Robustness and Fairness in Machine Learning

by

Nikola Konstantinov

February, 2022

A thesis submitted to the
Graduate School
of the
Institute of Science and Technology Austria
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Committee in charge:

Prof. Edouard Hannezo, Chair
Prof. Christoph H. Lampert
Prof. Dan Alistarh
Prof. Ingo Steinwart

The thesis of Nikola Konstantinov, titled *Robustness and Fairness in Machine Learning*, is approved by:

Supervisor: Prof. Christoph H. Lampert, IST Austria, Klosterneuburg, Austria

Signature: _____

Committee Member: Prof. Dan Alistarh, IST Austria, Klosterneuburg, Austria

Signature: _____

Committee Member: Prof. Ingo Steinwart, University of Stuttgart, Stuttgart, Germany

Signature: _____

Defense Chair: Prof. Edouard Hannezo, IST Austria, Klosterneuburg, Austria

Signature: _____

Signed page is on file

IST Austria Thesis, ISSN: 2663-337X

ISBN: 978-3-99078-015-2

I hereby declare that this thesis is my own work and that it does not contain other people's work without this being so stated; this thesis does not contain my previous work without this being stated, and the bibliography contains all the literature that I used in writing the dissertation.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee, and that this thesis has not been submitted for a higher degree to any other university or institution.

I certify that any republication of materials presented in this thesis has been approved by the relevant publishers and co-authors.

Signature: _____

Nikola Konstantinov
February, 2022

Signed page is on file

Abstract

Because of the increasing popularity of machine learning methods, it is becoming important to understand the impact of learned components on automated decision-making systems and to guarantee that their consequences are beneficial to society. In other words, it is necessary to ensure that machine learning is sufficiently *trustworthy* to be used in real-world applications. This thesis studies two properties of machine learning models that are highly desirable for the sake of reliability: *robustness* and *fairness*.

In the first part of the thesis we study the *robustness* of learning algorithms to training data corruption. Previous work has shown that machine learning models are vulnerable to a range of training set issues, varying from label noise through systematic biases to worst-case data manipulations. This is an especially relevant problem from a present perspective, since modern machine learning methods are particularly data hungry and therefore practitioners often have to rely on data collected from various external sources, e.g. from the Internet, from app users or via crowdsourcing. Naturally, such sources vary greatly in the quality and reliability of the data they provide. With these considerations in mind, we study the problem of designing machine learning algorithms that are robust to corruptions in data coming from multiple sources. We show that, in contrast to the case of a single dataset with outliers, successful learning within this model is possible both theoretically and practically, even under worst-case data corruptions.

The second part of this thesis deals with *fairness-aware* machine learning. There are multiple areas where machine learning models have shown promising results, but where careful considerations are required, in order to avoid discrimanative decisions taken by such learned components. Ensuring fairness can be particularly challenging, because real-world training datasets are expected to contain various forms of historical bias that may affect the learning process. In this thesis we show that data corruption can indeed render the problem of achieving fairness impossible, by tightly characterizing the theoretical limits of fair learning under worst-case data manipulations. However, assuming access to clean data, we also show how fairness-aware learning can be made practical in contexts beyond binary classification, in particular in the challenging learning to rank setting.

Acknowledgements

First and foremost I would like to express my sincere gratitude to my supervisor Christoph Lampert, who has continuously supported me throughout my PhD journey. Under his supervision I had the privilege to be able to fully commit to curiosity-driven, independent research, with sufficient time and resources given to me to allow for deep and rigorous exploration of the topics that interested me most. Thank you, Christoph, for teaching me so much not only about machine learning, but also about research in general.

I would also like to thank all members of my thesis committee for their invaluable guidance throughout my PhD studies. In particular, special thanks goes to Dan Alistarh for all the support, exciting discussions and interesting work we did together, especially during the early phases of my PhD, and for teaching me a great deal about optimization and lower bounds. I would also like to thank my external member, Ingo Steinwart, for his very helpful feedback during the qualifying exam, the progress reviews and the defense. Finally, many thanks to Edouard Hannezo for kindly stepping in as an exam chair for both the qual and the defense.

I am very grateful to Nicolò Cesa-Bianchi and his group for warmly welcoming me at the University of Milan, during my ELLIS exchange, and for many exciting discussions we had about online learning and multi-armed bandits. I would also like to thank Krishnendu Chatterjee and Christian Hilbe for the interesting rotation project on evolutionary game theory, as well as the members of the ICC group at Amazon for welcoming me during my internship there.

Special thanks goes to Dino Sejdinovic, my master thesis supervisor, who first showed me the exciting side of science and ML theory and fully supported me in my intention to pursue a PhD; as well as to Veliko Kolev, my high school teacher, whose classes and methods taught me that in mathematics, and life, the only person that can solve one's problems is oneself.

An important part of my PhD have been the fun and inspiring times spent with the rest of the MLCV group members. Many thanks to Alex K., Alex P., Alex Z., Amelie, Bernd, Jonny, Mary, Paul and all our interns and rotation students! The same holds for all members of the ML community at IST, especially everyone who took part in the MLCV tea talks and reading group. Thank you all, I have learned a lot from you. Special thanks goes to my co-authors Elias and Eugenia - for the interesting research we did together and for sharing my excitement about lower bounds and discrepancies.

My PhD times would not have been as much fun without the friendly, open, dynamic and international environment at IST. Thank you to all my friends and colleagues for the great times and especially to my 2017 cohort-mates for the many enjoyable outings we had together. Special thanks goes to Bernd, Harald, Heloisa, Nika, Vlad and Wojciech for the many memorable experiences we have shared. A warm thank you is also due to my non-IST friends, who have (hopefully) helped me to preserve my sanity, especially to Alastair, Tsvetomir and Venci.

Last, but definitely not least, I would like to thank my family, and especially my parents and sister, whose love and endless support made this thesis possible.

Funding sources This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 665385.

The candidate has also received funding from the ELISE Mobility Program for PhD students and postdocs funded by the project European Learning and Intelligent Systems Excellence (ELISE, Grant Agreement No. 951847).

About the Author

Nikola Konstantinov graduated from the Sofia High School of Mathematics in 2013 and went on to complete a Master's degree in "Mathematics and Statistics" at the University of Oxford. His master thesis, titled "Kernel Dependence Measures for Unsupervised Learning", was conducted under the supervision of Prof. Dino Sejdinovic. In 2017 Nikola joined IST Austria as a PhD Candidate, doing rotations on multi-task learning (with Prof. Christoph H. Lampert), distributed optimization (with Prof. Dan Alistarh) and evolutionary game theory (with Prof. Krishnendu Chatterjee) and doing coursework in data science and computer science. Nikola then joined the group of Prof. Christoph H. Lampert, where he worked on topics related to robustness and fairness in machine learning. In 2019 he did an internship in the "Intelligent Cloud Control" group of Amazon.com, where he developed evaluation and model comparison methods for predictive models in cloud computing. In 2020 Nikola became an ELLIS PhD student and as part of this program he visited Prof. Nicolò Cesa-Bianchi at the University of Milan.

List of Collaborators and Publications

This thesis is written based on the following first-author publications and manuscripts of Nikola Konstantinov. For each, a summary of the contributions of every author is provided below.

- Nikola Konstantinov, Elias Frantar, Dan Alistarh, and Christoph H. Lampert. On the sample complexity of adversarial multi-source pac learning. In *International Conference on Machine Learing (ICML)*, 2020
  - This paper constitutes the core of Chapter 3.
  - Nikola Konstantinov formulated the problem and research objectives. He proved the upper bounds results, as well as Theorem 6. He also created an initial draft of the paper and was involved in the editing later on.
  - Elias Frantar was working on the experimental validation of the algorithm achieving the upper bounds and later on on the lower bounds. In particular, he formulated and proved an early version of Theorem 5. During this work, Elias Frantar was advised by Christoph H. Lampert and worked closely with Nikola Konstantinov.
  - Dan Alistarh was involved in the lower bounds work, in particular contributing with important discussions and references related to the relevant proof techniques. He also contributed to the comparison to prior work, in particular to the Byzantine learning literature.
  - Christoph H. Lampert advised Nikola Konstantinov and Elias Frantar throughout the project, providing important suggestions about related work, proof techniques, experimental design and writing. Christoph H. Lampert also participated extensively in the editing of the paper.
- Nikola Konstantinov and Christoph H. Lampert. Robust learning from untrusted sources. In *International Conference on Machine Learing (ICML)*, 2019
  - This paper constitutes the core of Chapter 4.
  - Nikola Konstantinov formulated the problem and research objectives, proved the theoretical results and conducted the experiments. He also created the initial draft of the paper and was involved in the editing later on.
  - Christoph H. Lampert advised Nikola Konstantinov throughout the project, with important suggestions about related work, proof techniques, experimental design and writing. Christoph H. Lampert also participated extensively in the editing of the paper.
- Nikola Konstantinov and Christoph H. Lampert. Fairness-aware learning from corrupted data. *arXiv preprint arXiv:2102.06004*, 2021

  - Note: A short version of this work was published and presented as a contributed talk at the NeurIPS 2021 workshop "Algorithmic Fairness through the Lens of Causality and Robustness".
  - This paper constitutes the core of Chapter 5.
  - Nikola Konstantinov formulated the problem and research objectives. He proved the theoretical results and was responsible for the paper writing.
  - Christoph H. Lampert advised Nikola Konstantinov throughout the project, with important suggestions on related work and proof techniques. Christoph H. Lampert also provided many helpful suggestions about improvements to the text.

- Nikola Konstantinov and Christoph H. Lampert. Fairness through regularization for learning to rank. *arXiv preprint arXiv:2102.05996*, 2021
  - This paper constitutes the core of Chapter 6.
  - Nikola Konstantinov and Christoph H. Lampert jointly came up with the problem formulation and research objectives.
  - Christoph H. Lampert formulated the type of theoretical guarantees needed and came up with the idea to use the chromatic concentration bounds for dealing with sample dependence on a per-query level. Nikola Konstantinov formulated and proved the corresponding theoretical results.
  - Nikola Konstantinov designed and conducted the experiments included in the paper, although Christoph H. Lampert was involved in the experiments in an earlier stage of the project.
  - Nikola Konstantinov and Christoph H. Lampert wrote and edited the paper.

During his employment as a PhD student at IST Austria, Nikola Konstantinov also co-authored the following papers and manuscripts (not included in the thesis):

- Dan Alistarh, Christopher De Sa, and Nikola Konstantinov. The convergence of stochastic gradient descent in asynchronous shared memory. In *ACM Symposium on Principles of Distributed Computing (PODC)*, 2018
- Dan Alistarh, Torsten Hoefler, Mikael Johansson, Sarit Khirirat, Nikola Konstantinov, and Cédric Renggli. The convergence of sparsified gradient methods. *Conference on Neural Information Processing Systems (NeurIPS)*, 2018
- Eugenia Iofinova, Nikola Konstantinov, and Christoph H. Lampert. Flea: Provably fair multisource learning from unreliable training data. *arXiv preprint arXiv:2106.11732*, 2021

Table of Contents

List of Figures

List of Tables

List of Algorithms

CHAPTER 1

Introduction

Machine learning algorithms have shown great potential in recent years by improving substantially upon the state of the art in multiple real-world applications of computer science, for example in image recognition [HZRS16], language understanding [DCLT19] and protein structure prediction [JEP+21]. Due to this outstanding performance of learning-based algorithms, there is an increasing amount of interest by practitioners in producing such models tailored to their purposes.

Because of the popularity of machine learning methods, it is becoming increasingly important to understand the impact of such learned components on automated decision-making systems and to guarantee that their consequences are beneficial to society. In other words, it is necessary to ensure that machine learning is sufficiently *trustworthy* to be used in real-world applications.

One of the key challenges towards building reliable machine learning models is that of ensuring *robustness against training data corruption*. Indeed, previous work has shown that modern machine learning models are not robust to problems in the train data. In particular, issues varying from label noise through model misspecifications to worst-case train data corruptions (a.k.a. data poisoning) can easily affect the performance of learned classifiers [Hub64, BNL12a, DKK+16, KL17], leading to poor model performance at test time. In addition, several classic hardness results [KL93, BEK02] state that there is no learning algorithm that can provably recover an optimal decision rule in the presence of data manipulations, unless restrictive assumptions about the type of corruptions are made.

These issues are especially relevant from a present perspective, since modern machine learning methods are particularly data hungry. Therefore, when training models practitioners often rely on data collected from various external sources, e.g. from the Internet, from app users or via crowdsourcing. Naturally, such sources vary greatly in the quality and reliability of the data they provide. Moreover, the structure of the resulting training dataset is different than what is assumed in classic robust machine learning theory. In contrast to a single block of i.i.d. data, with some potential outliers inside, a dataset obtained from multiple external sources consists of groups/batches, one from each source, and we expect some of these groups to contain clean data, while others - potentially corrupted. Due to these differences, classic techniques for defending machine learning against data corruptions may be suboptimal in such a multi-source scenario.

With these considerations in mind, the first part of this thesis deals with the problem of designing machine learning algorithms that are *robust* to corruptions in data coming from multiple sources. In the setup we study, some of the data sources may provide data that is noisy or systematically or even maliciously perturbed. We show that, in contrast to the single dataset case, successful learning within this model is possible both theoretically and practically. To this end, we describe the fundamental limits of learning against a malicious agent that controls some of the sources. We also design a practical algorithm for such grouped data and show that it exhibits strong performance in an extensive experimental evaluation.

Next, we move on to explore another property of machine learning models that is crucial for their real-world reliability, namely their *fairness*. Indeed, some of the areas where machine learning has shown promising results require careful considerations regarding the social impact that automated decisions can have [BHN19, MMS+19]. For example, whenever a learned system is used for screening the quality of job or loan applications, it is crucial to ensure that the model does not discriminate applicants based on their gender, race or other attributes protected by law or by ethical considerations. Similarly, if a machine learning model is used for suggesting personalized recommendations on online music platforms, it is important to ensure fair exposure for the performers.

There are multiple challenges related to achieving fairness in the context of machine learning. Firstly, an obstacle can again be the presence of data corruption, because real-world data is usually based on past human decisions, which are often prone to historical bias [MMS+19]. At the same time little is known about the effect of general data corruptions, such as label noise or adversarial manipulations, on fairness-aware learning. A second challenge is that the majority of fairness-aware learning algorithms and the corresponding theoretical results only consider the case of binary decision making. In contrast, machine learning is extensively used in much more complicated domains, such as recommended systems, in which, as already argued, fairness concerns can also arise.

Therefore, a second topic addressed in this thesis is that of machine learning fairness. In particular, we study the limits of fairness-aware learning under worst-case data corruption. We find that, alarmingly, guaranteeing fairness under such data issues is impossible and moreover the amount of unfairness that can not be avoided increases for problems where one of the protected subpopulations in the data is underrepresented (e.g. there are not many applicants of color for a job position). We also demonstrate a way of transfering classic fair learning techniques from binary classification to ranking, thereby providing theoretical guarantees for fair ranking and designing a scalable fairness-aware algorithm for this task.

The rest of the thesis is structured as follows:

- In Chapter 2 we introduce several notions that are central to machine learning theory and on which we heavily rely throughout the thesis. We also discuss various concepts from the theory of robust and fair machine learning, including such that will be recurrent throughout the text.

- Chapters 3 and 4 are dedicated to the topic of robust learning from multiple unreliable data sources. In Chapter 3 we study the theoretical limits of learning against an adversarial opponent that controls a subset of the data sources. In Chapter 4 we provide a practical algorithm for the studied problem, under the additional assumption that a small trusted reference dataset is given.

- In Chapter 5 we study the interplay between robustness and fairness and tightly characterize the limits of fairness-aware learning from corrupted data.
- Chapter 6 deals with fairness in ranking. We show how a number of popular fairness notions from binary classification can be integrated into modern learning-based ranking methods with little computational overhead and with theoretical guarantees. Furthermore, we show that improving fairness in the context of ranking often comes with little to no sacrifice of model utility.

CHAPTER 2

Preliminaries

In this chapter we introduce several machine learning concepts and notions that will be central to the work presented in this thesis. First, we give a short introduction to (supervised) machine learning. We place particular emphasis on the classic PAC learning framework, which provides a tool for studying the hardness of learning problems and analyzing the properties of learning algorithms. We also discuss several concepts from trustworthy machine learning research, with a focus on those notions from the robustness and fairness literature that will be central to our analysis later on.

2.1 Supervised machine learning

2.1.1 Elements of the supervised learning problem

In a nutshell, the goal of supervised machine learning is to use a dataset of input-label pairs to construct a general rule for assigning (predicting) labels to future inputs. The underlying hypothesis is that if such a labeling rule is constructed in a way that explains the train data, then this rule will also *generalize*, in the sense that it will be accurate at predicting the true labels for the future inputs as well.

To formalize this, we adopt a classic statistical learning framework, following e.g. [SSBD14, MRT18]. In this context, the supervised learning problem has the following ingredients.

An input space $\mathcal{X}$ The input space is the set of instances that we will be aiming to label. This can for example be the set of all possible emails or the set of all possible applicants for a job position. In the case of emails, one may be interested in constructing a classifier that decides if an email is spam or ham. In the case of job applicants, a useful classifier is one that is able to predict if an applicant will do well at the job they are applying for or not.

Commonly, the inputs (i.e. the elements of $\mathcal{X}$) are represented via elements of some metric space, for example $\mathbb{R}^d$. That is, instead of working directly with the text of the emails, or the CV of the applicants, *feature representations* for each input are given instead. There could be handcrafted features based on domain-specific considerations (for example, bag-of-words representations [MSR08] for the case of text), or could themselves be extracted via a machine learning model pretrained on massive amounts of data for a similar machine learning tasks (for example, using a state-of-the-art NLP model, such as BERT [DCLT19]). In such cases, we will

identify the inputs with their feature representations directly, since we assume that the feature extractors are given and fixed. Therefore, we will also identify $\mathcal{X}$ with the corresponding metric space of the features, e.g. $\mathbb{R}^d$.

A label space $\mathcal{Y}$ The label (or output) space is the set of all possible labels that can be assigned to an instance. A common example is the case of classification, where $Y = \{0, 1, \ldots, k-1\}$. Problems for which $k = 2$, such as those mentioned above, are called binary classification tasks and problems where $k > 2$ are known as multi-class classification tasks. In other cases labels can take continuous values, for example when $Y = \mathbb{R}$, a case that we refer to as regression.

A training dataset The training set refers to the provided example input-label pairs. We will often denote the training data by $S = \{(x_i, y_i)\}_{i=1}^n \in (\mathcal{X} \times \mathcal{Y})^n$. This is the set of examples that should be used for constructing a labeling function. An important parameter is the number of samples, denoted here by n. Intuitively, the larger the value of n, the more information we have about the prediction task at hand. In other words, one would expect that the learning problem gets easier as access to more data is given.

A hypothesis space A hypothesis space is a set of possible labeling rules to choose from. Formally, we assume that a hypothesis space $\mathcal{H} \subset \mathcal{Y}^{\mathcal{X}}$ is given. Each element h of $\mathcal{H}$ is a function $h : \mathcal{X} \to \mathcal{Y}$ from the input space to the label space and represents a rule for labeling inputs. For example, if $\mathcal{X} = \mathbb{R}^d$ for some $d \in \mathbb{N}^+$ and $\mathcal{Y} = \{0, 1\}$, then a commonly used hypothesis space is the one of all linear classifiers on $\mathbb{R}^d$, namely $\mathcal{H} = \left\{h : h(x) = \textit{sign}\left(w^T x\right) \ \textit{for some } w \in \mathbb{R}^d\right\}$.

A learner A learner is a procedure that takes the set S of labeled examples and outputs a hypothesis from $\mathcal{H}$ that, ideally, accurately predicts labels given inputs. Formally, a learner is a function:

$$\mathcal{L} : \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{H}. \tag{2.1}$$

Note that here we think of the learner as *any procedure* that takes a dataset of arbitrary size and returns a hypothesis, thereby focusing on the statistical, rather than the computational, questions regarding learning.

A loss function The loss function measures the penalty incurred when assigning a wrong label to an instance. Intuitively, the goal of learning is to find a hypothesis that accurately predicts labels given inputs. For any hypothesis $h \in \mathcal{H}$ one can measure the quality of its prediction on a input-label pair (x, y) via *a loss function* $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$, by computing $\ell(h(x), y)$. A natural requirement is that $\ell(y, y) = 0$ and $\ell(y_1, y_2) > 0$ for any $y, y_1, y_2 \in \mathcal{Y}$ with $y_1 \neq y_2$.

Commonly used loss functions are the $0-1$ loss for classification, that is $\ell(y_1, y_2) = \mathbb{1}\{y_1 \neq y_2\}$; as well as the square loss for regression, i.e. $\ell(y_1, y_2) = (y_1 - y_2)^2$.

Although the loss function gives a way of measuring the performance of a hypothesis on one input-label point, it is unclear how to formalize the objective of finding a hypothesis that works well on multiple future inputs. Moreover, in order to have any hope of constructing such a hypothesis based on the training data, one needs to ensure that the points in S are in some sense representative of the future input-output pairs that the learner will be tested on.

Next, we present a standard way to formulate these issues in statistical terms, by adopting a standard PAC learning framework.

2.1.2 Statistical PAC learning

The objective of learning Since the learner only operates with a set of training examples, an underlying assumption for successful learning is that the learner will be able to *generalize* their knowledge based on these examples to new, unseen inputs and so to perform well at prediction time. Intuitively, this should be the case whenever the new inputs are *similar* to those that were observed by the learner at train time. To formalize this, a standard assumption is that both the training and the test examples are sampled from a distribution $\mathcal{D} \in \mathcal{P}(\mathcal{X} \times \mathcal{Y})$ in an independent and identically distributed (i.i.d.) manner.

This statistical treatment allows for a formalization of the learning objective. The intuitive goal of providing a hypothesis that works well on new inputs translates to the goal of finding a hypothesis with a small *expected loss* under the distribution $\mathcal{D}$, that is finding an $h \in \mathcal{H}$ such that:

$$\mathcal{R}(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left(\ell(h(x), y)\right) \tag{2.2}$$

is small. In the case when $\mathcal{Y} = \{0, 1\}$ and ℓ is the $0-1$ loss, $\mathcal{R}(h)$ is also called the *risk* of h.

PAC learnability Given the training dataset $S = \{(x_i, y_i)\}_{i=1}^n \in (\mathcal{X} \times \mathcal{Y})^n$, the learner predicts $\mathcal{L}(S) \in \mathcal{H}$. Then the performance of the learner can be measured via the expected loss of the hypothesis that it returns, that is via $\mathcal{R}(\mathcal{L}(S))$. While one may expect a perfect learner to always return a hypothesis with expected loss equal to 0, there are several limitation that may in general make this impossible:

- For many distributions there will be no hypothesis in $\mathcal{H}$ that achieves risk of 0. In such situations, the performance of the learner, as measured by $\mathcal{R}(\mathcal{L}(S))$, should be compared to the best possible hypothesis in the hypothesis space, that is to $\inf_{h\in\mathcal{H}} \mathcal{R}(h)$, rather than to 0. In this sense, a good learner learner has to be *agnostic* to any assumptions about the input-label distribution and work as well as possible for any distribution instead.
- Since the learner works with a finite set of samples, there is always a certain probability that the n samples in S will end up being unrepresentative of the underlying distribution $\mathcal{D}$. For example, if $\mathcal{X}$ is discrete, there is a non-zero probability that all inputs will end up being the same, even if $\mathcal{D}$ assigns a significant mass on the other points in $\mathcal{X}$ as well. Such situations, although increasingly unlikely for large values of n, imply that in general the learner can only be expected to *probably* (usually) work.
- Even when the learner is presented with a useful set of samples S, the sample size is still finite and any estimate of the underlying distribution $\mathcal{D}$ will in general only be an approximation. Therefore, we can only expect that the learner works *approximately* as well as the best hypotheses in $\mathcal{H}$.

