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 775–782. Omnipress, 2010.

D. McNamara, C. S. Ong, and R. C. Williamson. Costs and benefits of fair representation learning. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, 2019.

A. K. Menon and R. C. Williamson. Bipartite ranking: a risk-theoretic perspective. *Journal of Machine Learning Research*, 17(195):1–102, 2016.

A. K. Menon and R. C. Williamson. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency, FAT 2018*, volume 81 of *Proceedings of Machine Learning Research*, pages 107–118. PMLR, 2018.

A. E. Mesaoudi-Paul, E. Hüllermeier, and R. Busa-Fekete. Ranking distributions based on noisy sorting. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 3469–3477. PMLR, 2018.

M. Mignotte. *Mathematics for Computer Algebra*. Springer-Verlag, Berlin, Heidelberg, 1992.

S. Minaee, E. Azimi, and A. Abdolrashidi. Fingernet: Pushing the limits of fingerprint recognition using convolutional neural network. *CoRR*, abs/1907.12956, 2019.

M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of Machine Learning*. MIT Press, 2012.

M. Moreira and E. Mayoraz. Improved pairwise coupling classification with correcting classifiers. In *Machine Learning: ECML-98, 10th European Conference on Machine Learning, Proceedings*, volume 1398 of *Lecture Notes in Computer Science*, pages 160–171. Springer, 1998.

K. Musgrave, S. Belongie, and S.-N. Lim. A metric learning reality check. *CoRR*, abs/2003.08505, 2020.

S. Nagpal, M. Singh, R. Singh, M. Vatsa, and N. Ratha. Deep learning for face recognition: Pride or prejudiced? *arXiv preprint arXiv:1904.01219*, 2019.

C. O'Neil. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown Publishing Group, USA, 2016.

O. B. P. Bartlett and S. Mendelson. Localized rademacher complexities. *The Annals of Statistics*, 33(1):497–1537, 2005.

S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

G. Papa, S. Clémençon, and A. Bellet. SGD algorithms based on incomplete u-statistics: Large-scale minimization of empirical risk. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, pages 1027–1035, 2015.

A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 1965.

O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *Proceedings of the British Machine Vision Conference 2015, BMVC 2015*, pages 41.1–41.12. BMVA Press, 2015.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

K. B. Petersen and M. S. Pedersen. The matrix cookbook, 2008.

P. J. Phillips, F. Jiang, A. Narvekar, J. H. Ayyad, and A. J. O'Toole. An other-race effect for face recognition algorithms. *ACM Trans. Appl. Percept.*, 8(2):14:1–14:11, 2011.

R. L. Plackett. The analysis of permutations. *Applied Statistics*, 2(24):193–202, 1975.

G. Pleiss, M. Raghavan, F. Wu, J. M. Kleinberg, and K. Q. Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 5680–5689, 2017.

W. Polonik. Measuring Mass Concentrations and Estimating Density COntour Clusters - An Excess Mass Approach. *The Annals of Statistics*, 23(3):855–881, 1995.

W. Polonik. Minimum volume sets and generalized quantile processes. *Stochastic Processes and their Applications*, 69(1):1–24, 1997.

J. Quinlan. Induction of Decision Trees. *Machine Learning*, 1(1):1–81, 1986.

I. Redko, E. Morvant, A. Habrard, M. Sebban, and Y. Bennani. *Advances in Domain Adaptation Theory*. Elsevier, 2019.

P. Rigollet and X. Tong. Neyman-pearson classification, convexity and stochastic constraints. *Journal of Machine Learning Research*, 12:2831–2855, 2011.

S. Ruder. An overview of gradient descent optimization algorithms. *CoRR*, abs/1609.04747, 2016.

C. Rudin. Ranking with a p-norm push. In *Learning Theory, 19th Annual Conference on Learning Theory, COLT 2006*, volume 4005 of *Lecture Notes in Computer Science*, pages 589–604. Springer, 2006.

C. Rudin, C. Wang, and B. Coker. The age of secrecy and unfairness in recidivism prediction. *CoRR*, abs/1811.00731, 2018.

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015.

C. Scott. Performance measures for neyman-pearson classification. *IEEE Transactions on Information Theory*, 53:2852–2863, 2007.

C. Scott and R. Nowak. A Neyman-Pearson approach to statistical learning. *IEEE Transactions on Information Theory*, 51(11):3806–3819, Nov 2005.

C. Scott and R. Nowak. Learning minimum volume sets. *Journal of Machine Learning Research*, 7:665–704, 2006.

R. Sedgewick and K. Wayne. *Algorithms, 4th Edition.* Addison-Wesley, 2011.

R. J. Serfling. *Approximation theorems of mathematical statistics.* Wiley, 1980.

S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms.* Cambridge University Press, USA, 2014.

C. Shen, J. Kim, L. Wang, and A. van den Hengel. Positive Semidefinite Metric Learning Using Boosting-like Algorithms. *Journal of Machine Learning Research*, 13:1007–1036, 2012.

G. Shorack. Probability for Statisticians. *Springer*, 2000.

G. Shorack and J. a. Wellner. *Empirical Processes with applications to Statistics.* SIAM, 1989.

A. Singh and T. Joachims. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018*, pages 2219–2228. ACM, 2018.

A. Singh and T. Joachims. Policy learning for fairness in ranking. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019*, pages 5427–5437, 2019.

S. Smale and D.-X. Zhou. Estimating the approximation error in learning theory. *Analysis and Applications*, 01:17–41, 2003.

V. Smith, S. Forte, C. Ma, M. Takác, M. I. Jordan, and M. Jaggi. CoCoA: A General Framework for Communication-Efficient Distributed Optimization. *Journal of Machine Learning Research*, 18(230):1–49, 2018.

A. Storkey. When training and test sets are different: characterizing learning transfer. *Dataset shift in machine learning*, pages 3–28, 2009.

E. Strubell, A. Ganesh, and A. McCallum. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Volume 1: Long Papers*, pages 3645–3650, 2019.

M. Sugiyama, S. Nakajima, H. Kashima, P. von Bünau, and M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems*, pages 1433–1440. Curran Associates, Inc., 2007.

R. Szeliski. *Computer Vision: Algorithms and Applications.* Springer, 2011.

Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, pages 1701–1708. IEEE Computer Society, 2014.

G. Tsoumakas, I. Katakis, and I. Vlahavas. Mining multi-label data. In *Data mining and knowledge discovery handbook*, pages 667–685. Springer, 2009.

M. A. Turk and A. Pentland. Face recognition using eigenfaces. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 1991*, pages 586–591. IEEE, 1991.

L. van der Maaten and G. Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

A. W. van der Vaart. Asymptotic Statistics. *Cambridge Series in Statistical and Probabilistic Mathematics*, 2000.

A. W. van der Vaart and J. a. Wellner. *Weak convergence and empirical processes*. 1996.

V. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

S. Vembu and T. Gärtner. Label ranking algorithms: A survey. In *Preference learning*, pages 45–64. Springer, 2010.

N. Verma and K. Branson. Sample complexity of learning mahalanobis distance metrics. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, pages 2584–2592, 2015.

R. Vogel, A. Bellet, and S. Clémençon. A probabilistic theory of supervised similarity learning for pointwise ROC curve optimization. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 5062–5071. PMLR, 2018.

F. Wang, J. Cheng, W. Liu, and H. Liu. Additive margin softmax for face verification. *IEEE Signal Processing Letters*, 25(7):926–930, 2018.

M. Wang and W. Deng. Deep face recognition: A survey. *CoRR*, abs/1804.06655, 2018.

M. Wang, W. Deng, J. Hu, X. Tao, and Y. Huang. Racial faces in the wild: Reducing racial bias by information maximization adaptation network. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019*, pages 692–702. IEEE, 2019.

L. Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer, 2010.

K. Q. Weinberger and L. K. Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. *Journal of Machine Learning Research*, 10:207–244, 2009.

Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *Computer Vision - ECCV 2016 - 14th European Conference, Proceedings, Part VII*, volume 9911 of *Lecture Notes in Computer Science*, pages 499–515. Springer, 2016.

J. Weston and C. Watkins. Support vector machines for multi-class pattern recognition. In *ESANN 1999, 7th European Symposium on Artificial Neural Networks*, pages 219–224, 1999.

R. C. Williamson and A. K. Menon. Fairness risk measures. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019*, volume 97 of *Proceedings of Machine Learning Research*, pages 6786–6797. PMLR, 2019.

B. E. Woodworth, S. Gunasekar, M. I. Ohannessian, and N. Srebro. Learning non-discriminatory predictors. In *Proceedings of the 30th Conference on Learning Theory, COLT 2017*, volume 65 of *Proceedings of Machine Learning Research*, pages 1920–1953. PMLR, 2017.

T. Wu, C. Lin, and R. Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5:975–1005, 2004.

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

P. Xie and E. P. Xing. Large scale distributed distance metric learning. *CoRR*, abs/1412.5949, 2014.

E. P. Xing, A. Y. Ng, M. I. Jordan, and S. J. Russell. Distance metric learning with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15 [Neural Information Processing Systems, NIPS 2002]*, pages 505–512. MIT Press, 2002.

E. P. Xing, Q. Ho, W. Dai, J. K. Kim, J. Wei, S. Lee, X. Zheng, P. Xie, A. Kumar, and Y. Yu. Petuum: A New Platform for Distributed Machine Learning on Big Data. *IEEE Transactions on Big Data*, 1(2):49–67, 2015.

M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017*, pages 1171–1180. ACM, 2017a.

M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017*, volume 54 of *Proceedings of Machine Learning Research*, pages 962–970. PMLR, 2017b.

M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi. Fairness constraints: A flexible approach for fair classification. *Journal of Machine Learning Research*, 20(75):1–42, 2019.

M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica. Spark: Cluster computing with working sets. In *2nd USENIX Workshop on Hot Topics in Cloud Computing, HotCloud'10*. USENIX Association, 2010.