These considerations naturally lead to the concept of *agnostic probably approximately correct learning (agnostic PAC learning)*, as introduced by Leslie Valiant [Val84]. Intuitively, a learner will be successful if, given a sufficiently large training set from a distribution, they manage to output a hypothesis that is approximately as good as the best ones in the hypothesis class,

and do so with high probability with respect to the sampling of the data. If such a learner exists, then a hypothesis space is in a sense "learnable" - there is a learner that performs well on it given enough data. This is formalized in the following definition.

Definition 1 (Agnostic PAC Learnability)**.** *A hypothesis space $\mathcal{H}$ is agnostic PAC learnable with respect to a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathcal{R}^+$, if there exists a learner $\mathcal{L} : \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{H}$ and a function $m_{\mathcal{H}} : (0,1) \times (0,1) \to \mathbb{N}$, such that for any $\epsilon, \delta \in (0,1)$ and any distribution $\mathcal{D}$, if the learner takes as input a set S of at least $m_{\mathcal{H}}(\epsilon, \delta)$ points sampled i.i.d. from $\mathcal{D}$, then with probability at least $1 - \delta$ with respect to the sampling of the points in S:*

$$\mathcal{R}(\mathcal{L}(S)) \leq \inf_{h \in \mathcal{H}} \mathcal{R}(h) + \epsilon. \tag{2.3}$$

Whenever $\mathcal{H}$ is agnostic PAC learnable, the (point-wise) smallest possible function $m_{\mathcal{H}}$ determines the so-called *sample complexity* of $\mathcal{H}$, that is, how many samples are indeed so that an ϵ-good hypothesis can be recovered with probability at least $1 - \delta$, regardless of the underlying probability distribution.

Realizable PAC learning An important special case of the PAC learning problem is the one where the label space is binary and the distribution $\mathcal{D}$ under consideration is such that there exists a perfectly accurate hypothesis in $\mathcal{H}$, that is a hypothesis $h^* \in \mathcal{H}$, such that $\mathbb{P}_{(x,y)\sim\mathcal{D}}(h^*(x) = y) = 1$. Note that this means in particular that the labels are deterministic given the inputs (up to a set of measure 0). This scenario is referred to in the literature as the realizable PAC learning scenario. In such a context we have the following definition of PAC learnability (without the term "agnostic" included, since we are now making a specific assumption about the type of distributions $\mathcal{D}$ that are allowed).

Definition 2 (PAC Learnability (realizable case))**.** *A hypothesis space $\mathcal{H}$ is PAC learnable with respect to the $0-1$ loss function, if there exists a learner $\mathcal{L} : \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{H}$ and a function $m_{\mathcal{H}} : (0,1) \times (0,1) \to \mathbb{N}$, such that for any $\epsilon, \delta \in (0,1)$ and any distribution $\mathcal{D}$ for which the realizability assumption holds with respect to $\mathcal{H}$, if the learner takes as input a set S of at least $m_{\mathcal{H}}(\epsilon, \delta)$ points sampled i.i.d. from $\mathcal{D}$, then with probability at least $1 - \delta$ with respect to the sampling of the points in S:*

$$\mathbb{P}_{(X,Y)\sim\mathcal{D}}(\mathcal{L}(S)(X) \neq Y) \leq \epsilon. \tag{2.4}$$

Uniform convergence Since both the training and test samples are assumed to be drawn from $\mathcal{D}$ in an i.i.d. manner, one may hope that the performance of a hypothesis at prediction time can be estimated by looking at its performance on the data points in S. Therefore, it is natural to consider the empirical loss of any $h \in \mathcal{H}$, given by

$$\widehat{\mathcal{R}}(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(h(x_i), y_i). \tag{2.5}$$

This concept leads to a natural learning algorithm that simply selects a hypothesis by minimizing the empirical risk, giving rise to the empirical risk minimization principle (ERM principle). Formally, the ERM learner $\mathcal{L}_{ERM} : \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{H}$ is defined as

$$\mathcal{L}_{ERM}(S) = \operatorname*{argmin}_{h \in \mathcal{H}} \widehat{\mathcal{R}}(h) \quad \forall S \in \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n . \tag{2.6}$$

One may hope that if $\widehat{\mathcal{R}}(\mathcal{L}_{ERM}(S))$ is small, then this guarantees that the same is true for $\mathcal{R}(\mathcal{L}_{ERM}(S))$ and therefore that the ERM rule necessarily recovers a good hypothesis.

However, this does not follow directly from the law of large numbers, since $\mathcal{L}_{ERM}(S)$ depends on the data points in S and therefore standard concentration arguments are insufficient to show that the empirical and the true risk of this hypothesis are close to each other. Instead, in order to show this one needs to require that the empirical and the true risk are close to each other *for all hypothesis* in $\mathcal{H}$, that is, that concentration holds uniformly over the hypothesis space. This naturally needs to the concept of uniform convergence, which is presented in the following definition.

Definition 3 (Uniform Convergence)**.** *A hypothesis space $\mathcal{H}$ is uniformly convergent with respect to a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathcal{R}^+$, if there exists a function $m_{\mathcal{H}}^{UC} : (0,1) \times (0,1) \to \mathbb{N}$, such that for any $\epsilon, \delta \in (0,1)$ and any distribution $\mathcal{D}$, if the learner takes as input a set S of at least $m_{\mathcal{H}}^{UC}(\epsilon, \delta)$ points sampled i.i.d. from $\mathcal{D}$, then with probability at least $1 - \delta$ with respect to the sampling of the points in S:*

$$\sup_{h \in \mathcal{H}} |\mathcal{R}(h) - \widehat{\mathcal{R}}(h)| \leq \epsilon. \tag{2.7}$$

Whenever $\mathcal{H}$ is uniformly convergent, we refer to the component-wise smallest possible function $m_{\mathcal{H}}^{UC}$ as the rate of uniform convergence of $\mathcal{H}$.

It is easy to see that any uniformly convergent hypothesis space is also agnostic PAC learnable.

Proposition 1. *Whenever $\mathcal{H}$ is uniformly convergent with rate $m_{\mathcal{H}}^{UC}$, $\mathcal{H}$ is also agnostic PAC learnable with sample complexity at most $m_{\mathcal{H}}^{UC}\left(\frac{\epsilon}{2}, \delta\right)$.*

Proof. Fix any $\epsilon, \delta \in (0,1)$ and let $\mathcal{D}$ be an arbitrary distribution on $\mathcal{X} \times \mathcal{Y}$. For any dataset S of size at least $m_{\mathcal{H}}^{UC}\left(\frac{\epsilon}{2}, \delta\right)$, sampled i.i.d. from $\mathcal{D}$, with probability at least $1 - \delta$ we have that

$$\sup_{h \in \mathcal{H}} |\mathcal{R}(h) - \widehat{\mathcal{R}}(h)| \leq \frac{\epsilon}{2}$$

and so

$$\mathcal{R}(\mathcal{L}_{ERM}(S)) \leq \widehat{\mathcal{R}}(\mathcal{L}_{ERM}(S)) + \frac{\epsilon}{2} = \min_{h \in \mathcal{H}} \widehat{\mathcal{R}}(h) + \frac{\epsilon}{2} \leq \inf_{h \in \mathcal{H}} \mathcal{R}(h) + \epsilon$$

Therefore the ERM learner is an agnostic PAC learner for $\mathcal{H}$, with sample complexity $m_{\mathcal{H}}(\epsilon, \delta) \leq m_{\mathcal{H}}^{UC}\left(\frac{\epsilon}{2}, \delta\right)$. □

In particular, this result suggests that the ERM rule is a reasonable approach to learning and that, if the hypothesis space is uniformly convergent, ERM is in fact an agnostic PAC learner.

2.1.3 Measures of complexity

The bias-variance trade-off From a more practical perspective, one may want to decide what type of a hypothesis space should be chosen given a particular supervised learning problem, that is, for a fixed but unknown distribution and given some amount of data.

In the context of learning with the ERM rule, an intuitive trade-off emerges. On the one hand, selecting a hypothesis space of large size will increase the likelihood of finding a hypothesis that performs well on the training data set and so the ERM rule will yield a hypothesis with small empirical loss. On the other hand, if the hypothesis space is too large, the uniform

convergence property is more likely to be violated or at least occur at a slower rate. Therefore, because of the finite sample size, the ERM hypothesis might not generalize and perform poorly at test time.

It turns out that this reasoning applies to learning more generally, resulting in the so-called *bias-variance trade-off*. For hypothesis spaces that are too "simple", there might be no hypothesis that performs well on a given distribution, resulting in a large, irreducible *bias term*. At the same time, whenever a hypothesis space is too "rich", there might be multiple hypothesis that appear as good candidates based on the train data. However, the limited sample size and the large number of potential candidates result in a corresponding *variance term* that describes the hardness of accurately estimating the properties of the true data distribution given the training set only. In-between there is a sweet spot, where a hypothesis space whose "complexity" is the right one for the task at hand hits the best possible trade-off between bias and variance. Note that this reasoning can be seen as an instance of the Occam razor: out of various possible statistical models for a given task (that is, various choices of $\mathcal{H}$), one should select the one that works (has small bias) and is as simple as possible (has small variance).

These considerations emphasize on the importance of studying various notions of the complexity (i.e. richness) of hypothesis spaces in the context of learning. One may hope that an appropriate notion of complexity can then inform machine learning practitioners on how to select a hypothesis space given a particular learning task, through the lens of the bias-variance trade-off. Moreover, it is intuitive that good notions of complexity may provide sufficient and necessary conditions for PAC learnability, as hypothesis spaces that are "too rich" may induce learning problems that are impossible to solve even with infinite data.

We now discuss two popular complexity measures, the VC dimension and the Rademacher complexity, which are central concepts in learning theory and which allow for quantifying the bias-variance trade-off and proving PAC-style guarantees for learning algorithms.

VC dimension Consider the context of binary classification with the $0-1$ loss. Given n points $x_1, \dots, x_n \in \mathcal{X}$, denote

$$\mathcal{H}_{x_1,\dots,x_n} \coloneqq \{(h(x_1), \dots, h(x_n)) : h \in \mathcal{H}\}.$$

Note that $|\mathcal{H}_{x_1,\dots,x_n}|$ is the number of possible label assignments that the hypotheses in $\mathcal{H}$ can give to the n data points. Clearly $|\mathcal{H}_{x_1,\dots,x_n}| \leq 2^n$ and if $|\mathcal{H}_{x_1,\dots,x_n}| = 2^n$ we will say that $\mathcal{H}$ *shatters* the set of points $\{x_1, \dots, x_n\}$. Next define the growth function of $\mathcal{H}$ as

$$S_{\mathcal{H}}(n) = \sup_{x_1,\dots,x_n} |\mathcal{H}_{x_1,\dots,x_n}|.$$

Again, $S_{\mathcal{H}}(n) \leq 2^n$ and if $S_{\mathcal{H}}(n) = 2^n$ then there exists a set of n input points that $\mathcal{H}$ can shatter. Intuitively, if $\mathcal{H}$ can shatter a large number of input points, then $\mathcal{H}$ constitutes a very rich set of models. This motivates the following definition of what is known as the VC dimension of a binary hypothesis space $\mathcal{H}$.

Definition 4. *The VC dimension of a hypothesis space $\mathcal{H} \subset \{0,1\}^{\mathcal{X}}$ is the maximum number of points that $\mathcal{H}$ can shatter, that is the largest $n \in \mathbb{N}$, such that $S_{\mathcal{H}}(n) = 2^n$. If no such n exists, then we say that $\mathcal{H}$ is of infinite VC dimension.*

It turns out that the VC dimension being finite is a necessary and sufficient condition for agnostic PAC learnability.

Theorem 1 (The Fundamental Theorem of Statistical Learning, [SSBD14]). *Let $\mathcal{H} \subset \{0,1\}^{\mathcal{H}}$ and let the loss under consideration be the $0-1$ loss. Then the following statements are equivalent:*

- *$\mathcal{H}$ is agnostic PAC learnable.*
- *$\mathcal{H}$ is PAC learnable.*
- *$\mathcal{H}$ is uniformly convergent.*
- *$\mathcal{H}$ has a finite VC dimension.*
- *ERM is a successful (agnostic) PAC learner for $\mathcal{H}$.*

Furthermore, one can quantitatively describe the speed at which the uniform convergence property holds for $\mathcal{H}$, via the following result:

Theorem 2 (p. 342 in [SSBD14]). *Let $\mathcal{D} \in \mathcal{P}(\mathcal{X} \times \mathcal{Y})$ be a fixed distribution and suppose that $\mathcal{H}$ is of finite VC dimension $d = VC(\mathcal{H})$. Then for any $\delta \in (0,1)$*

$$\mathbb{P}\left(\sup_{h\in\mathcal{H}} |\widehat{\mathcal{R}}(h) - \mathcal{R}(h)| > 2\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(4/\delta)}{n}}\right) \leq \delta. \tag{2.8}$$

This result, together with the link between uniform convergence and agnostic PAC learning described in Section 2.1.2, can be used to show that a hypothesis space with VC dimension d is agnostic PAC learnable with sample complexity:

$$m_{\mathcal{H}}(\epsilon,\delta) = \widetilde{\Omega}\left(\frac{d + \log(1/\delta)}{\epsilon^2}\right). \tag{2.9}$$

As an alternative formulation, Theorem 2 implies that given a training set of size n, one can guarantee that the distance between the empirical and the true risk of a hypothesis is at most of $\widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}}\right)$.

In the realizable PAC learning case one can show that the sample complexity of a hypothesis space with a finite VC dimension is actually smaller that in the agnostic case. Specifically, one can show that [SSBD14]

$$m_{\mathcal{H}}(\epsilon,\delta) \leq \mathcal{O}\left(\frac{d\log(1/\epsilon) + \log(1/\delta)}{\epsilon}\right). \tag{2.10}$$

In particular, only $\mathcal{O}\left(\frac{1}{\epsilon}\right)$ samples are needed to find an ϵ-optimal solution, as compared to $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$ in the general case. It is therefore standard to refer to the $\mathcal{O}\left(\frac{1}{\epsilon}\right)$ rates achievable in the realizable PAC learning scenario as "fast statistical rates".

Rademacher complexity In the case of a non-binary label space and a general loss function a more sophisticated complexity measure, namely the Rademacher complexity, can be used to describe the sample complexity and uniform convergence rates of a hypothesis space.

Definition 5. *Let $\mathcal{H}$ be a hypothesis space and ℓ be a loss function. Let $S = \{(x_i, y_i)\}_{i=1}^n$ be a set of n i.i.d. data points from a distribution $\mathcal{D}$. Let $\sigma = (\sigma_1, \dots \sigma_n)$ be a set of n i.i.d. Rademacher random variables, that is, random variables uniformly distributed on $\{-1, 1\}$. Then the empirical Rademacher complexity of $\mathcal{H}$ with respect to ℓ and the sample S is defined as*

$$\widehat{\mathfrak{R}}_S(\ell \circ \mathcal{H}) = \mathbb{E}_\sigma \left(\sup_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^n \sigma_i \ell(h(x_i), y_i) \right). \tag{2.11}$$

The (distributional) Rademacher complexity of $\mathcal{H}$ with respect to ℓ and the distribution $\mathcal{D}$ is defined as

$$\mathfrak{R}_n(\ell \circ \mathcal{H}) = \mathbb{E}_{S \sim \mathcal{D}^n} \left(\mathbb{E}_\sigma \left(\sup_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^n \sigma_i \ell(h(x_i), y_i) \right) \right) = \mathbb{E}_{S \sim \mathcal{D}^n} \left(\widehat{\mathfrak{R}}_S(\ell \circ \mathcal{H}) \right). \tag{2.12}$$

Intuitively, the Rademacher complexity captures the degree to which the hypothesis space can capture random noise [MRT18] and is therefore a useful measures of the expressiveness of $\mathcal{H}$. As a result, one can also use the Rademacher complexity to bound the gap between the test time and the train time loss of a classifier, uniformly over the hypothesis space.

Theorem 3 (Theorem 3.3 in [MRT18])**.** *Let $\mathcal{H}$ be a hypothesis space and ℓ be a loss function such that $\ell(y_1, y_2) \leq M$ for some $M > 0$ and for all $y_1, y_2 \in \mathcal{Y}$. Then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the i.i.d. sampling of a training set $S \sim \mathcal{D}^n$ from a distribution $\mathcal{D}$, each of the following holds uniformly over all $h \in \mathcal{H}$:*

$$\mathcal{R}(h) \leq \widehat{\mathcal{R}}(h) + 2\mathfrak{R}_n(\ell \circ \mathcal{H}) + M\sqrt{\frac{\log(1/\delta)}{2n}} \tag{2.13}$$

and

$$\mathcal{R}(h) \leq \widehat{\mathcal{R}}(h) + 2\widehat{\mathfrak{R}}_S(\ell \circ \mathcal{H}) + 3M\sqrt{\frac{\log(2/\delta)}{2n}}. \tag{2.14}$$

In the special case of binary classification, the Rademacher complexity can be upper-bounded using the VC dimension $d = VC(\mathcal{H})$. The Rademacher complexity can first be bounded in terms of the growth function of $\mathcal{H}$ as follows:

$$\mathfrak{R}_n(\ell \circ \mathcal{H}) \leq \sqrt{\frac{\log(S_{\mathcal{H}}(n))}{2n}}. \tag{2.15}$$

Next, Sauer's lemma can be used to show that whenever $n \geq d$,

$$S_{\mathcal{H}}(n) \leq \left(\frac{en}{d}\right)^d.$$

Therefore, for any $n \geq d$,

$$\mathfrak{R}_n(\ell \circ \mathcal{H}) \leq \sqrt{\frac{d \log(en/d)}{2n}} \leq \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}}\right). \tag{2.16}$$

Note that this upper bound can be used together with Theorem 3 to show a bound on the excess risk with the same asymptotic behavior as the one in Theorem 2. In particular, the bound based on the VC dimension is looser. This is essentially because the VC dimension is a *distribution-agnostic* measure of complexity and generalization arguments that are based on it are necessarily worst-case over the properties of $\mathcal{D}$. In contrast, the Rademacher complexity is a tool for providing distribution or data-dependent guarantees for learning, which can be tighter for certain "easier" learning problems.

2.2 Learning from corrupted data

As discussed in the previous section, one of the central assumptions within the framework of PAC learning is that the training set S is sampled i.i.d. from the target data distribution $\mathcal{D}$. While being natural and convenient from a theory perspective, this assumption hardly ever holds in practice. There are various real-world data problems that can lead to a deviation from the i.i.d. model, for example dependence between the samples or underrepresentation of the certain subpopulations. In this thesis we focus on another issue that is of high relevance in practice, namely that of data corruption, that is, the presence of noisy, biased or even purposely manipulated entries within a subset of the training data.

In the presence of data corruption, a cause of concern is that the machine learning guarantees that are traditionally developed under the idealistic i.i.d. assumption may not apply anymore because this assumption is not fulfilled. Unfortunately, empirical observations have shown that such deviations from the classic theory can indeed lead to poor performance of learned models [BNL12a, CLL+17, SNT+20]. In other words, models trained via classic machine learning techniques are often not *robust* to the data corruption. This naturally hinders the applicability of such systems to real-world tasks.

A major focus of this thesis is on overcoming the aforementioned issues by developing machine learning algorithms that offer provable performance guarantees and strong empirical performance, even when a fraction of the training data is corrupted. In this section, we first motivate this problem and give specific examples of the types of data corruptions that can be expected in machine learning datasets. Then we introduce several classic models that allow us to reason about learning from a potentially corrupted dataset.

We note that learning from corrupted training data is a field with long history, where both theoretical and experimental issues have been widely studied, e.g. [Tuk60, Hub64, AL88, KL93, CBDF+99, BEK02, CS04, BNL12a, CSV17, SKL17, CLL+17, DKK+19b]. A complete overview of this field is therefore unavoidably beyond the scope of this thesis. Instead, we focus on concepts related to robustness in the context of PAC learning.

2.2.1 Motivation

There are various types of data corruption that are known to occur in real-world training data. Here we present a few motivating examples.

Label noise One of the most commonly cited problems with real-world data is that of label noise [FV13]. This issue is particularly prevalent in recent years, since it is becoming increasingly common to label datasets via crowdsourcing platforms, such as Amazon Turk, and the quality of the obtained labels is often low.

Measurement errors Various types of random or systematic errors can also appear on the level of the input variables [BEK02]. For example, in many scientific fields data is often obtained via taking measurements by using sophisticated devices. Since these devices are often used as black boxes, a practitioner may not notice certain malfunctions that result in random or systematic errors for some data entries.

Historical bias In many cases when machine learning models are trained as part of a real-world decision making system, one may expect training data issues related to bias stemming

from past human behavior [MMS+19]. For example, when creating models for evaluating job applications, the training data used will likely be based on previous rounds of applications, where human decision makers may not have been completely objective in their evaluations.

Adversarial manipulations of the data In some cases one may expect worst-case manipulations of the data that specifically target the model's accuracy or other desirable properties, such as the model's fairness. For example, in 2016 Microsoft released its own Twitter bot, called Tay, that was meant to learn from real-world Twitter data in an online manner [Hun16]. This resulted in an immediate surge of tweets containing inappropriate content, published by users seeking to sabotage the performance of the bot. Consequently Tay was taken down by its creators soon after it started mimicking the malicious data content that it was trained on. Additionally, multiple studies from the academic community, e.g. [BNL12b, CLL+17], have shown that even a small fraction of adversarially perturbed training data points can significantly impact the performance of the resulting model.

These examples suggest that the presence of data corruption is a common phenomenon in real-world data. Moreover, these and others data quality issues can have a direct negative impact on model performance. It is therefore important to study various models that go beyond the classic PAC learning setup by allowing for a certain fraction of the data to deviate from the clean distribution $\mathcal{D}$. The underlying intention is that such models can inform machine learning researchers about ways of designing learning algorithms that are robust to data corruption.

2.2.2 Learning against an adversary

Models of data corruption A common approach towards addressing the issue of poor data quality is to design learning models under specific assumptions about the types of corruptions that are present. For example, consider the case of creating a supervised learning dataset $S = \{(x_i, y_i)\}_{i=1}^n$ based on crowdsourcing, where the inputs are collected centrally and the crowdsourcing workers are only asked to provide labels. In such a scenario it is natural to assume that the input variables x_i are clean and indeed sampled i.i.d. from the marginal distribution of the inputs. The corruption model is then restricted to label manipulations. If it is reasonable to assume that labeling mistakes happen randomly and independently of the input values, then an appropriate model is that of label noise. In such a scenario, each data point (x_i, y_i) is assumed to be generated as follows. First, a clean data point (x_i^c, y_i^c) is drawn. Then with some (small) probability α, the label y_i^c is flipped and hence the final data point is $(x_i, y_i) = (x_i^c, 1 - y_i^c)$. Otherwise, with probability $1 - \alpha$, the returned point is $(x_i, y_i) = (x_i^c, y_i^c)$.

There are two main disadvantages of such application-specific data corruption models. Firstly, the types of corruptions present in the data vary greatly between different applications and therefore separate models might have to be developed and studied for different scenarios. Secondly, even within a single application it is often hard to foresee all possible types of data problems that might be present. Therefore, using a corruption model that is too specific may lead to flawed guarantees and therefore unexpected performance of the learned model at prediction time. This is undesirable, especially in security-critical applications of machine learning, such as autonomous driving and fraud detection. Moreover, the inability to provide worst-case guarantees for machine learning models may in the long run hinder their adoption for real-world decision-making tasks.

The adversary Due to the aforementioned issues, in this thesis we will mostly adopt an orthogonal, fully worst-case approach to modeling data corruptions. Following an established framework from the fields of cybersecurity and the software verification, we will assume the presence of a malicious opponent, called *the adversary* [BEY05], that may insert corrupted data points that can be generated with infinite computational resources and can be chosen with a broad knowledge of the underlying setup, the clean data points and even the learning algorithm. While in some cases such a worst-case approach may appear overly pessimistic, our treatment has the advantage of providing a "certificate" for the model performance: whenever our theoretical analysis provides guarantees against the adversary, these guarantees hold under a very broad range of possible data problems and therefore cover a rich set of applications and secure the model against multiple, known or unknown, data issues.

Formally, the adversary is simply a procedure that takes in a clean dataset and output a new, corrupted set of points of the same size as the original training set. That is, the adversary is a potentially randomized function $\mathfrak{A} : \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n \to \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n$, with the only constraint that $|\mathfrak{A}(S)| = |S|$ for any $S \subset (\mathcal{X} \times \mathcal{Y})$. Typically, we will assume that the adversary is subject to certain limitations, for example, only being able to affect a certain fraction of the data. We will denote the set of all possible adversaries, that is, all functions that satisfy the limitations, by $\mathfrak{A}$. Depending on the type of restrictions that are imposed on $\mathfrak{A}$, various adversarial models are obtained. We will present several examples of popular adversarial models in Section 2.2.3.

One common limitation enforced on the adversary that will be recurring throughout the thesis is that only a certain fraction of the data, say $\alpha \in [0, 1]$, can be manipulated. That is, the adversary can only manipulate up to αn points, with this upper bound being either approximate (e.g. $Bin(n, \alpha)$ points can be changed) or exact. In the fields of robust statistics and machine learning, it is standard to think of α as a small constant and in particular to consider $\alpha < 0.5$. Indeed, in the case when $\alpha \geq 0.5$, robust estimation has only been shown to be possible under additional assumptions, for example the possibility of returning multiple candidate estimates or the availability of a small subset of trusted, clean data [CSV17].

We will consider α as a crucial parameter that describes the power of the adversary and often refer to an adversary that can corrupt (approximately) αn points as an adversary of power α.

PAC learning against an adversary We now formalize the learner's objective in the presence of an adversary. Intuitively, given a supervised learning problem with a hypothesis space $\mathcal{H}$ and a loss function ℓ, a learner $\mathcal{L} : \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{H}$ will be successful against the set of adversaries $\mathfrak{A}$ if it is able to guarantee PAC learnability based on the corrupted data against any adversary in $\mathfrak{A} \in \mathfrak{A}$ and for any clean data distribution $\mathcal{D}$. Here we formalize this as follows:

Definition 6. *A hypothesis space $\mathcal{H}$ is adversarially agnostic PAC learnable with respect to a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathcal{R}^+$ and against the set of adversaries $\mathfrak{A}$, if there exists a learner $\mathcal{L} : \cup_{n=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{H}$ and a function $m_{\mathcal{H}} : (0, 1) \times (0, 1) \to \mathbb{N}$, such that for any $\epsilon, \delta \in (0, 1)$, any distribution $\mathcal{D}$ and any adversary $\mathfrak{A} \in \mathfrak{A}$, whenever S^c is an clean i.i.d. dataset of at least $m_{\mathcal{H}}(\epsilon, \delta)$ points sampled from $\mathcal{D}$ and $S^p = \mathfrak{A}(S^c)$, if the learner takes as input the set S^p , then with probability at least $1 - \delta$ with respect to the sampling of the points in S^c and the randomness of the adversary:*

$$\mathcal{R}(\mathcal{L}(S^p)) = \mathcal{R}(\mathcal{L}(\mathfrak{A}(S^c))) \leq \inf_{h \in \mathcal{H}} \mathcal{R}(h) + \epsilon. \tag{2.17}$$

Intuitively, it is assumed that the adversary has access to an initial clean training dataset S^c and manipulates it into a poisoned (corrupted) dataset S^p. The learner then works with the dataset S^p and hence only has access to the corrupted data.

Crucial in this definition is the ordering of the quantifiers. As discussed above, our approach to the problem of learning from corrupted data is a worst-case one. Therefore, we are assuming that the adversary acts with a full knowledge of the setup, including the hypothesis space and loss function, but also the clean distribution, the clean training data and even the learner itself. In contrast, the learner is assumed to have access only to the hypothesis space, the loss function and the corrupted data.

This is indeed reflected in the formulation of Definition 6, since the adversary is chosen after the learner. In particular, for a learner $\mathcal{L}$ to achieve PAC learnability, $\mathcal{L}$ needs to "work" in the usual PAC sense under any distribution-adversary pair. Since the adversary is chosen after $\mathcal{L}$ is fixed, this is a way to formalize the claim that the adversary "knows" the learner.

This reasoning is also reflected in the results in the following sections of the thesis. Whenever we state positive results that certify the existence of a learner achieving a certain performance, these results will be structured as follows:

There exists a learner $\mathcal{L}$, such that for any distribution $\mathcal{D}$, any adversary $\mathfrak{A} \in \mathfrak{A}$ and any $\delta \in (0,1)$, with probability at least $1-\delta$ ...

Since the learner is fixed before the distribution and the adversary are, it has to work for any such pair. In contrast, hardness results that show that the adversary can prevent the learner from finding models with certain properties will have the form:

For any learner $\mathcal{L}$ there exists a distribution $\mathcal{D}$ and an adversary $\mathfrak{A} \in \mathfrak{A}$, such that with constant probability ...

Note in particular that the adversary can be chosen after the learner is constructed and together with the distribution and it can therefore be tailored to their choice.

2.2.3 Adversarial models

We now discuss two established worst-case models of data corruption, namely the malicious adversary model [Val85] and the nasty adversary model [BEK02]. Throughout the thesis we will be considering adversaries that are inspired by these classic models and adapted to the corresponding contexts that we study.

For both models, we assume that a clean dataset $S^c \sim \mathcal{D}^n$ is sampled i.i.d. from $\mathcal{D}$ and we describe the (randomized) procedure for computing a corresponding corrupted dataset S^p.

The malicious adversary model The malicious adversary model was first introduced by [Val85] and extensively studied by [KL93] and [CBDF+99]. Informally, given a clean dataset, the malicious adversary has the power to *arbitrarily manipulate* a randomly chosen subset of the data of size $Bin(n, \alpha)$, for some corruption ratio $\alpha \in [0,1]$. Within this random subset, the adversary can substitute each data point with an arbitrary input-label pair, chosen with knowledge of the remaining data, the clean distribution and the learning algorithm. Outside of this set, the points have to remain the same.

Formally, the process of computing the corrupted dataset $S^p = \{(x_i^p, y_i^p)\}_{i=1}^n$ from the clean data $S^c = \{(x_i^c, y_i^c)\}_{i=1}^n \sim \mathcal{D}^n$ is as follows:

- The *malicious adversary* attempts to *mark every index/point* $i \in \{1, 2, \ldots, n\}$ and succeeds independently with probability α, for a fixed constant $\alpha \in [0, 0.5)$. Denote the set of all marked indexes by $\mathfrak{P} \subseteq [n]$ and note that $|\mathfrak{P}| \sim Bin(n, \alpha)$.
- The adversary computes, in a possibly randomized manner, a corrupted dataset $S^p = \{(x_i^p, y_i^p)\}_{i=1}^n \in (\mathcal{X} \times \mathcal{Y})^n$, with the only restriction that $(x_i^p, y_i^p) = (x_i^c, y_i^c)$ for all $i \notin \mathfrak{P}$. That is, the adversary can replace all marked data points in an arbitrary manner, with *no assumptions whatsoever* about the points (x_i^p, y_i^p) for $i \in \mathfrak{P}$.

A classic result by [KL93] states that in the context of binary classification any hypothesis space, excluding degenerate cases, is *not* adversarially agnostic PAC learnable against the set of all malicious adversaries. That is, there is no learner that can achieve close to optimal error with high probability against the malicious adversary, even in the infinite data limit. Moreover, Theorem 1 in [KL93] shows that a malicious adversary of power α can always ensure that the error incurred by the learning algorithm is at least $\frac{\alpha}{1-\alpha}$.

Crucially in this adversarial model, the indexes of the points that the adversary can change are independent of the observed data. This assumption is relaxed in the next adversarial model we study.

The nasty adversary model The nasty adversary model was introduced by [BEK02]. Just like the malicious adversary, the nasty adversary can *arbitrarily manipulate* a subset of the data of size $Bin(n, \alpha)$. However, the adversary can choose any randomized procedure, possibly dependent on the observed data, for selecting the subset that can be corrupted. The only constraint is that the adversary needs to ensure that the size of the corrupted subset is distributed as $Bin(n, \alpha)$ with respect to the randomness of the clean data and of the procedure.

Formally, the process of computing the corrupted dataset $S^p = \{(x_i^p, y_i^p)\}_{i=1}^n$ from the clean data $S^c = \{(x_i^c, y_i^c)\}_{i=1}^n \sim \mathcal{D}^n$ is the following:

- The *nasty adversary* observes the clean data and based on S^c computes, in a possibly randomized manner, a subset $\mathfrak{P} \subseteq [n]$ of *marked indexes*. The only constraint on this procedure is that it must be the case that $|\mathfrak{P}| \sim Bin(n, \alpha)$ with the randomness taken with respect to both the clean data and the adversary's computations.
- The adversary computes, in a possibly randomized manner, a corrupted dataset $S^p = \{(x_i^p, y_i^p)\}_{i=1}^n \in (\mathcal{X} \times \mathcal{Y})^n$, with the only restriction that $(x_i^p, y_i^p) = (x_i^c, y_i^c)$ for all $i \notin \mathfrak{P}$. That is, the adversary can replace all marked data points in an arbitrary manner, with *no assumptions whatsoever* about the points (x_i^p, y_i^p) for $i \in \mathfrak{P}$.

The nasty adversary model is strictly stronger than the malicious one. This is because the nasty adversary can always choose to mark every index independently with probability α, hence simulating the malicious adversary. In addition, the nasty adversary can already introduce a harmful bias into the data through the way that it selects the marked data points already. For example, if the clean data contains a small subpopulation of rare, but important, data points, the adversary can choose to always mark the points from this subpopulation. Therefore, the

adversary can ensure that this subpopulation is never present in the corrupted dataset at all. In contrast, the malicious adversary can not in general ensure that, since the subset of the data that it can manipulate is random.

In particular, the hardness results implying the impossibility of learning against a malicious adversary transfer also to the case of the nasty adversary. In fact, for binary classification one can show that a nasty adversary of power α can always ensure that the error incurred by a learning algorithm is at least 2α [BEK02].

2.3 Fairness in machine learning

We now turn to another topic in trustworthy machine learning that is studied in this thesis, namely the one of fairness. As argued in the introduction, with machine learning models penetrating modern automated decision-making systems, it is becoming increasingly important to ensure that ML systems do not discriminate individuals or organizations based on various characteristics protected by law or by certain ethical standards. Such considerations are especially relevant for tasks such as automated screening of job or loan applications, but also for medical data, where it is desirable that a disease recognition model works well for people of all races, say.

Here we discuss what types of fairness notions are commonly considered in the machine learning literature. Then we focus on one of these types, namely on group fairness notions. We discuss how these can be studied alongside with accuracy in the context of binary classification. We also present several group fairness notions that will be studied in this thesis. The short discussion in this section necessarily touches only on a few topics in fair machine learning. For a more thorough introduction we refer to [BHN19, MMS+19].

2.3.1 Types of machine learning fairness

Fairness being a rather general and sophisticated philosophical concept, there are many ways to rigorously define it in the context of supervised machine learning. Focusing on the context of classification, there are three commonly considered types of fairness, namely individual, group and counterfactual fairness.

Individual fairness Individual fairness, first discussed in a machine learning context by [DHP+12], is a constraint on machine learning systems that aims to ensure that, informally, similar instances (e.g. individuals) are treated similarly by the classifier. The fairness notion is *individual* in the sense that it is designed to protect every instance in the population, as it guarantees that each instance is treated similarly to others like it. Formally, the notion of individual fairness depends on two distance metrics - one on the space of the inputs (say, the space of applicants for a job) and one on the space of classifier outputs (e.g. the square distance between the two scores given by the classifier to two applicants). Individual fairness is then understood as a Lipschitz property of the classifier with respect to these two distance metrics [DHP+12]. Choosing the distance notions is therefore the key to incorporating application-specific fairness considerations.

Group fairness Groups fairness, e.g. [CKP09, HPS16], refers to the concept that the decisions of a classier should, on average over the population, be taken without exhibiting a discriminative behavior with respect to a certain protected attribute (e.g. race or gender) of

the inputs. The exact definition of "discriminative behavior" is typically stated in the form of a (conditional) independence property between the classifier output and the protected attribute and may vary between applications. As compared to individual fairness, group fairness properties are often much easier to state, since they do not require a careful construction of appropriate distance measures. On the other hand, group fairness ensures that a fairness property is satisfied in a statistical sense only, that is on average, and therefore it does not explicitly guarantee fair treatment for every individual instance.

Counterfactual fairness Counterfactual fairness [KLRS17] provides an intuitive notion of fair treatment with respect to a protected attribute, by aiming to ensure that the decisions of a classifier would not change in case of an intervention on a particular instance that changes the value of this protected attribute only. For example, in the case of loan applications decisions, the outcome of the application of any black individual should not have been different had they been white (all other presented information being the same). While being both intuitive and related to every individual of the population, counterfactual fairness notions are often hard to verify due to the inherent difficulties of counterfactual inference more generally.

2.3.2 PAC Learning and group fairness

In this thesis we will be focusing on group fairness, where the statistical nature of the fairness constraints allows for tools from PAC learning to be transferred to fairness-aware learning as well. Here we discuss how the fairness-aware learning problem can be formulated in statistical terms.

The key necessary addition to the classic PAC learning framework is the notion of a protected attribute. For any data point, we assume that, in addition to an input $x \in \mathcal{X}$ and an output $y \in \mathcal{Y}$, a protected attribute value $a \in \mathcal{A}$ is also given. This should correspond to a feature with respect to which discriminatory behavior should be avoided, for example race or gender. A training dataset then consists of n triplets $S = \{(x_i, a_i, y_i)\}^n \in (\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n$. We assume that these data points are sampled from an unknown distribution $\mathcal{D} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ over the input-attribute-label triplets space and denote the random variables corresponding to x, a and y as X, A and Y respectively. A fairness-aware learner is a function that takes in a dataset S of triplets and outputs a hypothesis h from a given hypothesis space $\mathcal{H}$.

Different works define $\mathcal{H}$ either as a subset of $\mathcal{Y}^{\mathcal{X}\times\mathcal{A}}$ or as a subset of $\mathcal{Y}^{\mathcal{X}}$. That is, some works, e.g. [HPS16, WGOS17], allow for the classifier to explicitly use the protected attribute when making a classification decision, while others, such as [ABD+18, MW18a], do not. In this thesis we opt for the latter option, so that the classifier is not explicitly using the variables $a \in \mathcal{A}$. Note that while the decision for any particular instance does not use the variable a directly, the way that the hypothesis is selected by the learner during training is fundamentally influenced by the protected attribute values, so that, hopefully, fair treatment is ensured at test time.

There are several reasons why we opt to study classifiers that do not use a directly for their decisions. Firstly, the protected attribute a can always be assumed to be explicitly present among the features of the input variable x, so that the other case is also covered by our analysis. Secondly, in cases when a is not part of the input x, this might be for good reasons: in many circumstances using a protected attribute value directly when making decisions can be considered unethical, even if the intentions are to ensure fair treatment. Finally, in some

situations protected attribute information may be unavailable at test time, for example for privacy reasons.

Similarly to the classic PAC learning setup, the performance of the learner is measured in terms of the performance of the returned hypothesis $h \in \mathcal{H}$ on the distribution $\mathcal{D}$. For measuring accuracy, the expected risk $\mathcal{R}(h)$ can be used again. For fairness, we will again be interested in the extent to which the fairness property is satisfied by h on a population level, that is, with respect to the distribution $\mathcal{D}$. To this end, one can define a *fairness deviation measure* $\Gamma(h, \mathcal{D})$, that quantifies the amount of unfairness that h possesses, with respect to the fairness definition of interest. We give a few examples of such measures in the following section.

2.3.3 Notions of group fairness

Finally, we present the three arguably most popular group fairness notions from the literature, which will also be considered throughout this thesis: demographic parity, equalized odds and equality of opportunity. We also give examples of corresponding fairness deviation measures. The fairness notions are well-defined for a general space $\mathcal{A}$, but we will explicitly consider the case where $\mathcal{A} = \{0, 1\}$ is binary, for example corresponding to a setup where a single disadvantaged group of candidates for a job should be protected from discriminatory decisions. We study the binary case separately since corresponding fairness deviation measures can be readily defined in this context.

Demographic parity Given a distribution $\mathcal{D} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ and a classifier $h : \mathcal{X} \to \mathcal{Y}$, a natural fairness property is to require that the decisions of the classifier are independent of the protected attribute, that is $h(X) \perp\!\!\!\perp A$. This fairness notion is known in the context of machine learning as *demographic parity* [CKP09, HPS16]. In the case when $\mathcal{A} = \{0, 1\}$, this is equivalent to enforcing:

$$\mathbb{P}_{(X,A,Y)\sim\mathcal{D}}(h(X) = 1|A = 0) = \mathbb{P}_{(X,A,Y)\sim\mathcal{D}}(h(X) = 1|A = 1). \tag{2.18}$$

In practice, few classifiers will achieve exact fairness in the sense of demographic parity. In addition, even if a classifier is perfectly fair, this will be impossible to infer from training data, due to the finite sample size effects. Therefore, it is natural to consider a corresponding fairness deviation measure [WGOS17, MW18a, WM19], describing the extend to which a classifier h is unfair. Here we adopt the *mean difference score* measure of [CV10] and [MW18a] for demographic parity

$$\Gamma^{par}(h, \mathcal{D}) = \Big|\mathbb{P}_{(X,A,Y)\sim\mathcal{D}}(h(X) = 1|A = 0) - \mathbb{P}_{(X,A,Y)\sim\mathcal{D}}(h(X) = 1|A = 1)\Big|. \tag{2.19}$$

While being a natural notion of fairness, demographic parity can in many cases be impossible to achieve alongside with high accuracy. In particular, in a realizable PAC learning setup, a perfectly accurate classifier $h^* \in \mathcal{H}$ exists, but it may be the case that $h^*(X)$ strongly correlates with the variable A. This reflects the observation that in many cases the true label, e.g. the suitability for a job, may in fact be correlated with the protected attribute, for example because of better access to education for people from non-disadvantaged backgrounds. This observation naturally leads to the notion of *equalized odds*.

Equalized odds Equalized odds requires that $h(X) \perp\!\!\!\perp A|Y$. That is, the decisions of the classifier can depend on the protected attribute A, in contrast to the implications of

demographic parity, but only through the value of the true label. Equivalently, for the case of binary classification, equalized odds is satisfied if and only if:

$$\mathbb{P}(h(X) = 1|A = 0, Y = y) = \mathbb{P}(h(X) = 1|A = 1, Y = y), \quad \text{for both } y \in \{0, 1\}. \tag{2.20}$$

As in the case of demographic parity, one can consider the amount of unfairness that a classifier possesses, in terms of the deviation from equalized odds, for example

$$\Gamma^{odds}(h, \mathbb{P}) = \frac{1}{2} \sum_{y \in \{0,1\}} \Big| \mathbb{P}(h(X) = 1|A = 0, Y = y) - \mathbb{P}(h(X) = 1|A = 1, Y = y) \Big|. \tag{2.21}$$

Equality of opportunity For some binary classification tasks, one might only be interested in the fairness of classification decisions regarding instances for which the correct label indicates a positive outcome (e.g. receiving a job or being granted a loan). In that case, the constraint of equalized odds can be relaxed for the true negatives. Assuming without loss of generality that a positive outcome is indicated by $Y = 1$, we obtain the *equality of opportunity* fairness constraint $h(X) \perp\!\!\!\perp A|Y = 1$, or equivalently:

$$\mathbb{P}(h(X) = 1|A = 0, Y = 1) = \mathbb{P}(h(X) = 1|A = 1, Y = 1). \tag{2.22}$$

Again, we can also measure the amount of unfairness by

$$\Gamma^{opp}(h, \mathbb{P}) = \Big| \mathbb{P}(h(X) = 1|A = 0, Y = 1) - \mathbb{P}(h(X) = 1|A = 1, Y = 1) \Big|. \tag{2.23}$$

CHAPTER 3

On the Sample Complexity of Adversarial Multi-Source PAC Learning

We now move to the first topic covered in this thesis, namely that of robust learning from corrupted data sources.

As discussed in Chapter 2, the problem of learning from adversarially corrupted data is in general very hard: the classic result of [KL93] states that when a fixed fraction of a training dataset is corrupted within the malicious adversary model, successful learning in the PAC sense is not possible anymore. In other words, there exists no *robust learning algorithm* that could overcome the effects of adversarial corruptions in a constant fraction of the training dataset and approach the optimal model, even in the limit of infinite data.

In this and the next chapter we will study learning from untrusted data in a different setup and show that by tailoring our adversarial model to a specific scenario, learning against a malicious opponent becomes possible, both in theory and in practice. Specifically, we will consider a scenario where a number of datasets, coming from different sources, are given for training. In addition, we will assume that while an unknown subset of these sets may contain corrupted data, the other datasets will contain data sampled i.i.d. from the target distribution. We refer to this problem as "adversarial multi-source learning". In the next section we argue why such a setup is highly relevant from a practical perspective. Then, for the rest of this chapter, we will study the limits of adversarial multi-source learning and provide PAC-style upper and lower bounds on the performance that a learner can achieve under two strong adversarial models. In Chapter 4 we will also study the problem from a practical perspective and provide an algorithm for multi-source learning that is designed to work against more moderate data corruptions, but exhibits a strong performance in a broad range of experiments.

3.1 Motivation and outline

Due to the outstanding performance of modern machine learning algorithms on various real-world tasks, there is an increasing amount of interest by practitioners in producing predictive models, specific to their purposes. In many application domains, however, it may be prohibitively expensive for a single expert to produce a high-quality labeled dataset, that is large enough for training a good model. Therefore, it has become a common practice to obtain data from various external data sources. Examples range from the use of crowdsourcing

platforms, through collecting data from different websites and social networks profiles, to collaborating with other parties working in similar domains. Once access to such datasets is granted, they might either be available for training centrally, or be stored distributedly and used for training via a distributed learning procedure, e.g. via using the recently developed techniques for *federated learning* [MR17].

Naturally, datasets obtained from such sources vary greatly in quality, reliability and relevance for the learning task. For instance, genetic data from multiple laboratories may have been obtained via different measurement devices or data preprocessing techniques [WMP+03]. In the case of crowdsourcing, a typical problem is label bias and label noise, due to incompetent or malicious workers [WLC+10]. More generally, such datasets might also contain gross errors, contaminations and adversarial modifications of the data [BGS+17]. The variety of possible deviations from the target data distribution, as well as the large volume and dimensionality of the data in real-world applications, make the assessment of the quality of the provided data a difficult task.