M. Zehlike, F. Bonchi, C. Castillo, S. Hajian, M. Megahed, and R. Baeza-Yates. Fa*ir: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017*, pages 1569–1578. ACM, 2017.

J. Zhao, T. Wang, M. Yatskar, V. Ordonez, and K. Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017*, pages 2979–2989. Association for Computational Linguistics, 2017.

P. Zhao, S. C. H. Hoi, R. Jin, and T. Yang. Online AUC maximization. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, pages 233–240. Omnipress, 2011.

B. Zou, H. Zhang, and Z. Xu. Learning from uniformly ergodic markov chains. *Journal of Complexity*, 25(2):188–200, 2009.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

Titre : Ordonnancement par Similarité pour la Biométrie: Théorie et Pratique

Mots clés : Biométrie, Apprentissage Statistique, Ordonnancement Bipartite, Apprentissage de Similarité, Biais algorithmique

Résumé : L'augmentation rapide de la population combinée à la mobilité croissante des individus a engendré le besoin de systèmes de gestion d'identités sophistiqués. À cet effet, le terme biométrie se réfère généralement aux méthodes permettant d'identifier les individus en utilisant des caractéristiques biologiques ou comportementales. Les méthodes les plus populaires, c'est-à-dire la reconnaissance d'empreintes digitales, d'iris ou de visages, se basent toutes sur des méthodes de vision par ordinateur. L'adoption de réseaux convolutifs profonds, rendue possible par le calcul générique sur processeur graphique, ont porté les récentes avancées en vision par ordinateur. Ces avancées ont permis une amélioration drastique des performances des méthodes conventionnelles en biométrie, ce qui a accéléré leur adoption pour des usages concrets, et a provoqué un débat public sur l'utilisation de ces techniques. Dans ce contexte, les concepteurs de systèmes biométriques sont confrontés à un grand nombre de challenges dans l'apprentissage de ces réseaux.
Dans cette thèse, nous considérons ces challenges du point de vue de l'apprentissage statistique théorique, ce qui nous amène à proposer ou esquisser des solutions concrètes. Premièrement, nous répondons à une prolifération de travaux sur l'apprentissage de similarité pour les réseaux profonds, qui optimisent des fonctions objectif détachées du but naturel d'ordonnancement recherché en biométrie. Précisément, nous introduisons la notion d'*ordonnancement par similarité*, en mettant en évidence la relation entre l'ordonnancement bipartite et la recherche d'une similarité adaptée à l'identification biométrique. Nous étendons ensuite la théorie sur l'ordonnancement bipartite à ce nouveau problème, tout en l'adaptant aux spécificités de l'apprentissage sur paires, notamment concernant son coût computationnel.
Les fonctions objectif usuelles permettent d'optimiser la performance prédictive, mais de récents travaux ont mis en évidence la nécessité de prendre en compte d'autres facteurs lors de l'entraînement d'un système biométrique, comme les biais présents dans les données, la robustesse des prédictions ou encore des questions d'équité. La thèse aborde ces trois exemples, en propose une étude statistique minutieuse, ainsi que des méthodes pratiques qui donnent les outils nécessaires aux concepteurs de systèmes biométriques pour adresser ces problématiques, sans compromettre la performance de leurs algorithmes.

Titre : Similarity Ranking for Biometrics: Theory and Practice

Keywords : Biometrics, Statistical Learning Theory, Ranking, Similarity Learning, Algorithmic biais

Abstract : The rapid growth in population, combined with the increased mobility of people has created a need for sophisticated identity management systems. For this purpose, biometrics refers to the identification of individuals using behavioral or biological characteristics. The most popular approaches, *i.e.* fingerprint, iris or face recognition, are all based on computer vision methods. The adoption of deep convolutional networks, enabled by general purpose computing on graphics processing units, made the recent advances in computer vision possible. These advances have led to drastic improvements for conventional biometric methods, which boosted their adoption in practical settings, and stirred up public debate about these technologies. In this respect, biometric systems providers face many challenges when learning those networks.
In this thesis, we consider those challenges from the angle of statistical learning theory, which leads us to propose or sketch practical solutions. First, we answer to the proliferation of papers on similarity learning for deep neural networks that optimize objective functions that are disconnected with the natural ranking aim sought out in biometrics. Precisely, we introduce the notion of *similarity ranking*, by highlighting the relationship between bipartite ranking and the requirements for similarities that are well suited to biometric identification. We then extend the theory of bipartite ranking to this new problem, by adapting it to the specificities of pairwise learning, particularly those regarding its computational cost.
Usual objective functions optimize for predictive performance, but recent work has underlined the necessity to consider other aspects when training a biometric system, such as dataset bias, prediction robustness or notions of fairness. The thesis tackles all of those three examples by proposing their careful statistical analysis, as well as practical methods that provide the necessary tools to biometric systems manufacturers to address those issues, without jeopardizing the performance of their algorithms.

Institut Polytechnique de Paris
91120 Palaiseau, France