In this and the next chapter we will study the problem of *how to learn from multiple untrusted sources*, while being *robust to any corruptions* of the data provided by some of them, be it such coming from negligence, bias or malicious behavior. The analogous question to the classic problem of robust learning from one dataset against an adversary is as follows. *Given a number of i.i.d. datasets, a constant fraction of which might have been adversarially manipulated, is there a learning algorithm that overcomes the effect of the corruptions and approaches an optimal model?*

For the rest of this chapter we study this problem, in the centralized data case, from a formal PAC learning perspective and provide a positive answer. Specifically, *our main result is an upper bound on the sample complexity of adversarial multi-source learning, that holds as long as less than half of sources are manipulated (Theorem 4).*

A number of interesting results follow as immediate corollaries. First, we show that any hypothesis class that is uniformly convergent and hence PAC-learnable in the classic i.i.d. sense is also PAC-learnable in the adversarial multi-source scenario. This is in stark contrast to the single-source situation where, as mentioned above, no non-trivial hypothesis class is robustly PAC-learnable. As a second consequence, we obtain the insight that in a cooperative learning scenario, every honest party can benefit from sharing their data with others, as compared to using their own data only, even if some of the participants are malicious.

Besides our main result we prove two additional theorems that shed light on the difficulty of adversarial multi-source learning. First, we prove that the naïve but common strategy of simply merging all data sources and training with some robust procedure on the joint dataset cannot result in a robust learning algorithm (Theorem 5). Second, we prove a lower bound on the sample complexity under very weak conditions (Theorem 6). This result shows that under adversarial conditions a slowdown of convergence is unavoidable, and that in order to approach optimal performance, the number of samples per source must necessarily grow, while increasing the number of sources need not help.

3.2 Related work

To our knowledge, our results are the first that formally characterize the statistical hardness of supervised learning from multiple i.i.d. sources, when a constant fraction of them might be

adversarially corrupted. There are a number of conceptually related works, though, which we will discuss for the rest of this section.

The limits of learning from unreliable data sources Most related is the work of [QV18], as well as the follow-up works of [CLM19, JO19], that aim at estimating discrete distributions from multiple batches of data, some of which have been adversarially corrupted. The main difference to our results is the focus on finite data domains and on estimating the underlying probability distribution rather than learning a hypothesis.

[Qia18] studies collaborative binary classification: a learning system has access to multiple training datasets and a subset of them can be adversarially corrupted. In this setup, the uncorrupted sources are allowed to have different input distributions, but share a common labelling function. The author proves that it is possible to robustly learn individual hypotheses for each source, but a single shared hypothesis cannot be learned robustly. For the specific case that all data distributions are identical, the setup matches ours, though only for binary classification in the realizable case, and with a different adversarial model.

In a similar setting, [MMM19] show, in particular, that an adversary can increase the probability of any "bad property" of the learned hypothesis by a term at least proportional to the fraction of manipulated sources. These results differ from ours, by their assumption that different sources have different distributions, which renders the learning problem much harder.

Byzantine-resilience in distributed and federated learning Another related general direction is the research on Byzantine-resilient distributed learning, which has seen significant interest recently, e.g. [BGS+17, CSX17, YCRB18, YCKB19, AAZL18]. There the focus is on learning by exchanging gradient updates between nodes in a distributed system, an unknown fraction of which might be corrupted by an omniscient adversary and may behave arbitrarily. These works tend to design defences for specific gradient-based optimization algorithms, such as SGD, and their theoretical analysis usually assumes strict conditions on the objective function, such as convexity or smoothness. Nevertheless, the (nearly) tight sample complexity upper and lower bounds developed for Byzantine-resilient gradient descent [YCRB18] and its stochastic variant [AAZL18] are relevant to our results and are therefore discussed in detail in Sections 3.4.2 and 3.5.2. The worst-case performance of distributed SGD has also been studied in the context of asynchronous training [DSZO+15, ADSK18].

The non-i.i.d. split of the data on local devices is one of the main characteristics of *federated learning* and the pioneering work of [MMR+17] addresses this by occasionally averaging local models to ensure global consistency. There is also a large body of literature on attacks and defences in this context, e.g. [SCST17, BCMC19, FYB18]. Apart from focusing on iterative gradient-based optimization procedures, these works also allow for natural variability in the distributions of the uncorrupted data sources.

Other approaches to robust learning from unreliable sources Learning from multiple sources is a topic relevant for many applications of machine learning and data corruption is a problem acknowledged in some of these areas. In particular, [BWKT14, KTK12, ABHM17] and references therein consider the problem of label noise in *crowdsourced data*. However, they only focus on label corruptions. [Fen17] considers the fundamental limits of learning from adversarial distributed data, but in the case when *each of the nodes* can iteratively send corrupted updates with certain probability. [FXM14] provide a method for distributing the computation of any robust learning algorithm that operates on a single large dataset.

Robustness has also been explored in the context of *multi-view* learning, where data arrives from various feature extractors [ZXXS17, Xie17, ZIK17].

Robust learning from a single data source There is a vast body of literature focusing on robustness of learning algorithms to corruptions *within* a dataset, e.g. [Tuk60, KL93, BEK02, Hub11, DKK+16, PSBR18], and on identifying data corruptions at *prediction time*, e.g. [HG17, SL18]. These lines of work are orthogonal to ours, since we consider *multiple training datasets*, some of which are corrupted, and hence a literature review in this direction is beyond the scope of our discussion. Most relevant are the works of [KL93, BEK02], who, as already discussed in Chapter 2, study the fundamental limits of PAC learning from a single, adversarially corrupted data source. We compare our results to this classic scenario in Section 3.4.1.

3.3 Preliminaries

In this section we introduce the technical definitions that are necessary to formulate and prove our main results. We start by introducing the relevant notation and reminding the reader of the setup of supervised learning, as laid out in Section 2.1.2. We then introduce the setting of learning from multiple sources and notions of adversaries of different strengths in this context.

3.3.1 Notation and Background

Let $\mathcal{X}$ and $\mathcal{Y}$ be given input and output sets, respectively, and $\mathcal{D} \in \mathcal{P}(\mathcal{X} \times \mathcal{Y})$ be a fixed but unknown probability distribution. By $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ we denote a loss function, and by $\mathcal{H} \subset \{h : \mathcal{X} \to \mathcal{Y}\}$ a set of hypotheses. All of these quantities are assumed arbitrary but fixed for the purpose of the work within this chapter.

A *(statistical) learner* is a function $\mathcal{L} : \cup_{m=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^m \to \mathcal{H}$. In the classic supervised learning scenario, the learner has access to a *training set* of m labelled examples, $\{(x_1, y_1), \ldots, (x_m, y_m)\}$, sampled i.i.d. from D, and aims at learning a hypothesis $h \in \mathcal{H}$ with small *risk*, i.e. expected loss, under the unknown data distribution,

$$\mathcal{R}(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}(\ell(h(x), y)). \tag{3.1}$$

Recall that *PAC-learnability* is a key property of the hypothesis set, which ensures the existence of an algorithm that performs successful learning:

Definition 7 (**PAC-Learnability**). *We call $\mathcal{H}$ (agnostic)* probably approximately correct (PAC) learnable *with respect to ℓ, if there exists a learner $\mathcal{L}$ and a function $m_{\mathcal{H},\ell} : (0,1)\times(0,1) \to \mathbb{N}$, such that for any $\epsilon, \delta \in (0,1)$, whenever S is a set of $m \geq m_{\mathcal{H},\ell}(\epsilon, \delta)$ i.i.d. labelled samples from $\mathcal{D}$, then with probability at least $1 - \delta$ over the sampling of S:*

$$\mathcal{R}(\mathcal{L}(S)) \leq \min_{h\in\mathcal{H}} \mathcal{R}(h) + \epsilon. \tag{3.2}$$

Another important concept related to PAC-learnability is that of *uniform convergence*. Here we use a version of this property that is slightly different than the one presented in Section 2.1.2.

Definition 8 (**Uniform convergence**). *We say that $\mathcal{H}$ has the* uniform convergence *property with respect to ℓ with rate $s_{\mathcal{H},\ell}$, if there exists a function $s_{\mathcal{H},\ell} : \mathbb{N}\times(0,1)\times\bigcup_{m=1}^{\infty} (\mathcal{X}\times\mathcal{Y})^m \to \mathbb{R}$, such that for any distribution $\mathcal{D} \in \mathcal{P}(\mathcal{X}\times\mathcal{Y})$ and any $\delta \in (0,1)$:*

- *given* m *samples* $S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \overset{i.i.d.}{\sim} \mathcal{D}$*, with probability at least* $1 - \delta$ *over the data :*

$$\sup_{h \in \mathcal{H}} |\mathcal{R}(h) - \widehat{\mathcal{R}}(h)| \leq s_{\mathcal{H},\ell}(m, \delta, S), \tag{3.3}$$

  where $\hat{\mathcal{R}}(h)$ *is the empirical risk of the hypothesis* h*.*

- $s_{\mathcal{H},\ell}(m, \delta, S_m) \to 0$ *as* $m \to \infty$*, for any sequence* $(S_m)_{m \in \mathbb{N}}$ *with* $S_m \in (\mathcal{X} \times \mathcal{Y})^m$*.*

It is easy to see that the this version is equivalent to the one in Definition 3. We only state the property with the sample complexity used as an explicit bound on the gap between the empirical and the true risk, as this simplifies the layout of our analysis later on.

Throughout this chapter we drop the dependence of $s_{\mathcal{H},\ell}$ on $\mathcal{H}$ and ℓ and simply write s.

3.3.2 Multi-source learning

Our focus in this chapter is on learning from multiple data sources. For simplicity of exposition, we assume that they all provide the same number of data points, i.e. the training data consists of N groups of m samples each, where $m, N \in \mathbb{N}$ are fixed integers.

Formally, we denote by $(\mathcal{X} \times \mathcal{Y})^{N \times m}$ the set of all possible collections (i.e. unordered sequences) of N groups of m datapoints each. A (statistical) *multi-source learner* is a function $\mathcal{L} : \cup_{N=1}^{\infty} \cup_{m=1}^{\infty} (\mathcal{X} \times \mathcal{Y})^{N \times m} \to \mathcal{H}$ that takes such a collection of datasets and returns a predictor from $\mathcal{H}$.

3.3.3 Robust Multi-Source Learning

Informally, one considers a learning system *robust* if it is able to learn a good hypothesis, even when the training data is not perfectly i.i.d., but contains some artifacts, e.g. annotation errors, a selection bias or even malicious manipulations.

In lines with the discussion in Section 2.2.2, we model this by assuming the presence of an *adversary*, that observes the original datasets and outputs potentially manipulated versions. The learner then has to operate on the manipulated data without knowledge of what the original one had been or what manipulations have been made.

The multi-source analogue to the definition of a single-source adversary is as follows:

Definition 9 (Adversary). *An adversary is any function* $\mathfrak{A} : (\mathcal{X} \times \mathcal{Y})^{N \times m} \to (\mathcal{X} \times \mathcal{Y})^{N \times m}$*.*

Throughout the chapter, we denote by $S' = \{S'_1, S'_2, \ldots, S'_N\}$ the original, uncorrupted datasets, drawn i.i.d. from $\mathcal{D}$, and by $S = \{S_1, S_2, \ldots, S_N\} = \mathfrak{A}(S')$ the datasets returned by the adversary.

Different scenarios are obtained by giving the adversary different amounts of power. For example, a weak adversary might only be able to randomly flip labels, i.e. simulate the presence of label noise. A much stronger adversary would be one that can potentially manipulate all data and do so with knowledge not only of all of the datasets but also of the underlying data distribution and the learning algorithm to be used later.

Here we adopt the latter view, as it leads to much stronger robustness guarantees. We define two adversary types that can make arbitrary manipulations to data sources, but only influence

a certain subset of them. The two notions are inspired by the two adversarial models presented in Section 2.2.3.

Definition 10 (**Fixed-Set Adversary**). *Let $G \subset [N]$. An adversary is called* fixed-set (with preserved set G)*, if it only influences the datasets outside of G. That is, $S_i = S_i'$ for all $i \in G$.*

Definition 11 (**Flexible-Set Adversary**). *Let $k \in \{0, 1, \ldots, N\}$. An adversary is called* flexible-set *(with preserved size k), if it can influence any $N - k$ of the N given datasets. That is, there exists a set $G \subset [N]$, such that $|G| = k$ and $S_i = S_i'$ for all $i \in G$.*

In both cases, we call the fraction α of corrupted datasets the *power* of the adversary, i.e. $\alpha = \frac{N-|G|}{N}$ for the fixed-set and $\alpha = \frac{N-k}{N}$ for the flexible-set adversaries.

While similarly defined, the fixed-set adversary is strictly weaker than the flexible-set one, as the latter one can first inspect all data and then choose which subset to modify, while the former one is restricted to a fixed, data-independent subset of sources. In particular, the flexible-set adversary can already bias the distribution of the data by throwing out a carefully chosen set of sources, before replacing them with new data.

Both adversary models are inspired by real-world considerations and analogues have appeared in a number of other research areas. The fixed-set adversary is essentially the multi-source equivalent of the malicious adversary model [KL93]. It is an appropriate model in situations in which N parties collaborate on a single learning task, but an unknown and fixed set of them are compromised, e.g. by hackers that can act maliciously and collude with each other or due to systematic data issues, such as label bias. This is a similar reasoning as in *Byzantine-robust optimization* [AAZL18], where an unknown subset of computing nodes are assumed to behave arbitrarily, thereby disrupting the optimization progress.

The flexible-set adversary corresponds to the nasty adversary model of [BEK02]. It models a situation where a malicious party can observe all of the available datasets and choose which ones to corrupt, up to a certain budget. This is similar not only to the classic model of [BEK02], but also to models from robust mean estimation, e.g. [DKK+19a], where the adversary can again influence which subset of the data to modify once the whole dataset is observed.

Whether robust learning in the presence of an adversary is possible for a certain hypothesis set or not is captured by the following definition, which is a multi-source analogue of Definition 6:

Definition 12. *A hypothesis set, $\mathcal{H}$, is called* multi-source PAC-learnable *against the class of fixed-set adversaries (or flexible-set adversaries) and with respect to ℓ, if there exists a multi-source learner $\mathcal{L}$ and a function $m : (0,1)^2 \to \mathbb{N}$, such that for any $\epsilon, \delta \in (0,1)$, any distribution $\mathcal{D} \in \mathcal{P}(\mathcal{X} \times \mathcal{Y})$ and any set $G \subset [N]$ of size $|G| > \frac{1}{2}N$ (or any $\alpha < \frac{1}{2}$), whenever $S' \in (\mathcal{X} \times \mathcal{Y})^{N \times m}$ is a collection of N datasets of $m \geq m(\epsilon, \delta)$ i.i.d. labelled samples from $\mathcal{D}$ each, then with probability at least $1 - \delta$ over the sampling of S':*

$$\mathcal{R}(\mathcal{L}(\mathfrak{A}(S')) \leq \min_{h \in \mathcal{H}} \mathcal{R}(h) + \epsilon, \tag{3.4}$$

uniformly against all fixed-set adversaries with preserved set G (or all flexible-set adversaries of power α). A learner, $\mathcal{L}$, with this property is called robust multi-source learner *for $\mathcal{H}$.*

In particular, the same learner $\mathcal{L}$ should work against any adversary and for any α or set G. At the same time, the adversary is arbitrary once $\mathcal{L}$ is fixed, so in particular it can depend on the learning algorithm.

Note that the robust learner should achieve optimal error as $m \to \infty$, while N can stay constant. This reflects that we want to study adversarial multi-source learning in the context of a constant and potentially not very large number of sources. In fact, our lower bound results in Section 3.5 show that the adversary can always prevent the learner from approaching optimal risk in the opposite regime of constant m and $N \to \infty$.

3.4 On the sample complexity of adversarial multi-source learning

In this section, we present our main result in this chapter, a theorem that states that whenever $\mathcal{H}$ has the uniform convergence property, there exists an algorithm that guarantees a bounded excess risk against both the fixed-set and the flexible-set adversary. We then derive and discuss some instantiations of the general result that shed light on the sample complexity of PAC learning in the adversarial multi-source learning setting. Finally, we provide a high-level sketch of the theorem's proof.

3.4.1 Main result

Theorem 4. *Let $N, m, k \in \mathbb{N}$ be integers, such that $k \in (N/2, N]$. Let $\alpha = \frac{N-k}{N} < \frac{1}{2}$ be the proportion of corrupted sources. Assume that $\mathcal{H}$ has the uniform convergence property with rate function s. Then there exists a learner $\mathcal{L} : (\mathcal{X} \times \mathcal{Y})^{N \times m} \to \mathcal{H}$ with the following two properties.*

(a) Let G be a fixed subset of $[N]$ of size $|G| = k$. For $S' = \{S'_1, \dots, S'_N\} \overset{i.i.d.}{\sim} \mathcal{D}$, with probability at least $1 - \delta$ over the sampling of S':

$$\mathcal{R}(\mathcal{L}(\mathcal{A}(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) \leq 2s\Big(km, \frac{\delta}{2}, S_G\Big) + 6\alpha \max_{i \in [N]} s\Big(m, \frac{\delta}{2N}, S_i\Big) \tag{3.5}$$

uniformly against all fixed-set adversaries with preserved set G, where $S = \{S_1, \dots, S_N\} = \mathcal{A}(S')$ is the dataset modified the adversary and $S_G = \cup_{i \in G} S_i$ is the set of all uncorrupted data.

(b) For $S' = \{S'_1, \dots, S'_N\} \overset{i.i.d.}{\sim} \mathcal{D}$, with probability at least $1 - \delta$ over the sampling of S':

$$\mathcal{R}(\mathcal{L}(\mathfrak{A}(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) \leq 2s\Big(km, \frac{\delta}{2\binom{N}{k}}, S_G\Big) + 6\alpha \max_{i \in [N]} s\Big(m, \frac{\delta}{2N}, S_i\Big) \tag{3.6}$$

uniformly against all flexible-set adversaries with preserved size k, where $S = \{S_1, \dots, S_N\} = \mathfrak{A}(S')$ is the dataset returned by the adversary, G is the set of sources not modified by the adversary and $S_G = \cup_{i \in G} S_i$ is the set of all uncorrupted data.

The learner $\mathcal{L}$ is in fact explicit, we define and discuss it in the proof sketch that we provide in Section 3.4.3. The complete proof is provided in Appendix A.1.

As an immediate consequence we obtain:

Corollary 1. *Assume that $\mathcal{H}$ has the uniform convergence property. Then $\mathcal{H}$ is multi-source PAC-learnable against the class of fixed-set and the class of flexible-set adversaries.*

Proof. It suffices to show that for any $\delta \in (0,1)$, the right hand sides of (3.5) and (3.6) converge to 0 for $m \to \infty$. This it true, since $s(\bar{m}, \bar{\delta}, \bar{S}) \to 0$ as $\bar{m} \to \infty$ for any $\bar{\delta}$ and $\bar{S}$, by the definition of uniform convergence. Since the same learner works regardless of the choice of G and/or α, the result follows. □

Discussion. Corollary 1 is in sharp contrast with the situation of single dataset PAC robustness. As discussed in the Preliminaries section, in the single dataset case, where an adversary controls an α-fraction of the individual data points, a malicious adversary can ensure that no learner can recover a hypothesis with accuracy better than $\alpha/(1-\alpha)$ [KL93]. Similarly, a nasty adversary can ensure an irreducible error of 2α [BEK02]. Both results hold regardless of the value of m, thus showing that PAC-learnability is not fulfilled.

3.4.2 Rates of convergence

While Theorem 4 is most general, it does not yet provide much insight into the actual sample complexity of the adversarial multi-source PAC learning problem, because the rate function s might behave in different ways. In this section we give more explicit upper bounds in terms a standard complexity measure of hypothesis spaces – the Rademacher complexity. Let

$$\mathfrak{R}_S(\ell \circ \mathcal{H}) = \mathbb{E}_\sigma\Big(\sup_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i \ell(h(x_i), y_i)\Big), \tag{3.7}$$

be the (empirical) Rademacher complexity of $\mathcal{H}$ with respect to the loss function ℓ on a training set $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. Here $\{\sigma_i\}_{i=1}^n$ are i.i.d. Rademacher random variables. Let $S_G = \bigcup_{i\in G} S_i$, $\mathfrak{R}_i = \mathfrak{R}_{S_i}(\ell \circ \mathcal{H})$ and $\mathfrak{R}_G = \mathfrak{R}_{S_G}(\ell \circ \mathcal{H})$. Assume also that the loss function ℓ is bounded, so that for some constant $M > 0$, $\ell(y_1, y_2) \le M$ for all $y_1, y_2 \in \mathcal{Y}$.

Rates for the fixed-set adversary.

An application of Theorem 4 together with a standard uniform concentration result gives:

Corollary 2. *In the setup of Theorem 4, against any fixed-set adversary, it holds that*

$$\mathcal{R}(\mathcal{L}(\mathfrak{A}(S'))) - \min_{h\in\mathcal{H}} \mathcal{R}(h) \le 4\mathfrak{R}_G + 6M\sqrt{\frac{\log(\frac{4}{\delta})}{2km}} + \quad \alpha\Big(18M\sqrt{\frac{\log\left(\frac{4N}{\delta}\right)}{2m}} + 12\max_{i\in[N]}\mathfrak{R}_i\Big). \tag{3.8}$$

The full proof is included in the supplementary material.

In many common learning settings, the Rademacher complexity scales as $\mathcal{O}(1/\sqrt{n})$ with the sample size n (see e.g. [BBL04]). Thereby, we obtain the following rates against the fixed-set adversary:

$$\widetilde{\mathcal{O}}\Big(\frac{1}{\sqrt{km}} + \alpha\frac{1}{\sqrt{m}}\Big), \tag{3.9}$$

where the $\widetilde{\mathcal{O}}$-notation hides constant and logarithmic factors.

The results in Corollary 2 and Equation (3.9) allow us to reason about the type of guarantees that can be achieved given a certain amount of data. However, they also imply an explicit upper bound on the sample complexity of adversarial multi-source learning (i.e. an upper bound on the smallest possible $m(\epsilon, \delta)$ in Definition 12) of the form:

$$m(\epsilon, \delta) \leq \mathcal{O}\left(\frac{\log(\frac{N}{\delta})}{\epsilon^2}\left(\frac{1}{\sqrt{(1-\alpha)N}} + \alpha\right)^2\right). \tag{3.10}$$

Discussion. We can make a number of observations from Equation (3.9). The $\sqrt{1/km}$-term is the rate one expects when learning from k (uncorrupted) sources of m samples each, that is from all the available uncorrupted data. The $\sqrt{1/m}$-term reflects the rate when learning from any single source of m samples, i.e. without the benefit of sharing information between sources. The latter enters weighted by α, i.e. it is directly proportional to the power of the adversary. In the limit of $\alpha \to 0$ (i.e. all N sources are uncorrupted, $k \to N$), the bound becomes $\widetilde{\mathcal{O}}(\sqrt{1/Nm})$. Thus, we recover the classic convergence rate for learning from Nm samples in the non-realizable case. This fact is interesting, as the robust learner of Theorem 4 actually does not need to know the value of α for its operation. Consequently, the same algorithm will work robustly if the data contains manipulations but without an unnecessary overhead (i.e. with optimal rate), if all data sources are in fact uncorrupted.

Another insight follows from the fact that for reasonably small α, we have:

$$\widetilde{\mathcal{O}}\Big(\frac{1}{\sqrt{km}} + \alpha\frac{1}{\sqrt{m}}\Big) \ll \widetilde{\mathcal{O}}\Big(\frac{1}{\sqrt{m}}\Big), \tag{3.11}$$

so learning from multiple, even potentially manipulated, datasets converges to a good hypothesis faster than learning from a single uncorrupted dataset. This fact can be interpreted as encouraging cooperation: any of the *honest* parties in the multi-source setting with fixed-set adversary will benefit from making their data available for multi-source learning, even if some of the other parties are malicious.

Comparison to Byzantine-robust optimization. Our obtained rates for the fixed-set adversary can also be compared to the state-of-art convergence results for Byzantine-robust distributed optimization, where the compromised nodes are also fixed, but unknown. [YCRB18] and [AAZL18] develop robust algorithms for gradient descent and stochastic gradient descent respectively, achieving convergence rates of order

$$\widetilde{\mathcal{O}}\Big(\frac{1}{\sqrt{km}} + \alpha\frac{1}{\sqrt{m}} + \frac{1}{m}\Big) \tag{3.12}$$

for $\alpha < 1/2$ unknown. Clearly, these rates resemble ours, except for the additional $1/m$-term, which matters when α is 0 or very small. As shown in [YCRB18], this term can also be made to disappear if an upper bound $\beta \geq \alpha$ is assumed to be known a priori.

Overall, these similarities should not be over-interpreted, as the results for Byzantine-robust optimization describe practical gradient-based algorithms for distributed optimization under various technical assumptions, such as convexity, smoothness of the loss function and bounded variance of the gradients. In contrast, our results are purely statistical, not taking computational cost into account, but holds in a much broader context, for any hypothesis space that has the uniform convergence property of suitable rate and without constraints on the optimization method to be used. Additionally, our rates improve automatically in situations where uniform convergence is faster.

Algorithm 3.1: Dataset filtering for robust multi-source learning

Inputs: $S_1, \ldots, S_N$
Initialize $\mathrm{T} = \{\}$ // trusted sources
for $i = 1, \ldots, N$ **do**
  if $d_{\mathcal{H}}\left(S_i, S_j\right) \leq s\left(m, \frac{\delta}{2N}, S_i\right) + s\left(m, \frac{\delta}{2N}, S_j\right)$,
    for at least $\lfloor \frac{N}{2} \rfloor$ values of $j \neq i$, **then**
    $\mathrm{T} = \mathrm{T} \cup \{i\}$
  end if
end for
Return: $\bigcup_{i \in \mathrm{T}} S_i$ // all data of trusted sources

Rates for the flexible-set adversary

An analogous result to Corollary 2 holds also for flexible-set adversaries:

Corollary 3. *In the setup of Theorem 4, against any flexible-set adversary, it holds that*

$$\mathcal{R}(\mathcal{L}(\mathfrak{A}(S^{'}))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) \leq 4\mathfrak{R}_G + 12\alpha \max_{i \in [N]} \mathfrak{R}_i + \widetilde{\mathcal{O}}\left(\frac{\sqrt[4]{\alpha}}{\sqrt{m}}\right). \tag{3.13}$$

The proof is provided in the supplemental material.

Making the same assumptions as above, we obtain a sample complexity rate

$$\widetilde{\mathcal{O}}\left(\frac{1}{\sqrt{km}} + \frac{\sqrt[4]{\alpha}}{\sqrt{m}}\right). \tag{3.14}$$

which differs from (3.9) only in the rate of dependence on α, which, if at all, matters only for very small (but non-zero) α. Despite the difference, most of our discussion above still applies. In particular, even for the flexible-set adversary the same learning algorithm exhibits robustness for $\alpha > 0$ and achieves optimal rates for $\alpha = 0$.

Moreover, an explicit upper bound on the sample complexity against a flexible-set adversary is given by:

$$m(\epsilon, \delta) \leq \widetilde{\mathcal{O}}\left(\frac{1}{\epsilon^2}\left(\frac{1}{\sqrt{(1-\alpha)N}} + \sqrt[4]{\alpha}\right)^2\right). \tag{3.15}$$

3.4.3 Proof Sketch for Theorem 4

The proof of Theorem 4 consists of two parts. First, we introduce a filtering algorithm, that attempts to determine which of the data sources can be trusted, meaning that it should be safe to use them for training a hypothesis. Note that this can be because they were not manipulated, or because the manipulations are too small to have negative consequences. The output of the algorithm is a new *filtered* training set, consisting of all data from the trusted sources only. Second, we show that training a standard single-source learner on the filtered training set yields the desired results.

Step 1. Pseudo-code for the filtering algorithm is provided in Algorithm 3.1. The crucial component is a carefully chosen notion of distance between the datasets, called *discrepancy*,

that we define and discuss below. It guarantees that if two sources are close to each other then the difference of training on one of them compared to the other is small.

To identify the trusted sources, the algorithm checks for each source how close it is to all other sources with respect to the discrepancy distance. If it finds the source to be closer than a threshold to at least half of the other sources, the source is marked as trusted, otherwise it is not. To show that this procedure does what it is intended to do it suffices to show that two properties hold with high probability: 1) all trusted sources are safe to be used for training, 2) at least all uncorrupted sources will be trusted.

Property 1) follows from the fact that if a source has small distance to at least half of the other datasets, it must be close to at least one of the uncorrupted sources. By the property of the discrepancy distance, including it in the training set will therefore not affect the learning very negatively. Property 2) follows from a concentration of mass argument, which guarantees that for any uncorrupted source its distance to all other uncorrupted sources will approach zero at a well-understood rate. Therefore, with a suitably selected threshold, at least all uncorrupted sources will be close to each other and end up in the trusted set with high probability.

Discrepancy Distance. For any dataset $S_i \in (\mathcal{X} \times \mathcal{Y})^m$, let

$$\widehat{\mathcal{R}}_i(h) = \frac{1}{m} \sum_{(x,y) \in S_i} \ell(h(x), y) \tag{3.16}$$

be the empirical risk of a hypothesis h with respect to the loss ℓ. The (empirical) *discrepancy distance* between two datasets, S_i and S_j, is defined as

$$d_{\mathcal{H}}(S_i, S_j) = \sup_{h \in \mathcal{H}} \Big(|\widehat{\mathcal{R}}_i(h) - \widehat{\mathcal{R}}_j(h)| \Big). \tag{3.17}$$

This is the empirical counterpart of the so-called discrepancy distance, which, together with its unsupervised form, is widely adopted within the field of domain adaptation [KBDG04, BDBC+10, MM12]. Intuitively, the discrepancy between the two (empirical) distributions is large, if there exists a predictor that performs well on one of them and badly on the other. On the other hand, if all functions in the hypothesis class perform similarly on both, then the distributions have low discrepancy. Therefore, the discrepancy is used to bound the maximum possible effect of distribution drift on a learning system.

Unlike other notions of distance between distributions, such as the total variation distance or the Kullback-Leibler divergence, the discrepancy is easy to estimate from samples and is not overly strict, since it depends on the learning setup. As shown in [KBDG04, BDBC+10], for randomly sampled datasets, the empirical discrepancy concentrates with well-understood rates to its distributional value, in particular to zero, if two sources have the same underlying data distributions. The empirical discrepancy is well-defined even for data not sampled from a distribution, though, and together with the uniform convergence property it allows us to bound the effect of training on one dataset rather than another.

Step 2. Let $S_T = \bigcup_{i \in T} S_i$ be the output of the filtering algorithm, i.e. the union of all trusted datasets. Then, for any $h \in \mathcal{H}$, the empirical risk over S_T can be written as

$$\widehat{\mathcal{R}}_{\mathrm{T}}(h) = \frac{1}{|\mathrm{T}|} \sum_{i \in \mathrm{T}} \widehat{\mathcal{R}}_i(h) \tag{3.18}$$

We need to show that training on S_{T}, e.g. by minimizing $\widehat{\mathcal{R}}_{\mathrm{T}}(h)$, with high probability leads to a hypothesis with small risk under the true data distribution $\mathcal{D}$.

By construction, we know that for any trusted source S_i, there exists an uncorrupted source S_j, such that the difference between $\widehat{\mathcal{R}}_i(h)$ and $\widehat{\mathcal{R}}_j(h)$ is bounded by a suitably chosen term (that depends on the growth function s). By the uniform convergence property of $\mathcal{H}$, we know that for any uncorrupted source, the difference between $\widehat{\mathcal{R}}_i(h)$ and the true risk $\mathcal{R}(h)$ can also be bounded in terms of the growth function s. In combination, we obtain that $\widehat{\mathcal{R}}_{\mathrm{T}}(h)$ is a suitably good estimator of the true risk, uniformly over all $h \in \mathcal{H}$. Consequently, S_{T} can be used for successful learning.

For the formal derivations and, in particular, the choice of thresholds, please see the supplemental material.

3.5 On the hardness of adversarial multi-source learning

We now take an orthogonal view compared to Section 3.4, and study where the hardness of the multi-source PAC learning stems from and what allows us to nevertheless overcome it. For this, we prove two additional results that describe fundamental limits of how well a learner can perform in the multi-source adversarial setting.

For simplicity of exposition we focus on binary classification. Let $\mathcal{Y} = \{-1, 1\}$ and ℓ be the zero-one loss, i.e. $\ell(y, \bar{y}) = \mathbb{1}\{y \neq \bar{y}\}$. Following [BEK02], we define:

Definition 13. *A hypothesis space $\mathcal{H}$ over an input set $\mathcal{X}$ is said to be non-trivial, if there exist two points $x_1, x_2 \in \mathcal{X}$ and two hypotheses $h_1, h_2 \in \mathcal{H}$, such that $h_1(x_1) = h_2(x_1)$, but $h_1(x_2) \neq h_2(x_2)$.*

3.5.1 What makes robust learning possible?

We show that if the learner does not make use of the multi-source structure of the data, i.e. it behaves as a single-source learner on the union of all data samples, then a (multi-source) fixed-set adversary can always *prevent* PAC-learnability.

Theorem 5. *Let $\mathcal{H}$ be a non-trivial hypothesis space. Let m and N be any positive integers and let G be a fixed subset of $[N]$ of size $k \in \{1, \ldots, N-1\}$. Let $\mathcal{L} : (\mathcal{X} \times \mathcal{Y})^{N \times m} \to \mathcal{H}$ be a multi-source learner that acts by merging the data from all sources and then calling a single-source learner. Let $S' \in (\mathcal{X} \times \mathcal{Y})^{N \times m}$ be drawn i.i.d. from $\mathcal{D}$. Then there exists a distribution $\mathcal{D}$ with $\min_{h \in \mathcal{H}} \mathcal{R}(h) = 0$ and a fixed-set adversary $\mathfrak{A}$ with index set G, such that:*

$$\mathbb{P}_{S' \sim \mathcal{D}}\Big(\mathcal{R}\big(\mathcal{L}(\mathfrak{A}(S')\big) > \frac{\alpha}{8(1-\alpha)}\Big) > \frac{1}{20}, \tag{3.19}$$

where $\alpha = \frac{N-k}{N}$ is the power of the adversary.

The proof is provided in the supplemental material. Note that, since the theorem holds for the fixed-set adversary, it automatically also holds for the stronger flexible-set adversary.

The theorem sheds light on why PAC-learnability is possible in the multi-source setting, while in the single source setting it is not. The reason is not simply that the adversary is weaker, because it is restricted to manipulating samples in a subset of datasets instead of being able to choose freely. Inequality (3.19) implies that even against such a weaker adversary, a single-source learner cannot be adversarially robust. Consequently, it is the additional

information that the data comes in multiple datasets, some of which remain uncorrupted even after the adversary was active, that gives the multi-source learner the power to learn robustly.

An immediate consequence of Theorem 5 is also that the common practice of merging the data from all sources and performing a form of empirical risk minimization on the resulting dataset is not a robust learner and therefore suboptimal in the studied context.

3.5.2 How hard is robust learning?

As a tool for understanding the limiting factors of learning in the adversarial multi-source setting, we now establish a lower bound on the achievable excess risk in terms of the number of samples per source and the power of the adversary.

Theorem 6. *Let $\mathcal{H} \subset \{h : \mathcal{X} \to \mathcal{Y}\}$ be a hypothesis space, let m and N be any integers and let G be a fixed subset of $[N]$ of size $k \in \{1, \dots, N-1\}$. Let $S' \in (\mathcal{X} \times \mathcal{Y})^{N \times m}$ be drawn i.i.d. from $\mathcal{D}$. Then the following statements hold for any multi-source learner $\mathcal{L}$:*

(a) Suppose that $\mathcal{H}$ is non-trivial. Then there exists a distribution $\mathcal{D}$ on $\mathcal{X}$ with $\min_{h \in \mathcal{H}} \mathcal{R}(h) = 0$, and a fixed-set adversary $\mathfrak{A}$ with index set G, such that:

$$\mathbb{P}_{S'}\Big(\mathcal{R}\big(\mathcal{L}(\mathfrak{A}(S')\big) > \frac{\alpha}{8m}\Big) > \frac{1}{20}. \tag{3.20}$$

(b) Suppose that $\mathcal{H}$ has VC dimension $d \geq 2$. Then there exists a distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$ and a fixed-set adversary $\mathfrak{A}$ with index set G, such that:

$$\mathbb{P}_{S'}\left(\mathcal{R}\big(\mathcal{L}(\mathfrak{A}(S')\big) - \min_{h \in \mathcal{H}} \mathcal{R}(h) > \sqrt{\frac{d}{1280Nm}} + \frac{\alpha}{16m}\right) > \frac{1}{64}. \tag{3.21}$$

In both cases, $\alpha = \frac{N-k}{N}$ is the power of the adversary.

The proof is provided in the supplemental material. As for Theorem 5, it is clear that the same result holds also for flexible-set adversaries with preserved size k.

Analysis. Inequality (3.20) shows that even in the realizable scenario, the risk might not shrink faster than with rate $\Omega(\alpha/m)$, regardless of how many data sources, and therefore data samples, are available. This is contrast to the i.i.d. situation, where the corresponding rate is $\Omega(1/Nm)$. The difference shows that robust learning with a constant fraction of corrupted sources is only possible if the number of samples per dataset grows.

In inequality (3.21), the term $\Omega(\sqrt{d/Nm})$ is due to the classic lower bound on the sample complexity of binary classification (e.g. Theorem 3.23 in [MRT18]) and corresponds to the fundamental limits of learning, now in the non-realizable case. The $\Omega(\alpha/m)$-term appears as the price of robustness, and as before, it implies that for constant α, $m \to \infty$ is necessary in order to achieve arbitrarily small excess risk, while just $N \to \infty$ does not suffice.

Relation to prior work. Lower bounds of similar structure as in Theorem 6 have also been derived for Byzantine optimization and collaborative learning. In particular, [YCRB18] prove that in the case of distributed mean estimation of a d-dimensional Gaussian on N machines, an α fraction of which can be Byzantine, any algorithm would incur loss of $\Omega(\frac{\alpha}{\sqrt{m}} + \sqrt{\frac{d}{Nm}})$. [AAZL18] construct specific examples of a Lipschitz continuous and a strongly convex function,

such that no distributed stochastic optimization algorithm, working with an α-fraction of Byzantine machines, can optimize the function to error less than $\Omega(\frac{\alpha}{\sqrt{m}} + \sqrt{\frac{d}{Nm}})$, where d is the number of parameters. For realizable binary classification in the context of collaborative learning, [Qia18] prove that there exists a hypothesis space of VC dimension d, such that no learner can achieve excess risk less than $\Omega(\alpha d/m)$.

Besides the different application scenario, the main difference between these results and Theorem 6 is that our bounds hold for *any* hypothesis space $\mathcal{H}$ that is non-trivial (Ineq. (3.20)), or has VC-dimension $d \geq 2$ (Ineq. (3.21)), while the mentioned references construct explicit examples of hypothesis spaces or stochastic optimization problems where the bounds hold. In particular, our results show that the limitations on the learner due the finite total number of samples, the finite number of samples per source and the fraction of unreliable sources α are inherent and not specific to a subset of hard-to-learn hypotheses.

3.6 Summary and subsequent work

We studied the problem of robust learning from multiple unreliable datasets. Rephrasing this task as learning from datasets that might be adversarially corrupted, we introduced the formal problem of adversarial learning from multiple sources, which we studied in the classic PAC setting.

Our main results provide a characterization of the hardness of this learning task from above and below. First, we showed that adversarial multi-source PAC learning is possible for any hypothesis class with the uniform convergence property, and we provided explicit rates for the excess risk (Theorem 4 and Corollaries). The proof is constructive and shows also that integrating robustness comes at a minor statistical cost, as our robust learner achieves optimal rates when run on data without manipulations. Second, we proved that adversarial PAC learning from multiple sources is far from trivial. In particular, it is impossible to achieve for learners that ignore the multi-source structure of the data (Theorem 5). Third, we proved lower bounds on the excess risk under very general conditions (Theorem 6), which highlight an unavoidable slowdown of the convergence rate proportional to the adversary's strength compared to the i.i.d. (adversarial-free) case. Furthermore, in order to facilitate successful learning with a constant fraction of corrupted sources, the number of samples per source has to grow.

Two relevant subsequent works are those of [JO20] and [HK20]. In particular, [JO20] extend the framework of [QV18] to learning of distributions over infinite domains, from untrusted batches of data. They also develop a robust algorithm for binary classification, achieving similar statistical rates to ours. Regarding hardness results, they show how a result from [QV18] can be adapted to prove a lower bound of the form $\mathcal{O}\left(\sqrt{\frac{d}{Nm}} + \frac{\alpha}{\sqrt{m}}\right)$, essentially closing the gap between the lower and the upper bounds known for the adversarial multi-source learning problem. Also regarding hardness results, [HK20] recently gave a number of sample complexity lower bounds for multi-task learning with N tasks that share an optimal hypothesis. Their work shows in particular that no learning procedure can achieve optimal risk as $N \to \infty$, even in this non-adversarial setting, as long as the number of data points per task is constant and the learner has no additional information about the relationship between the tasks, apart from the given data.

CHAPTER 4

Adversarial Multi-Source Learning in Practice

4.1 Motivation and outline

In this chapter we study the problem of multi-source adversarial learning from a more practical perspective. Indeed, while Algorithm 3.1, as studied in the previous section, provides provable guarantees against the strong adversarial models we have considered, it is by and large impractical for real-world tasks. This is because the thresholds used for accepting or rejecting sources are based on worst-case generalization bounds and are therefore often too large and do not detect malicious sources in practice, unless the number of samples per source is prohibitively large. An additional complication is that for many practical learning scenarios the data might have to remain decentralized, because of high communication costs, or it might not be directly available for inspection, due to privacy constraints. In contrast, the analysis from the previous chapter only concerns the centralized data case.

In order to design a more practical learning algorithm for the general problem of learning from unreliable data sources, we make two simplifying, but natural, assumptions. Firstly, we assume that a *small trusted reference dataset* is provided to the learner, in addition to the N untrusted data sources. This is justified in many situations where obtaining *some* clean data is possible, even if expensive. For example, this is the case for medical data, where a trusted professional can be asked to label a limited number of x-ray images manually, even if it is impractical to obtain a dataset large enough for training only from this single expert. Secondly, we will focus on a slightly different data corruption model: we assume that each of the untrusted sources follows its own distribution, which may or may not be close to the target one. This is realistic in many cases where small differences between the distributions of the clean sources are expected, but sources with truly irrelevant or contaminated distributions should be avoided.

In this context, we provide an alternative to the naive approaches of simply training on all data or only on the trusted subset: we propose a method that automatically assigns weights to each of the untrusted sources. To this end, we build up on techniques from the domain adaptation literature and prove an upper bound on the expected loss of a predictor, learned by minimizing any weighted version of the empirical loss. Based on these theoretical insights, our algorithm selects the weights for the sources by approximately minimizing this upper bound.

Intuitively, *the weights are assigned to the sources according to the quality and reliability of the data they provide*, quantified by an appropriate measure of trust we introduce. This is achieved by comparing the data from each source to the *small reference dataset*. The measure can also be computed locally at every source or by a gradient-based optimization procedure, which allows for the implementation of the algorithm under *privacy constraints*, as well as its integration into any *standard distributed learning framework*.

We perform an extensive experimental evaluation[1] of our algorithm and demonstrate its ability to learn from all available data, while successfully suppressing the effect of corrupted or irrelevant sources. It consistently outperforms both naive approaches of learning on all available data directly or learning on the reference dataset only, *for any amount and any type of data contamination* considered. We also observe its performance to be superior to multiple baseline methods from robust statistics and robust distributed learning.

4.2 Related work

In addition to the related work discussed in Chapter 3, a number of works are relevant to the approach to the adversarial multi-source problem taken in this chapter.

In particular, [CSV17, HMWG18a] also study learning with a reference dataset as a protection against data corruption, but focus on a single untrusted dataset only and on convex objectives and label noise respectively.

On the methodological level, we borrow techniques from the field of domain adaptation. To measure the difference between data distributions, we use the same integral probability metric as [MM12, ZL17]. The problem we study is related to *multi-source domain adaptation*, e.g. [CKW08, BDBC$^+$10], and to *multi-task learning* with unlabeled data [PL17]. In particular, our Theorem 7 is similar to a result in [ZZY13]. We refer to the paragraph after Theorem 7 for a more detailed comparison. However, all these works focus on sharing information between similar domains, in order to obtain better predictors for a target task, while we are interested in applying such techniques for detecting untrustworthy sources of data and improving the robustness of the learning procedure.

A relation between robustness and domain adaptation has been explored in the work of [MS14], who use a property called *algorithmic robustness* to derive generalization bounds for domain adaptation. Another related line of work is the one of [MMR09, HMZ18], who provide guarantees for a classifier learned on data from N domains on any target distribution that is a mixture of the distributions of the sources. Domain adaptation techniques were also used by [SHWH19], for improving the test-time robustness of predictive models to adversarial examples.

4.3 Robust learning from untrusted sources

Given a *small reference dataset*, we want to leverage additional training data from *multiple untrusted sources* in an optimal way, so that the obtained predictor performs well on a target distribution. A naive approach will be to trust all data, merge it into one dataset and train end-to-end to obtain a predictive model. Such an approach will intuitively be vulnerable to

[1] Code is available at https://github.com/NikolaKon1994/Robust-Learning-from-Untrusted-Sources

irrelevant or low-quality data provided by some sources. In this section, we design a more *robust* algorithm that instead minimizes a weighted empirical loss.

4.3.1 Theory

Setup. Let $\mathcal{X}$ be an input space and $\mathcal{Y}$ be an output space. Our theoretical setup covers both the case of classification ($\mathcal{Y} = \{1, 2, \ldots, K\}$) and regression ($\mathcal{Y} = \mathbb{R}$). We assume that the learner has access to a *small reference dataset* $S_T := \{(x_{T,1}, y_{T,1}), \ldots, (x_{T,m_T}, y_{T,m_T})\}$ of m_T samples drawn i.i.d. from a target distribution $\mathcal{D}_T$ over $\mathcal{X} \times \mathcal{Y}$. In addition, training data is available from N *untrusted data sources*, each of them characterized by its own distribution, $\mathcal{D}_i$, over $\mathcal{X} \times \mathcal{Y}$, possibly different from $\mathcal{D}_T$. We denote the number of samples from source i by m_i. Let the corresponding i.i.d. datasets be $S_i := \{(x_{i,1}, y_{i,1}), \ldots, (x_{i,m_i}, y_{i,m_i})\} \overset{i.i.d.}{\sim} \mathcal{D}_i$ for each $i = 1, \ldots, N$.

Let $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ be a loss function, bounded by some $M > 0$. For any distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$ and any function $h : \mathcal{X} \to \mathcal{Y}$, denote by

$$\mathcal{R}_{\mathcal{D}}(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left(\ell\left(h(x), y\right)\right)$$

the expected loss of the predictor h with respect to the distribution $\mathbb{P}$. Let $\mathcal{R}_i(h) = \mathcal{R}_{\mathcal{D}_i}(h)$ be the expected loss of a predictor h on the distribution of the i-th source. Denote by $\hat{\mathcal{R}}_i$ the corresponding empirical counterparts.

Given a hypothesis class $\mathcal{H} \subset \{h : \mathcal{X} \to \mathcal{Y}\}$, our goal is to use all samples from the *sources* to construct a hypothesis with low expected loss on the target distribution $\mathcal{D}_T$. Note that if we also want to use the reference data at training time, we can simply include it as one of the data sources.

Source-specific weights. For a vector of weights $\alpha = (\alpha_1, \ldots, \alpha_N)$, such that $\sum_{i=1}^N \alpha_i = 1$ and $\alpha_i \geq 0$ for all i, we define the α-weighted expected risk of a predictor h as:

$$\mathcal{R}_\alpha(h) = \sum_{i=1}^N \alpha_i \mathcal{R}_i(h) = \sum_{i=1}^N \alpha_i \mathbb{E}_{(x,y)\sim\mathcal{D}_i}\left(\ell(h(x), y)\right) \tag{4.1}$$

and its empirical counterpart as:

$$\hat{\mathcal{R}}_\alpha(h) = \sum_{i=1}^N \alpha_i \hat{\mathcal{R}}_i(h) = \sum_{i=1}^N \frac{\alpha_i}{m_i} \sum_{j=1}^{m_i} \ell\left(h(x_{i,j}), y_{i,j}\right). \tag{4.2}$$

With $\mathcal{H}$ as our hypothesis class, let $\hat{h}_\alpha = \operatorname{argmin}_{h\in\mathcal{H}} \hat{\mathcal{R}}_\alpha(h)$.

We aim to find weights α, such that the predictor $\hat{h}_\alpha$ performs well on the target task, i.e. such that $\mathcal{R}_T(\hat{h}_\alpha)$ is small.

Evaluating the quality of a source. Intuitively, a good learning algorithm will assign more weight to sources, whose distribution is similar to the target one, and less weight to those that provide different or low-quality data. Although any standard distance measure on the space of distributions could in theory be used to measure such differences, most of them would not provide any guarantees on the performance of the learned classifier. Furthermore, most similarity measures between distributions, e.g. the Kullback-Leibler divergence, are hard to estimate from finite data and overly strict, as they are independent of the learning setup.

We therefore again adopt the discrepancy distance [MM12], whose empirical version we used in Chapter 3 as a specific notion of distance that depends on the hypothesis class and allows us to reason about the change in performance of a predictor from $\mathcal{H}$ learned on one distribution, but applied to the other. Formally, define the *discrepancy* between the distributions $\mathcal{D}_i$ and $\mathcal{D}_T$ with respect to the hypothesis class $\mathcal{H}$ as:

$$d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right) = \sup_{h \in \mathcal{H}} \left(\left|\mathcal{R}_i(h) - \mathcal{R}_T(h)\right|\right). \tag{4.3}$$

Recall that the discrepancy between the two distributions is large, if there exists a predictor that performs well on one of them and badly on the other. On the other hand, if all functions in the hypothesis class perform similarly on both, then $\mathcal{D}_i$ and $\mathcal{D}_T$ have low discrepancy.

The following theorem provides a bound on the expected loss on the target distribution of the predictor $\hat{h}_\alpha$, i.e. the minimizer of the α-weighted sum of the empirical losses over the source data.

Theorem 7. *Given the setup above, let* $\hat{h}_\alpha = argmin_{h \in \mathcal{H}} \hat{\mathcal{R}}_\alpha(h)$ *and* $h_T^* = argmin_{h \in \mathcal{H}} \mathcal{R}_T(h)$. *For any* $\delta > 0$, *with probability at least* $1 - \delta$ *over the data:*

$$\mathcal{R}_T(\hat{h}_\alpha) \leq \mathcal{R}_T(h_T^*) + 4 \sum_{i=1}^{N} \alpha_i \mathfrak{R}_i\left(\mathcal{H}\right) + 2 \sum_{i=1}^{N} \alpha_i d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right) + 6 \sqrt{\frac{\log\left(\frac{4}{\delta}\right) M^2}{2}} \sqrt{\sum_{i=1}^{N} \frac{\alpha_i^2}{m_i}}, \tag{4.4}$$

where, for each source $i = 1, \ldots, N$,

$$\mathfrak{R}_i\left(\mathcal{H}\right) = \mathbb{E}_\sigma \left(\sup_{f \in \mathcal{H}} \left(\frac{1}{m_i} \sum_{j=1}^{m_i} \sigma_{i,j} \ell(f(x_{i,j}), y_{i,j}) \right) \right)$$

and $\sigma_{i,j}$ *are independent Rademacher random variables.*

A proof is provided in the supplementary material.

We note that a similar result appears as Theorem 5.2 in the arXiv version [ZZY13] of the NIPS paper [ZZY12]. The authors bound the gap between the weighted empirical loss on the source data of any classifier and its expected loss on the target task, with the additional assumption of a deterministic labeling function for each source. Based on this, they study the asymptotic convergence of domain adaptation algorithms as the sample sizes at all sources go to infinity. In contrast, our theorem compares the performance of the minimizer $\hat{h}_\alpha$ of the α-weighted empirical loss on the target task to the performance of the optimal (but unknown) h_T^* and does not require deterministic labeling functions. Our target application is also different, since we use the bound to design learning algorithms that are robust to corrupted or irrelevant data, given finite amount of samples from each source.

4.3.2 From bound to algorithm

Algorithm description. To obtain a good predictor for the target task, we would like to choose α, such that $\mathcal{R}_T(\hat{h}_\alpha)$ is as close as possible to $\mathcal{R}_T(h_T^*)$ (the expected loss of the best hypothesis in $\mathcal{H}$). This suggests selecting the weights by minimizing the right-hand side of (4.4).

Algorithm 4.1: Robust learning from untrusted sources

Inputs: 1. Loss ℓ, hypothesis set $\mathcal{H}$, parameter λ
2. Reference dataset S_T
3. Datasets $S_1, \ldots, S_N$ from the N sources

for $i = 1$ **to** N **do** {Potentially in parallel}
Compute $d_{\mathcal{H}}(S_i, S_T)$
end for
Select α by solving (4.6).
Minimize α-weighted loss: $\hat{h}_\alpha = \operatorname{argmin}_{h \in \mathcal{H}} \hat{\mathcal{R}}_\alpha(h)$
Return: $\hat{h}_\alpha$

While the Rademacher complexities are functions of both the underlying distribution and the hypothesis class, in practice one usually works with a computable upper bound that is distribution-independent (e.g. using VC dimension). For some common examples of such bounds we refer to [BBL04, SSBD14]. In our setting the hypothesis space $\mathcal{H}$ is fixed and therefore these bounds would be identical for all i. Therefore, we expect the $\mathfrak{R}_i(\mathcal{H})$ to be of similar order to each other and the impact of α on the second term in the bound to be negligible. We thus concentrate on optimizing the remaining terms.

Because the true discrepancies are unknown, we estimate them from the data by their empirical counterparts:

$$\begin{aligned} d_{\mathcal{H}}(S_i, S_T) &= \sup_{h \in \mathcal{H}} \left(|\hat{\mathcal{R}}_i(h) - \hat{\mathcal{R}}_T(h)| \right) \\ &= \sup_{h \in \mathcal{H}} \left(\left| \frac{1}{m_i} \sum_{j=1}^{m_i} \ell\left(h\left(x_{i,j}\right), y_{i,j}\right) - \frac{1}{m_T} \sum_{j=1}^{m_T} \ell\left(h\left(x_{T,j}\right), y_{T,j}\right) \right| \right). \end{aligned} \tag{4.5}$$

In summary, the bound suggests to choose a weighting for the sources by minimizing:

$$\begin{aligned} &\min_{\alpha} \sum_{i=1}^{N} \alpha_i d_{\mathcal{H}}(S_i, S_T) + \lambda \sqrt{\sum_{i=1}^{N} \frac{\alpha_i^2}{m_i}}, \\ &\text{subject to: } \sum_{i=1}^{N} \alpha_i = 1 \text{ and } \alpha_i \geq 0 \text{ for all } i, \end{aligned} \tag{4.6}$$

where $\lambda > 0$ is a hyperparameter that can be selected by cross-validation on the reference dataset. The algorithm then proceeds to minimize the α-weighted empirical risk over the sources (4.2), possibly with a regularization term. Pseudocode of the algorithm is given in Algorithm 4.1.

We note that in general computing the empirical discrepancies can be a hard problem. However,

for the $0/1$-loss, a symmetric hypothesis class $\mathcal{H}$ and for $\mathcal{Y} = \{-1, +1\}$, we have:

$$\begin{aligned} d_{\mathcal{H}}\left(S_i, S_T\right) &= \sup_{h \in \mathcal{H}} \left| \frac{1}{m_i} \sum_{j=1}^{m_i} \mathbb{1}_{\{h(x_{i,j}) y_{i,j} < 0\}} - \frac{1}{m_T} \sum_{j=1}^{m_T} \mathbb{1}_{\{h(x_{T,j}) y_{T,j} < 0\}} \right| \\ &= \sup_{h \in \mathcal{H}} \left(\frac{1}{m_i} \sum_{j=1}^{m_i} \mathbb{1}_{\{h(x_{i,j}) y_{i,j} < 0\}} - \frac{1}{m_T} \sum_{j=1}^{m_T} \mathbb{1}_{\{h(x_{T,j}) y_{T,j} < 0\}} \right) \\ &= \sup_{h \in \mathcal{H}} \left(1 - \left(\frac{1}{m_i} \sum_{j=1}^{m_i} \mathbb{1}_{\{h(x_{i,j}) \bar{y}_{i,j} < 0\}} + \frac{1}{m_T} \sum_{j=1}^{m_T} \mathbb{1}_{\{h(x_{T,j}) y_{T,j} < 0\}} \right) \right) \\ &= 1 - \inf_{h \in \mathcal{H}} \left(\frac{1}{m_i} \sum_{j=1}^{m_i} \mathbb{1}_{\{h(x_{i,j}) \bar{y}_{i,j} < 0\}} + \frac{1}{m_T} \sum_{j=1}^{m_T} \mathbb{1}_{\{h(x_{T,j}) y_{T,j} < 0\}} \right), \end{aligned} \tag{4.7}$$

where $\bar{y}_{i,j} = 1 - y_{i,j}$ is the flipped label of the j-th data point from the i-th source. Now notice that computing the infimum in equation (4.7) is equivalent to solving a (weighted) empirical risk minimization problem with the input data from the source and the target merged and the labels being the flipped labels from the source and the actual labels from the target.

Therefore, computing the empirical discrepancies in this case is equivalent to solving an empirical risk minimization problem and standard convex upper bounds on the $0-1$ loss can be applied to design computationally efficient approximate algorithms. In our experiments, we solve the ERM problem by using the square loss.

Discussion. While derived from our theoretical results, the minimization procedure for selecting the weights also has an intuitive interpretation. Note that the first term in (4.6) is small whenever large weights are paired with small discrepancies and hence encourages trusting sources that provide data similar to the reference target sample. The second term is small whenever the weights are distributed proportionally to the number of samples per source. Thus, it acts as a form of regularization, by encouraging the usage of information from as many sources as possible.

The hyperparamater λ controls a trade-off between exploiting similar tasks and leveraging information from all sources. As $\lambda \to \infty$, all tasks are assigned weights proportional to the amount of training samples they provide and the model minimizes the empirical risk over all the data, regardless of the quality of the samples. In contrast, as $\lambda \to 0$, the model becomes more sensitive to differences between the source data and the clean reference set, until all weight is assigned to the source closest to the target domain. Assuming that the reference set is included as one of the data sources, these extremes correspond to the naive approach of trusting all sources and training on a merged dataset and not trusting any of them and training on the initial clean data only. In our experiments in Section 4.4 we will see that there is a better operating point between those two extremes. It naturally depends on the actual quality of the available data and our algorithm identifies it successfully.

4.3.3 Learning from private or decentralized data

The described algorithm is straightforward to implement on top of any standard learning procedure, when the data from all N sources is directly available to the learner. We now discuss how we can learn robustly in cases where the sources cannot fully reveal their data. There are many applications where such a situation can arise. For example, this can be due to privacy reasons in the case of medical and biological data or to communication costs and storage limitations in the case of distributed learning [MMR+17].

Here we focus on ways to compute the discrepancies under such constraints. Once this is done, the vector α can be computed easily and then any standard distributed training procedure, e.g. [DCM+12, MMR+17], can be used to obtain the α-weighted empirical loss minimizer. Standard approaches in distributed learning only require the exchange of gradients of minibatches with respect to the current state of the model between the data sources and the central server, so in particular the actual local datasets are never observed by the learner. In cases when the gradients may reveal sensitive information about the data, secure aggregation [BIK+17] or other privacy-preserving distributed learning methods [SS15] can be used on top to ensure privacy.

We distinguish two cases, depending on whether the reference dataset can be shared with the sources.

Case 1: the reference dataset is available to all nodes. If the reference dataset can be shared with the sources without privacy and communication complications, the discrepancies can be estimated *locally on every source, in parallel*. If necessary, the computational protocol can be executed via a trusted computation method [CMM09], for example by using Software Guard Extensions (SGX extensions) [MAA+16], to ensure the correctness of the procedure. The discrepancies alone can then be sent to the learner and the algorithm proceeds as described above. This approach ensures the privacy of the local datasets and allows for all discrepancies to be computed in parallel.

Case 2: the reference dataset can not be shared. In this case the learner can still compute the empirical discrepancies without observing the data from the sources directly, by using a gradient-based optimization procedure. This is because the function inside the supremum in (4.5) decomposes into a term depending only on the reference dataset and a term depending only on the data of the source. Therefore, each discrepancy can be estimated by using a sequence of queries to the source about the gradient of a minibatch from its data with respect to a current candidate for the predictor achieving the supremum.

4.4 Experiments

4.4.1 Method and baselines

We perform two large sets of experiments, following the setup considered in this chapter. We train our algorithm on the data from all sources, including the reference dataset. The hyperparameter λ is selected by 5-fold cross-validation *on the trusted data*. The prediction tasks we consider here are binary classification problems with the $0/1$-loss, so we compute the empirical discrepancies by approximately solving the optimization problem (4.5) as follows. Given the two datasets S_i and S_T, the binary labels of one of them are flipped. The optimization can then be reduced to an empirical risk minimization problem that we solve using a standard convex relaxation approach. We refer to the supplementary material for a more formal description.

We compare the performance of our algorithm to the two naive approaches: training on the reference dataset only (corresponding to $\lambda = 0$ in our algorithm; denoted as *"Reference only"* in the plots and tables) and merging the sources and training on all the data (corresponding to $\lambda \to \infty$; referred to as *"All data"* in the plots and tables). All three methods use linear predictors and are trained by regularized logistic regression. The regularization parameter is always selected by 5-fold cross-validation on the reference data. The learned models are then evaluated on held out test data.

Our aim is to test whether the proposed algorithm successfully leverages information from the sources, while being robust to various perturbations in the distributions of the local datasets, and whether exploiting the multi-source structure of the data gives any improvement over the two standard learning procedures. We also compare the performance of our algorithm to the following robust learning baselines.

Robust aggregation of local models. We consider two recently proposed approaches for robust distributed learning. Following [FXM14], one baseline learns a separate linear model based on each of the source datasets. The final linear predictor is then constructed as the geometric median of these locally learned weight vectors. Another baseline, inspired by [YCKB18], takes the component-wise median instead. Thirdly, based on the locally learned models all N estimates for the probability that a test point belongs to a certain class are computed and the final prediction for the label of that point is obtained by taking the median of these probabilities and thresholding it (referred to as "Median of probs" in the plots and tables). All these baselines aim at learning a robust ensemble of local models.

Robust logistic regression. We use the method of [Pre82], based on the minimization of a Huber-type modification of the logistic loss. Specifically, the method minimizes the following robust loss function, instead of the classic logistic loss:

$$\ell\left(\boldsymbol{w}, \boldsymbol{x}, y\right) = \begin{cases} \log(1+e^{-y\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}}), \text{ if } \log(1+e^{-y\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}}) \leq c \\ 2\sqrt{c\log(1+e^{-y\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}})} - c, \text{ otherwise} \end{cases}$$

In our experiments, we use the recommended threshold value of $c = 1.345^2$, under which the estimate of the linear predictor has been shown to achieve a 95% asymptotic relative efficiency [Pre82]. We also include a regularization term here and learn the regularization parameter by 5-fold cross-validation on the reference data. This baseline is an example of learning robustly on the whole dataset.

Batch normalization. Inspired by the success of *batch normalization* in deep learning [IS15], we compute the mean and standard deviation of the data at each source separately. We then subtract from each data point the mean and divide by the standard deviation of its corresponding dataset. We do the same for the reference data. We then merge all data together and train a logistic regression model with a regularization term. Finally, at test time every input is preprocessed by subtracting the mean and dividing by the standard deviation of the reference dataset, before applying the classifier. This approach aims at increasing robustness to source-specific biases.

4.4.2 Amazon Products data

Our first set of experiments is on the "Multitask dataset of product reviews"[2] [PL17], containing customer reviews for 957 Amazon products from the "Amazon product data" [MPL15, MTSVDH15], together with a binary label indicating whether each review is positive or negative. All reviews in the data set are represented via 25-dimensional feature vectors, obtained by computing a GloVe word embedding [PSM14] and applying the sentence embedding procedure of [ALM17]. We treat the classification of a review as positive or negative as a separate prediction task for each of the products, resulting in a total of 957 input-output distributions.

[2] http://cvml.ist.ac.at/productreviews/

Table 4.1: Results from the experiment on all 957 products.

Algorithm	Mean classification error
Ours	$\mathbf{0.289 \pm 0.0016}$
Reference only	0.301 ± 0.0019
All data	0.312 ± 0.0017
Median of probs.	0.325 ± 0.0021
Geom.median [FXM14]	0.329 ± 0.0021
Comp.median [YCKB18]	0.329 ± 0.0021
Robust loss [Pre82]	0.353 ± 0.0021
Batch norm	0.298 ± 0.0016

As a first, illustrative, experiment, we chose 20 books and 20 other, purposely different, products (e.g. USB drives, mobile apps, meal replacement products). For simplicity, we refer to these additional products as "non-books". Intuitively, when learning to classify book reviews and given access to reviews from both some books and some non-books, a good learning algorithm will be able to leverage all this information, while being robust to the potentially misleading data coming from the less relevant products.

We randomly sample one of the books and 300 positive and 300 negative reviews for it. Out of those, 100 randomly selected reviews are made available to the learner as a reference dataset. The 500 remaining reviews from the product are used for testing. For a given value of $n \in \{0, 1, \ldots, 10\}$ the learner also has access to 100 labeled reviews from each of $10 - n$ other randomly selected books and from each of n randomly selected non-books. Our algorithm, as well as all baselines, are trained on this available data and the learned predictors are evaluated on the test set for the target product. For each n, we repeat this experiment 1000 times.

The results are plotted in Figure 4.1. The x-axis corresponds to the number n of non-books and the y-axis gives the average classification error. The error bars correspond to the standard errors of the mean estimates. We see that our method (green) performs uniformly better than the naive approaches of training on the reference dataset from the target product only (red) and training by merging all data together (blue). When reviews from many books are available, our algorithm is able to use this additional information even better than the model learned on all data. As the proportion of non-books increases, the performance of the second approach degrades, confirming the intuition that the reviews for the non-books provide less useful information for the target task. On the other hand, our algorithm successfully incorporates the information from the useful sources only, converging to the performance of the model learned on the reference data as all additional sources become non-books.

Our algorithm also outperforms all baselines. The batch normalization approach appears to reduce the effect of irrelevant sources, but its performance degrades as $n \to 10$. The median-based approaches perform reasonably when at most half of the sources are non-books, but eventually become worse than the other methods. The component-wise median and the robust loss baselines were excluded from the plot for clarity, as they performed uniformly worse than the other baselines, ranging in average classification error from 0.338 to 0.375 and from 0.348 to 0.372 respectively. Note that the robust loss function of [Pre82] is non-convex, so the poor performance of this baseline is presumably due to failure of the gradient descent optimization procedure to converge to a good local minimum.

Additionally, we performed an experiment on the set of all 957 products. With every product as a prediction task, we randomly selected 100 reviews from it as a reference dataset, leaving 500

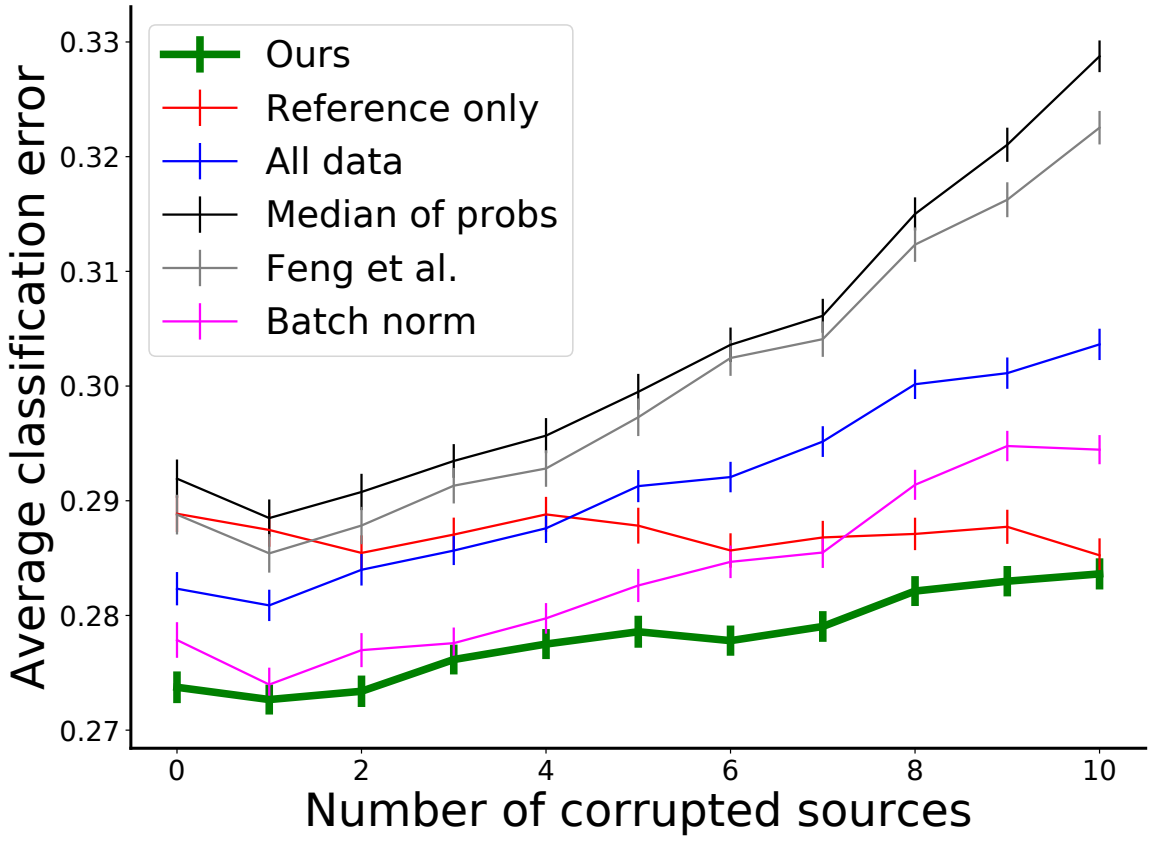


Figure 4.1: Results from the experiments on 20 books and 20 other products from the "Multitask dataset of product reviews". The x-axis gives the number n of non-books in an experiment and the y-axis - the mean classification error. Error bars give the standard error of the estimates.

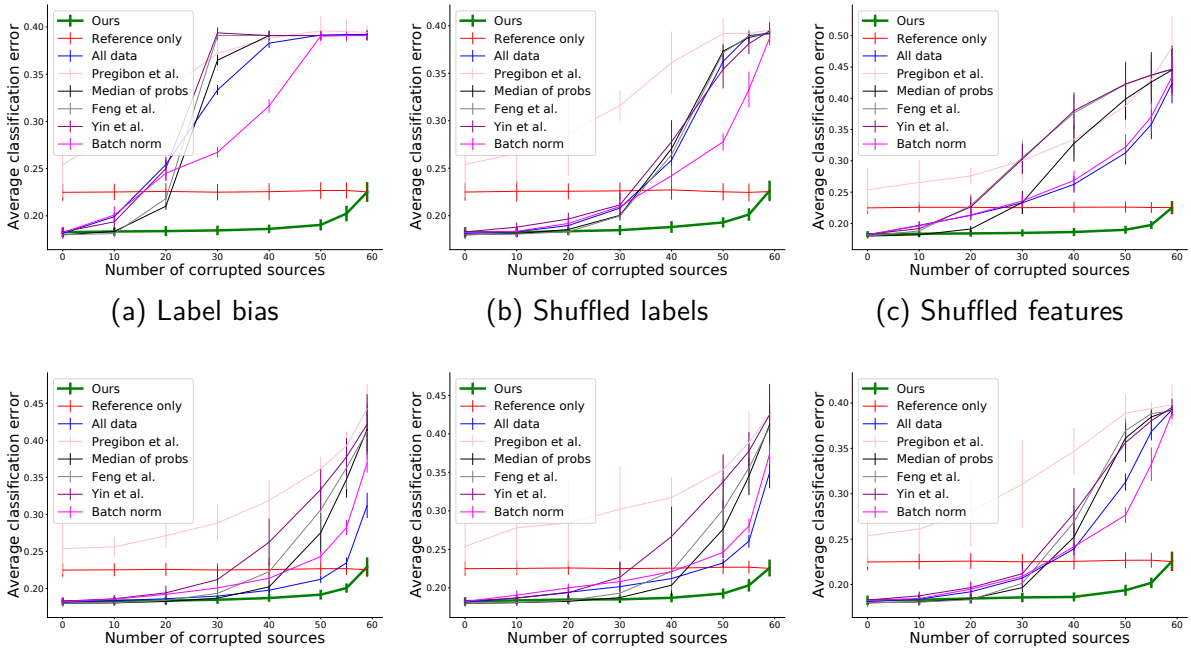
for testing. An additional set of 100 labeled reviews were available from every other product. The algorithms were trained on all available data and evaluated on the test set. The average classification errors achieved by the algorithms are presented in Table 4.1, together with the standard errors of those estimates. We see in particular that our algorithm successfully uses the information from multiple sources to achieve the best overall performance.

4.4.3 Animals with Attributes 2

The Animals with Attributes 2 dataset [XLSA18] contains 37322 images of 50 animal classes. The classes are aligned to 85 binary attributes, e.g. color, habitat and others, via a class-attribute binary matrix, indicating whether an animal possesses each feature. This results in a total of 85 different binary prediction tasks of identifying whether an animal on a given image possesses a certain attribute or not.

Feature representations of the images are obtained via the following procedure. We use a ResNet50 network [HZRS16], pretrained [3] on ImageNet [RDS+15], to obtain feature representations of the ImageNet data and reduce their dimension to 100 by PCA. Finally, for each image in the Animals with Attributes 2 dataset, we compute the ResNet50 feature representation and apply the PCA projection pre-learned on ImageNet.

[3] We use a pretrained model from the TensorNets package, https://github.com/taehoonlee/tensornets.



(a) Label bias (b) Shuffled labels (c) Shuffled features

(d) Blurred images (e) Dead pixels (f) RGB channels swapped

Figure 4.2: Results for the attribute "black" from the Animals with Attributes 2 dataset. Each plot corresponds to a different contamination type. The x-axis gives the number n of corrupted sources and the y-axis gives the average classification error of the algorithms, achieved over 100 different runs. Error bars correspond to the standard deviation around those means.

We perform an independent set of experiments for each attribute and for various types and levels of corruption of the data sources. In each run, we randomly split the data into 60 groups of 500 images, with the remaining 7322 images left out for testing. One of the groups is selected at random as the clean reference dataset available to the learner. The remaining 59 groups correspond to the data sources, some of which provide low-quality or corrupted data. We consider six different types of corruptions. Three act on the labels or the feature representations directly and the next three are synthetic modifications of the images themselves. In the second case, the corresponding images are manipulated before the feature representations are extracted.

Baseline \ n	$n=0$	$n=10$	$n=20$	$n=30$	$n=40$	$n=50$	$n=55$	$n=59$
Reference only	84/1/0	505/5/0	497/13/0	487/23/0	475/35/0	442/68/0	325/185/0	0/510/0
All data	0/85/0	115/395/0	267/243/0	370/140/0	438/72/0	468/42/0	479/31/0	484/26/0
Med of probs.	9/76/0	47/463/0	172/338/0	336/174/0	469/41/0	504/6/0	502/8/0	499/11/0
[FXM14]	8/77/0	32/478/0	110/400/0	338/172/0	457/53/0	504/6/0	502/8/0	497/13/0
[YCKB18]	14/71/0	179/331/0	390/120/0	432/78/0	472/38/0	502/8/0	503/7/0	497/13/0
[Pre82]	55/30/0	308/202/0	361/149/0	416/94/0	437/73/0	455/55/0	470/40/0	485/25/0
Batch norm	0/85/0	107/403/0	317/193/0	416/94/0	446/63/1	478/32/0	487/23/0	482/28/0

Table 4.2: Summary of the results from the Animals with Attributes 2 experiments, over all 85 prediction tasks and all 6 types of corruption. Given a number of corrupted sources n (columns) and a baseline (rows), we report values in the form A/B/C, where A is the number of times that our method performed significantly better than the corresponding baseline, B is the number of times it performed equally well and C is the number of times it performed significantly worse, summed over the various types of corruptions and all attributes. More details are provided in the main body of the text.

- Label bias: The labels of all (corrupted) samples are switched to class 1.
- Shuffled labels: The labels of all samples are shuffled randomly, separately in each corrupted source.
- Shuffled features: Given a permutation of the indexes between 1 and 100, the features of all samples are shuffled according to it.
- Blurred images: Each image is blurred by filtering with a Gaussian kernel with standard deviation $\sigma = 6$.
- Dead pixels: In each image a random 30% of the pixels are set to pure black or white.
- RGB channels swapped: The values in the red and the blue color channels of each image are swapped.

Given an attribute, a type of corruption and a value of $n \in \{0, 10, 20, 30, 40, 50, 55, 59\}$, the data is split randomly, as described above, and the samples of n randomly chosen sources are corrupted. Our algorithm, as well as all baselines, then learn a model based on the resulting data and the performance of the obtained predictors is evaluated on the test data. For any combination of target attribute, corruption strategy and value of n, the experiment is repeated 100 times with a different random seed to obtain error estimates.

The results for the first attribute from the Animals with Attributes 2 data ("black") are given in Figure 4.2. Each plot corresponds to a different type of contamination. The x-axis gives the number of sources providing corrupted data and the y-axis corresponds to the average error that an algorithm achieved on the test set, over the 100 runs for each experimental setup. The error bars give the standard deviation around this average.

Our algorithm (green) performs at least as well as or strictly better than all baselines, for *any* type of corruption and *any* proportion of corrupted sources. When all sources provide clean data, the performance of our method matches the one of the classic regularized logistic regression approach on i.i.d. data (blue). As the number of corrupted sources increases, the performance of all baselines gradually degrades, while our algorithm is able to leverage the remaining clean data and suppress the effect of the corruptions. The median-based baselines perform reasonably when less than half of the sources are corrupted, but fail for larger proportions. The robust logistic regression baseline performs poorly, again likely due to the non-convexity of the loss function. As all sources become unreliable, our method performs as well as the approach of learning from the reference dataset only, which is indeed optimal since all other data is corrupted.

We summarize the results from all attributes in Table 4.2. For any number of corrupted sources n (columns), we compare our method to the performance of each baseline (rows). We report values in the form A/B/C, where A is the number of times that our method performed significantly better than the corresponding baseline, B is the number of times it performed equally well and C is the number of times it performed significantly worse, summed over the various types of corruptions and all attributes. For a fixed type of corruption and attribute, we say that one method performs significantly better than another over the set of 100 runs with this setup, if the difference in the average performance of the two models is larger than the sum of the standard deviations around those means (that is, if the error bars, as in Figure 4.2, do not intersect).

The results in Table 4.2 show that our method *performs significantly better than all baselines for many types of corruption and many values of* n, especially for high levels of contamination, while *essentially never performing significantly worse than any baseline*.

In addition, we performed experiments in which a proportion p of the samples in the n corrupted sources are modified (instead of all of them). Apart from $p = 1$, we experimented with $p = 0.5$ and $p = 0.2$. We present the same table of results for these cases in Table 4.3 and 4.4 respectively.

Table 4.3: Summary of the results for $p = 0.5$, over all 85 prediction tasks and all corruptions.

Baseline \ n	$n = 0$	$n = 10$	$n = 20$	$n = 30$	$n = 40$	$n = 50$	$n = 55$	$n = 59$
Reference only	84/1/0	508/2/0	501/9/0	488/22/0	471/39/0	424/86/0	303/207/0	156/354/0
All data	0/85/0	0/510/0	82/428/0	158/352/0	215/295/0	241/269/0	223/287/0	168/342/0
Median of probs.	9/76/0	30/480/0	53/457/0	93/417/0	189/321/0	272/238/0	252/258/0	216/294/0
[FXM14]	8/77/0	28/482/0	19/491/0	84/426/0	172/338/0	254/256/0	253/257/0	217/293/0
[YCKB18]	14/71/0	123/387/0	227/283/0	155/355/0	247/259/4	295/215/0	282/228/0	224/286/0
[Pre82]	55/30/0	287/223/0	282/228/0	329/181/0	350/160/0	358/152/0	374/136/0	367/143/0
Batch norm	0/85/0	2/508/0	78/432/0	139/370/1	183/326/1	186/323/1	155/354/1	97/412/1

Table 4.4: Summary of the results for $p = 0.2$, over all 85 prediction tasks and all corruptions.

Baseline \ n	$n = 0$	$n = 10$	$n = 20$	$n = 30$	$n = 40$	$n = 50$	$n = 55$	$n = 59$
Reference only	84/1/0	507/3/0	505/5/0	504/6/0	492/18/0	459/51/0	429/81/0	404/106/0
All data	0/85/0	0/510/0	0/510/0	0/510/0	0/510/0	1/509/0	2/508/0	1/509/0
Median of probs.	9/76/0	28/482/0	21/489/0	16/494/0	17/493/0	28/482/0	30/478/2	31/479/0
[FXM14]	8/77/0	30/480/0	24/486/0	16/494/0	16/494/0	20/489/1	23/485/2	26/484/0
[YCKB18]	14/71/0	95/415/0	146/364/0	34/476/0	42/468/0	39/471/0	36/474/0	40/470/0
[Pre82]	55/30/0	282/228/0	282/228/0	275/235/0	287/223/0	264/246/0	281/229/0	267/243/0
Batch norm	0/85/0	0/510/0	0/510/0	0/510/0	0/510/0	0/509/1	0/509/1	1/508/1

4.5 Summary

We introduced an algorithm for learning from data provided by multiple untrusted sources. It incorporates information from all of them, while being robust to arbitrary corruptions and manipulations of the data. By making use of the grouped structure of the task and a reference dataset, the method is able to successfully learn even if more than half of the available data is corrupted or uninformative. Our method is theoretically justified and easy to implement, even in cases when the data is decentralized and/or private. We demonstrated its effectiveness through two sets of extensive experiments, showing its superior performance to all baselines, for various levels and types of corruption.

CHAPTER 5

Fairness-Aware PAC Learning from Corrupted Data

We now turn to another aspect of learning from corrupted data that is especially relevant from a present perspective - that of studying whether it is possible to ensure that in addition to accuracy, the fairness of the learned classifier is also protected from malicious data effects. Here we study the problem from a PAC learning perspective, in the single dataset scenario.

5.1 Motivation and outline

As explained in Section 2.3 many ways of measuring and optimizing the fairness of learned models have been developed. The problem is perhaps best studied in the context of group fairness in classification, where the decisions of a binary classifier have to be nondiscriminatory with respect to a certain protected attribute, such as gender or race [BHN19]. This is typically done by formulating a desirable fairness property for the task at hand and then optimizing for this property, alongside with accuracy, be it via a data preprocessing step, a modification of the training procedure, or by post-processing of a learned classifier on held-out data [MMS+19]. The underlying assumption is that by ensuring that the fairness property holds exactly or approximately based on the available data, one obtains a classifier whose decisions will also be fair at prediction time.

A major drawback of this framework is that for many real-world applications in which fairness can be a concern the training and validation data available are often times unreliable and biased [BR18, MMS+19]. For example, demographic data collected via surveys or online polls is often difficult and expensive to verify. More generally any human-generated data is likely to contain various historical biases. Datasets collected via crowdsourcing or web crawing are also prone to both unwittingly created errors and conscious or even adversarially created biases.

These issues naturally raise concerns about the current practice of training and certifying fair models on such datasets. In fact, recent work has demonstrated empirically that strong poisoning attacks can negatively impact the fairness of *specific learners* based on loss minimization. At the same time, little is known about the fundamental limits of fairness-aware learning from corrupted data. Previous work has only partially addressed the problem by studying weak data corruption models, for example by making specific label/attribute noise assumptions. However, these assumptions do not cover all possible (often unknown) problems that real-world data

can possess. More generally, in order to avoid a cat-and-mouse game of designing defenses and attacks for fair machine learning models, one would need to be able to *certify fairness* as a property that holds when training under arbitrary, even adversarial, manipulations of the training data [KL93].

Outline In this chapter, we address the aforementioned issues by studying the effect of arbitrary data corruptions on fair learning algorithms. Specifically, we explore the fundamental limits of fairness-aware PAC learning within the *malicious adversary model* of [Val85]. We focus on binary classification with a binary protected attribute and on the demographic parity [CKP09] and equal opportunity [HPS16] fairness notions.

First we show that learning under this adversarial model is provably impossible in a PAC sense - there is *no learning algorithm that can ensure convergence with high probability to a point on the accuracy-fairness Pareto front* on the set of all finite hypothesis spaces, even in the limit of infinite training data. Furthermore, the irreducible excess gap in the fairness measures we study is inversely proportional to the frequency of the rarer of the two protected attributes groups. This makes the robust learning problem especially hard when one of the protected subgroups in the data is underrepresented. These hardness results hold for *any learning algorithm* based on a corrupted dataset, including pre-, in- and post-processing methods in particular.

Perhaps an even more concerning result from a practical perspective is that the adversary can also ensure that any learning algorithm will output a classifier that is *optimal in terms of accuracy, but exhibits a large amount of unfairness*. The bias of such a classifier might go unnoticed for a long time in production systems, especially in applications where sensitive attributes are not revealed to the system at prediction time for privacy reasons.

We also show that our hardness results are tight up to constant factors, in terms of the corruption ratio and the protected group frequencies, by proving matching upper bounds. To this end we study the performance of two natural types of learning algorithms under the malicious adversary model. We show that both algorithms achieve order-optimal performance in the infinite data regime, *thereby providing tight upper and lower bounds on the irreducible error of fairness-aware statistical learning under adversarial data corruption.*

We conclude with a discussion on the implications of our hardness results, emphasizing the need for developing and studying further data corruption models for fairness-aware learning, as well as on the importance of strict data collection practices in the context of fair machine learning.

5.2 Related work

To the best of our knowledge, we are the first to investigate the information-theoretic limits of fairness-aware learning against a malicious adversary. There is, however, related previous work on PAC learning analysis of fair algorithms, robust fair learning, and learning with poisoned training data, that we discuss in this section.

Fairness in classification Fairness-aware learning has been widely studied in the context of classification: we refer to our discussion in Section 2.3 for a brief overview and to [BHN19] for an exhaustive introduction to the field. On the methodological side, our upper bounds analysis employs a technique for proving concentration of estimates of conditional probabilities that has previously been used in the context of group fairness by [WGOS17] and [ABD$^+$18]. A

number of hardness results for fair learning are also known. In particular, [KMR17] prove the incompatability of three fairness notions for a broad class of learning problems and [MW18b] quantify fundamental trade-offs between fairness and accuracy. Both of these works, however, focus on learning with i.i.d. clean data.

Fairness and data corruption Most relevant for our setup are a number of recent works that empirically study attacks and defenses on fair learners under adversarial data poisoning. In particular, [SBC20], [CNM+20] and [MNMG20] consider practical, gradient-based poisoning attacks against machine learning algorithms. All of these works demonstrate empirically that poisoned data can severely damage the performance of fair learners that are based on empirical loss minimization. In our work we go beyond this by proving a set of hardness results that hold for *arbitrary learning algorithms*. On the defense side, [RLWS20] design and empirically study an adversarial training approach for dealing with data corruption when training fair models. Their defense is shown to be effective against specific poisoning attacks that aim to reduce the model accuracy. In contrast, for our upper bounds we are interested in learners that provably work against any poisoning attack, including those that can target the fairness properties of the model as well.

Among works focusing on weaker adversarial models, a particularly popular topic is the one of fair learning with noisy or adversarially perturbed protected attributes [LZMV19, AKM20, WGN+20, CHKV21, MC21, CMV21]. Under the explicit assumption that the corruption does not affect the inputs and the labels, these works propose algorithms that can recover a fair model despite the data corruption. A related, but conceptually different topic is the one of fair learning without demographic information [HSNL18, KMZ20, MOS20, LBC+20]. Another commonly assumed type of corruption is label noise, which is shown to be overcomable under various assumptions by [DADC18], [JN20], [WLL20] and [FGC20]. A distributionally robust approach for certifying fairness is taken by [TNKB20], under the assumption that the real data distribution falls within a Wasserstein ball centered at the empirical data distribution. In [ICS+20] a formal methods framework for certifying fairness through unawareness, even in the presence of a specific type of data bias that targets their desired fairness measure, is provided. The vulnerability of fair learning algorithms to specific types of data corruption has also been demonstrated on real-world data by [CŽ13] and [KZ18].

An orthogonal line of work shows that imposing fairness constraints can neutralize the effects of corrupted data, under specific assumptions on the type of bias present [BS20]. Also related are the works of [TRO+19] and [LBSS21] who propose procedures for data cleaning/outlier detection, without a specific adversarial model, that in particular improve fairness performance.

Learning against an adversary Learning from corrupted training data is a field with long history, where both the theoretical and the practical aspects of attacking and defending ML models have been widely studied [Tuk60, Hub64, AL88, KL93, CBDF+99, BEK02, CS04, BNL12a, CSV17, SKL17, CLL+17, DKK+19b]. In this work we study fair learning within the malicious adversary model [Val85, KL93, CBDF+99]. This chapter adds an additional dimension to this line of work, where fairness is considered alongside with accuracy as an objective for the learner.

5.3 Preliminaries

In this section we formalize the problem of fairness-aware learning against a malicious adversary, by giving precise definitions of the learning objectives and the studied data corruption model.

5.3.1 Fairness-aware learning

Throughout the chapter we adopt the standard group fairness classification framework, as introduced in Section 2.3. We consider a product space $\mathcal{X} \times \mathcal{A} \times \mathcal{Y}$, where $\mathcal{X}$ is an input space, $\mathcal{Y} = \{0, 1\}$ is a binary label space and $\mathcal{A} = \{0, 1\}$ is a set corresponding to a binary protected attribute (for example, being part of a majority/minority group). We assume that there is an unknown true data distribution $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ from which the clean data is sampled. Denote by $\mathcal{H} \subseteq \{h : \mathcal{X} \to \mathcal{Y}\}$ the hypothesis space of all classifiers to be considered. We denote the random variables corresponding to randomly sampled inputs, labels and protected attributes as X, Y and A respectively.

PAC learning Adopting a statistical PAC learning setup, we are interested in designing learning procedures that find a classifier based on training examples. Formally, a (statistical) fairness-aware learner $\mathcal{L} : \cup_{n \in \mathbb{N}} (\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$ is a function that takes a labeled dataset of an arbitrary size and outputs a hypothesis. Note that we again consider learning in the purely statistical sense here, focusing on *any* procedure that outputs a hypothesis, regardless of its computational complexity, and seeking learners that are sample-efficient instead.

In a clean data setup, the learner is trained on a dataset $S^c = \{(x_i^c, a_i^c, y_i^c)\}_{i=1}^n$ sampled i.i.d. from $\mathbb{P}$ and outputs a hypothesis $h \coloneqq \mathcal{L}(S^c)$. The performance of a learner can be measured via the expected $0/1$ loss (a.k.a. the risk) with respect to the distribution $\mathbb{P}$

$$\mathcal{R}(h, \mathbb{P}) = \mathbb{P}(h(X) \neq Y). \tag{5.1}$$

Group fairness in classification In (group) fairness-aware learning, an additional desirable property of the classifier $h = \mathcal{L}(S^c)$ is that its decisions are fair in the sense that it does not exhibit discrimination with respect to one of the protected subgroups in the population. Many different formal notions of group fairness have previously been proposed in the literature. Here we consider two of the group fairness measures we discussed in Section 2.3.3, namely demographic parity and equality of opportunity. Recall that *demographic parity* [CKP09], requires that the decisions of the classifier are independent of the protected attribute, that is

$$\mathbb{P}(h(X) = 1 | A = 0) = \mathbb{P}(h(X) = 1 | A = 1). \tag{5.2}$$

The second notion, *equality of opportunity* [HPS16], states that the true positive rates of the classifier should be equal across the protected groups, that is

$$\mathbb{P}(h(X) = 1 | A = 0, Y = 1) = \mathbb{P}(h(X) = 1 | A = 1, Y = 1). \tag{5.3}$$

In this definition, an implicit assumption is that $Y = 1$ corresponds to a beneficial outcome (for example, an applicant receiving a job), so that this fairness notion only considers instances where the correct outcome should be advantageous.

In practice, it is rarely the case that a classifier achieves perfect fairness. Therefore, we will instead be interested in controlling the *amount of unfairness* that h possesses, measured via

corresponding fairness deviation measures $\Gamma(h)$ [WGOS17, MW18a, WM19]. Here we adopt the *mean difference score* measure of [CV10] and [MW18a] for demographic parity

$$\Gamma^{DP}(h, \mathbb{P}) = \Big|\mathbb{P}(h(X) = 1|A = 0) - \mathbb{P}(h(X) = 1|A = 1)\Big| \tag{5.4}$$

and its analog for equality of opportunity

$$\Gamma^{EOp}(h, \mathbb{P}) = \Big|\mathbb{P}(h(X) = 1|A = 0, Y = 1) - \mathbb{P}(h(X) = 1|A = 1, Y = 1)\Big|. \tag{5.5}$$

To avoid degenerate cases for these measures, we assume throughout the chapter that $P_a = \mathbb{P}(A = a) > 0$ and $P_{1a} = \mathbb{P}(Y = 1, A = a) > 0$ for both $a \in \{0, 1\}$. For the rest of the chapter, whenever we are interested in demographic parity fairness, we assume without loss of generality that $A = 0$ is the minority class, so that $P_0 \leq \frac{1}{2} \leq P_1$. Similarly, whenever the fairness notion is equality of opportunity, we will assume that $P_{10} \leq P_{11}$.

Whenever the underlying distribution is clear from the context, we will drop the dependence of $\mathcal{R}(h, \mathbb{P})$ and $\Gamma(h, \mathbb{P})$ on $\mathbb{P}$ and simply write $\mathcal{R}(h)$ and $\Gamma(h)$.

5.3.2 Learning against an adversary

As argued previously, machine learning models are often trained on unreliable datasets, where some of the points might be corrupted by noise, human biases and/or malicious agents. To model arbitrary manipulations of the data, we assume the presence of an adversary that can modify a certain fraction of the dataset and study fair learning in this context. In addition to not being partial to a specific type of data corruption, this worst-case approach has the advantage of providing a *certificate for fairness*: if a system can work against a strong adversarial model, it will be effective under *any circumstances that are covered by the model*.

Similarly to the classic robust machine learning setup, a *fairness-aware adversary* is any procedure for manipulating a dataset, that is a *possibly randomized function*

$$\mathfrak{A} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n$$

that takes in a clean dataset $S^c = \{(x_i^c, a_i^c, y_i^c)\}_{i=1}^n$ sampled i.i.d. from $\mathbb{P}$ and outputs a new, corrupted, dataset $S^p = \{(x_i^p, a_i^p, y_i^p)\}_{i=1}^n$ *of the same size*. Depending on the type of restrictions that are imposed on the adversary, various adversarial models can be obtained.

In this chapter we focus on the *malicious adversary model*, as introduced in Section 2.2.3, and adapt it to the fairness-aware learning case. The formal data generating procedure is as follows:

- An i.i.d. *clean dataset* $S^c = \{(x_i^c, a_i^c, y_i^c)\}_{i=1}^n$ is sampled from $\mathbb{P}$.
- Each index/point $i \in \{1, 2, \ldots, n\}$ is *marked* independently with probability α, for a fixed constant $\alpha \in [0, 0.5)$. Denote all marked indexes by $\mathfrak{P} \subseteq [n]$.
- The *malicious adversary* computes, in a possibly randomized manner, a corrupted dataset $S^p = \{(x_i^p, a_i^p, y_i^p)\}_{i=1}^n \in (\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n$, with the only restriction that $(x_i^p, a_i^p, y_i^p) = (x_i^c, a_i^c, y_i^c)$ for all $i \notin \mathfrak{P}$. That is, the adversary can replace all marked data points in an arbitrary manner, with *no assumptions whatsoever* about the points (x_i^p, a_i^p, y_i^p) for $i \in \mathfrak{P}$.

- The corrupted dataset S^p is then passed on to the learner, who computes $\mathcal{L}(S^p)$.

For a fixed $\alpha \in [0, 0.5)$, we say that $\mathfrak{A}$ is a malicious adversary of power α. Note that the number of marked points is $|\mathfrak{P}| \sim Bin(n, \alpha)$.

Since no assumptions are made on the corrupted data points, they can, in particular, depend on the learner $\mathcal{L}$, the data distribution $\mathbb{P}$, the clean data S^c and all other parameters of the learning problem. That is, the adversary acts with full knowledge of the learning setup and without any computational constraints, which is in lines with our worst-case approach. Note that this is in contrast to the learner $\mathcal{L}$ that can only access the data points in S^p. We refer to Section 5.3.4 for a more formal treatment.

5.3.3 Multi-objective learning

Our goal is to study the performance of the classifier $\mathcal{L}(S^p)$ *learned on the corrupted data*, both in terms of its expected loss $\mathcal{R}(\mathcal{L}(S^p), \mathbb{P})$ and its fairness deviation $\Gamma(\mathcal{L}(S^p), \mathbb{P})$ *on the clean (test) distribution* $\mathbb{P}$. We will be interested in the probabilities of these quantities being large or small, under the randomness of the sampling of S^p - that is the randomness of the clean data, the marked points and the adversary.

Note that it is not a priori clear how to trade-off the two metrics and that this is likely to be application-dependent. Therefore it is also unclear how to evaluate the quality of a hypothesis. Here we study two possible ways to do so.

Weighted objective One approach is to assume that a (application dependent) trade-off parameter λ is predetermined, so that the learner has to approximately minimize

$$L_\lambda(h) = \mathcal{R}(h) + \lambda\Gamma(h). \tag{5.6}$$

In particular, the quality of the hypothesis $\mathcal{L}(h^S)$ can be directly measured via $L_\lambda(\mathcal{L}(h^S)) - \min_{h\in\mathcal{H}} L_\lambda(h)$. We will use L_λ^{DP} and L_λ^{EOp} to denote the weighted objectives with Γ^{DP} and Γ^{EOp} respectively.

Element-wise comparisons Alternatively, one may want to consider the two objectives independently. Given a classifier $h \in \mathcal{H}$, denote by $\mathfrak{V}(h) = (\mathcal{R}(h), \Gamma(h))$ the vector consisting of the values of the two objectives. Note that $\mathfrak{V}$ does not, in general, induce a total order on $\mathcal{H}$. Instead we can only compare two classifiers $h_1, h_2 \in \mathcal{H}$ if, say, h_1 dominates h_2 in the sense that both $\mathcal{R}(h_1) \leq \mathcal{R}(h_2)$ and $\Gamma(h_1) \leq \Gamma(h_2)$. We denote this relation by $\mathfrak{V}(h_1) \preceq \mathfrak{V}(h_2)$. As we will see, these component-wise comparisons are still useful for understanding the limits of learning against an adversary.

To be able to measure how far a classifier is from optimal under the $\preceq$ relation, it is natural to consider situations where there exists a classifier that is optimal both in terms of fairness and accuracy. Then the quality of any other hypothesis can be measured with respect to this optimal classifier. That is, one may assume that there exists a $h^* \in \mathcal{H}$, such that $h^* \in \operatorname{argmin}_{h\in\mathcal{H}} \mathcal{R}(h)$ and $h^* \in \operatorname{argmin}_{h\in\mathcal{H}} \Gamma(h)$, so that $\mathfrak{V}(h^*) \preceq \mathfrak{V}(h)$ for all $h \in \mathcal{H}$. Then the quality of $\mathcal{L}(S^p)$ can be measured as the $\mathbb{R}^2$ vector

$$\mathbf{L}(\mathcal{L}(S^p)) = \mathfrak{V}(\mathcal{L}(S^p)) - \mathfrak{V}(h^*). \tag{5.7}$$

As with the weighted objective, we use $\mathbf{L}^{DP}(\mathcal{L}(S^p))$ and $\mathbf{L}^{EOp}(\mathcal{L}(S^p))$ to denote the loss vector when demographic parity and equality of opportunity are used respectively.

One particular situation that we will study in which a component-wise optimal classifier h^* exists, is within the realizable PAC learning model with equality of opportunity fairness. Indeed, whenever a classifier $h^* \in \mathcal{H}$ satisfies $\mathbb{P}(h^*(X) = Y) = 1$, we have that both $\mathcal{R}(h^*) = 0$ and $\Gamma^{EOp}(h^*) = 0$ and so $\mathbf{L}^{EOp}(\mathcal{L}(S^p)) = \mathfrak{V}^{EOp}(\mathcal{L}(S^p))$.

5.3.4 The limits of fairness-aware learning against an adversary

Lower and upper bounds analysis Over the next sections we will be showing lower and upper bounds on $L_\lambda(\mathcal{L}(S^p))$ and $\mathbf{L}(\mathcal{L}(S^p))$, that is, the risk and the fairness deviation measure achieved by the learner when trained on the corrupted data. *Our lower bounds* can be thought of as hardness results that describe a limit on how well the learner can perform against the adversary. These are based on explicit constructions of hard learning problems and adversaries that demonstrate these limitations. *Our upper bounds* complement the hardness results by constructing learners that recover a classifier with guarantees on fairness and accuracy that match the lower bounds, for a wide range of learning problems and adversaries.

As discussed in Section 2.2.2, crucial in these results is the ordering of the quantifiers. These matter not only for the comparison between the upper and the lower bounds, but also for the sake of formalizing the powers of the adversary and the learner. Recall that the learner only operates with knowledge of the corrupted dataset. At the same time, the adversary is assumed to know not only the clean data, but also the target distribution and the learner. Therefore, our lower bounds are structured as follows:

For any learner $\mathcal{L}$ there exists a distribution $\mathbb{P}$ and an adversary $\mathfrak{A}$,
such that with constant probability ...

Note in particular that the adversary can be chosen after the learner is constructed and together with the distribution and it can therefore be tailored to their choice. At the same time, our upper bounds read as:

There exists a learner $\mathcal{L}$, such that for any distribution $\mathbb{P}$, any adversary $\mathfrak{A}$ and any $\delta \in (0, 1)$,
with probability at least $1 - \delta$ ...

Since the learner is fixed before the distribution and the adversary are, it has to work for any such pair.

We note that all probability statements refer to the randomness in the full generation process of the dataset S^p, that is the randomness of the clean data, the marked points and the adversary. For a fixed clean data distribution $\mathbb{P}$ and a fixed adversary $\mathfrak{A}$, we denote the distribution of S^p as $\mathbb{P}^{\mathfrak{A}}$.

Role of the hypothesis space Learnability in our setup can be studied either as a property of any fixed hypothesis space, or as a property of a class of hypothesis spaces, for example the hypothesis spaces of finite size or finite VC dimension. However, one can easily see that for certain hypothesis spaces fairness can be satisfied trivially. For example, whenever $\mathcal{H}$ contains a classifier that is constant on the whole input space (that is, always predicts 1 or always predicts 0), a learner that returns this constant classifier, regardless of the observed data, will always be perfectly fair with respect to both fairness notions, under any distribution

and against any adversary. We therefore opt to study the learnability of *classes of hypothesis spaces.*

In particular, our hardness results demonstrate the *existence of a finite hypothesis space*, such that a certain amount of excess inaccuracy and/or unfairness is unavoidable. Therefore, no learner can achieve better guarantees on the class of all finite hypothesis spaces, even in the infinite training data limit. This is contrast to, for example, classic PAC learning with clean data, where the ERM algorithm is a PAC learner for all finite hypothesis spaces and more generally all spaces of finite VC dimension [SSBD14].

On the other hand, the learners we construct for the upper bounds are shown to work for *any hypothesis space* that is finite or of finite VC dimension, in all cases matching the lower bounds.

Parameters of the learning problem Our bounds will depend explicitly on the corruption ratio α and on the smaller of the protected class frequencies $P_0 = \mathbb{P}(A = 0)$ (for demographic parity) or on $P_{10} = \mathbb{P}(Y = 1, A = 0) \leq \mathbb{P}(Y = 1, A = 1)$ (for equality of opportunity). To understand the limits of fairness-aware learning against a malicious adversary, we will analyze our bounds for small values of α and P_0 or P_{10}. Intuitively, the smaller the corruption rate α is, the easier it is for the learner to recover an accurate and fair hypothesis. On the other hand, a small value for P_0 or P_{10} implies that one of the subgroups is underrepresented in the population, and so intuitively the adversary can hide a lot of information about this group and thus prevent the learner from finding a fair hypothesis.

As we will see, this intuition is reflected in our bounds, which give a tool for understanding the effect of these quantities on the hardness of the learning problem. Comparing the lower bounds, which hold regardless of the sample size n, to the upper bounds in the limit of $n \to \infty$ allows us to reason about the absolute limits of fairness-aware learning against a malicious adversary. Indeed, in this large data limit, we find that our upper and lower bounds match in terms of their dependence on α and P_0 or P_{10} up to constant factors. We note that designing algorithms that achieve *sample-optimal* guarantees in our context is beyond the scope of this work. However, we will also be interested in the *statistical rates of convergence* of the studied learners to the irreducible gap certified by the lower bounds. We refer to Section 5.5.2 for a formal treatment.

5.4 Lower bounds

We now present a series of hardness results that demonstrate that fair learning in the presence of a malicious adversary is provably impossible in a PAC learning sense. **Complete proofs of all results in this section can be found in Appendix C.1.**

5.4.1 Pareto lower bounds

We begin by presenting two hardness results that intuitively show that for some hypothesis spaces $\mathcal{H}$ the adversary can prevent any learner from reaching the Pareto front of the accuracy-fairness optimization problem. We first demonstrate this for demographic parity:

Theorem 8. *Let $0 \leq \alpha < 0.5, 0 < P_0 \leq 0.5$. For any input set $\mathcal{X}$ with at least four distinct points, there exists a finite hypothesis space $\mathcal{H}$, such that for any learning algorithm $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$, there exists a distribution $\mathbb{P}$ for which $\mathbb{P}(A = 0) = P_0$, a*

malicious adversary $\mathfrak{A}$ of power α and a hypothesis $h^* \in \mathcal{H}$, such that with probability at least 0.5

$$\mathcal{R}(\mathcal{L}(S^p), \mathbb{P}) - \mathcal{R}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{1-\alpha}, 2P_0P_1\right\}$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}) - \Gamma^{DP}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{2P_0P_1(1-\alpha)}, 1\right\}.$$

The proof of this theorem (as well as of the other hardness results presented in this section) is based on the so-called *method of induced distributions*, pioneered by [KL93]. The idea is to construct two distributions that are sufficiently different, so that different classifiers perform well on each, yet can be made indistinguishable after the modifications of the adversary. Then no fixed learner with access only to the corrupted data can be "correct" with high probability on both distributions and so any learner will incur an excessively high loss and exhibit excessively high unfairness on at least one of them, regardless of the amount of available data.

Here we provide a sketch proof of Theorem 8, to illustrate the type of construction used. A complete proof can be found in Appendix C.1.

Proof. **(Sketch)** Let $\eta = \frac{\alpha}{1-\alpha}$, so that $\alpha = \frac{\eta}{1+\eta}$. We assume here that $\eta = \frac{\alpha}{1-\alpha} \leq 2P_0(1-P_0)$, with the other case following from a similar construction, but with an adversary that uses a smaller value of α (so that it leaves some of the data points at its disposal untouched).

Take four distinct points $\{x_1, x_2, x_3, x_4\} \in \mathcal{X}$. We consider two distributions $\mathbb{P}_0$ and $\mathbb{P}_1$, where each $\mathbb{P}_i$ is defined as

$$\mathbb{P}_i(x, a, y) = \begin{cases} 1 - P_0 - \eta/2 & \text{if } x = x_1, a = 1, y = 1 \\ P_0 - \eta/2 & \text{if } x = x_2, a = 0, y = 0 \\ \eta/2 & \text{if } x = x_3, a = i, y = \neg i \\ \eta/2 & \text{if } x = x_4, a = \neg i, y = i \\ 0 & \text{otherwise} \end{cases}$$

Note that these are valid distributions, since $\eta \leq 2P_0(1-P_0) \leq 2P_0 \leq 2(1-P_0)$ by assumption and also that $P_0 = \mathbb{P}_i(A = 0)$ for both $i \in \{0, 1\}$. Consider the hypothesis space $\mathcal{H} = \{h_0, h_1\}$, with

$$h_0(x_1) = 1 \quad h_0(x_2) = 0 \quad h_0(x_3) = 1 \quad h_0(x_4) = 0$$

and

$$h_1(x_1) = 1 \quad h_1(x_2) = 0 \quad h_1(x_3) = 0 \quad h_1(x_4) = 1.$$

The point of this construction is as follows: there are only two points, x_3 and x_4, where the two distributions differ. This is also where the classifiers differ and, in fact, each classifier h_i is better performing on the distribution $\mathbb{P}_i$, in both accuracy and fairness, than the other classifier.

Indeed, it is easy to verify that

$$L(h_{\neg i}, \mathbb{P}_i) - L(h_i, \mathbb{P}_i) = \eta, \quad \text{for both } i = 0, 1. \tag{5.8}$$

Moreover,

$$\Gamma^{DP}(h_{\neg i}, \mathbb{P}_i) - \Gamma^{DP}(h_i, \mathbb{P}_i) = \frac{\eta}{2P_0(1-P_0)}, \quad \text{for both } i = 0, 1. \tag{5.9}$$

Now what the adversary does is to use all of the marked data to insert points with inputs x_3 and x_4, but with flipped labels and protected attributes. Then, since the points with inputs x_3 and x_4 in the original data are sufficiently rare, the adversary manages to hide which one of the two distributions was the original one.

Specifically, consider a (randomized) malicious adversary $\mathfrak{A}_i$ of power α, that given a clean distribution $\mathbb{P}_i$, changes every marked point to $(x_3, \neg i, i)$ with probability 0.5 and to $(x_4, i, \neg i)$ otherwise. Under a distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, the probability of seeing a point $(x_3, i, \neg i)$ is $\frac{\eta}{2}(1-\alpha) = \frac{\eta}{2}\frac{1}{1+\eta} = \alpha/2$, which is equal to the probability of seeing a point $(x_3, \neg i, i)$. Therefore, denoting the probability distribution of the corrupted dataset, under a clean distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, by $\mathbb{P}'_i$, one can verify that $\mathbb{P}'_0 = \mathbb{P}'_1$, so the two initial distributions $\mathbb{P}_0$ and $\mathbb{P}_1$ become indistinguishable under the adversarial manipulation.

The proof concludes by formalizing the observation that any fixed learner $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \{h_0, h_1\}$ will perform poorly on at least one of the distribution-adversary pairs $(\mathbb{P}_i, \mathfrak{A}_i)$, since the resulting corrupted data distributions are the same, but the optimal classifiers differ. □

Discussion Our hardness result implies that no learner can guarantee reaching a point on the Pareto front in a PAC learning sense, even for a simple family of hypothesis spaces, namely the finite ones. To prove the theorem we explicitly construct a hypothesis space that is not learnable against the malicious adversary. As discussed in Section 5.3.4, a constructive proof is necessary here, because fairness can be trivially satisfied on some hypothesis spaces, for example those that contain a constant classifier, which is fair under any distribution and against any adversary.

We now analyze the bounds and their behavior for small values of α and P_0. First assume that $\frac{\alpha}{1-\alpha} < 2P_0P_1$, which in particular is the case whenever $2\alpha < P_0$. Then under the conditions of the theorem, with probability at least 0.5[1]

$$\mathcal{R}(\mathcal{L}(S^p)) - \mathcal{R}(h^*) \geq \Omega(\alpha) \tag{5.10}$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p)) - \Gamma^{DP}(h^*) \geq \Omega\left(\frac{\alpha}{P_0}\right). \tag{5.11}$$

The lower bound on the excess loss (5.10) is known to hold for any hypothesis space as shown by [KL93] (see also the discussion in Section 2.2.3). What Theorem 8 adds to this classic result is that for certain hypothesis spaces: 1) the learner can at the same time be forced to produce an excessively unfair classifier; 2) the fairness deviation measure Γ^{DP} can be increased by $\Omega(\alpha/P_0)$. Note that *these results hold regardless of the sample size* n.

Equations (5.10) and (5.11) immediately imply the following lower bounds on L_λ and $\mathbf{L}^{DP}$:

$$L_\lambda^{DP}(L(S^p)) - \min_{h\in\mathcal{H}} \mathcal{L}_\lambda^{DP}(h) \geq \Omega\left(\alpha + \lambda\frac{\alpha}{P_0}\right). \tag{5.12}$$

[1]We use the Ω-notation for lower bounds on the growth rates of functions.

$$\mathbf{L}^{DP}(\mathcal{L}(S^p)) \succeq \left(\Omega(\alpha), \Omega\left(\frac{\alpha}{P_0}\right)\right) \tag{5.13}$$

In the second case, when $\frac{\alpha}{1-\alpha} \geq 2P_0P_1$, the adversary can force a constant increase in the loss and make the classifier completely unfair, so that $\Gamma^{DP}(\mathcal{L}(S^p)) = 1$. These observations, combined with the rates from the first case, indicate that unless $\alpha = o(P_0)$, the adversary can ensure that the resulting model's demographic parity deviation measure is constant. In particular, *if one of the protected groups is rare, even very small levels of data corruption can lead to a biased model*.

Next we show a similar result for equality of opportunity.

Theorem 9. *Let $0 \leq \alpha < 0.5$ and $P_{10} \leq P_{11} < 1$ be such that $P_{10} + P_{11} < 1$. For any input set $\mathcal{X}$ with at least five distinct points, there exists a finite hypothesis space $\mathcal{H}$, such that for any learning algorithm $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$, there exists a distribution $\mathbb{P}$ for which $\mathbb{P}(A = a, Y = 1) = P_{1a}$ for $a \in \{0, 1\}$, a malicious adversary $\mathfrak{A}$ of power α and a hypothesis $h^* \in \mathcal{H}$, such that with probability at least 0.5*

$$\mathcal{R}(\mathcal{L}(S^p), \mathbb{P}) - \mathcal{R}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{1-\alpha}, 2P_{10}, 2(1 - P_{10} - P_{11})\right\}$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}) - \Gamma^{EOp}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{2(1-\alpha)P_{10}}, 1, \frac{1 - P_{10} - P_{11}}{P_{10}}\right\}.$$

Discussion A similar analysis to the one after Theorem 8 applies here as well. In particular, whenever $\frac{\alpha}{1-\alpha} \leq 2\min\{P_{10}, 1 - P_{10} - P_{11}\}$, we obtain

$$L_\lambda^{EOp}(\mathcal{L}(S^p)) - \min_{h\in\mathcal{H}} L_\lambda^{EOp}(h) \geq \Omega\left(\alpha + \lambda\frac{\alpha}{P_{10}}\right) \tag{5.14}$$

$$\mathbf{L}^{EOp}(\mathcal{L}(S^p)) \succeq \left(\Omega(\alpha), \Omega\left(\frac{\alpha}{P_{10}}\right)\right). \tag{5.15}$$

The case when $\frac{\alpha}{1-\alpha} > 2\min\{P_{10}, 1 - P_{10} - P_{11}\}$ leads to a constant equality of opportunity deviation measure. If in addition we have that $1 - P_{10} - P_{11} \geq P_{10}$, a completely unfair classifier will be returned. Consequently, if positive examples associated with one of the protected groups are rare (that is, if $P_{10} = \mathbb{P}(Y = 1, A = 0)$ is small), then even very small corruption ratios can lead to a biased model.

5.4.2 Hurting fairness without affecting accuracy

While the results above shed light on the fundamental limits of robust fairness-aware learning against an adversary, models that are inaccurate are often easy to detect in practice. On the other hand, a model that has good accuracy, but exhibits a bias with respect to the protected attribute, can be much more problematic. This is especially true in applications where demographic data is not collected at prediction time for privacy reasons. In this case the model's bias might go unnoticed for a long time, thus adversely affecting one of the population subgroups and potentially extrapolating existing biases from the training data to future decisions.

We now show that such an unfortunate situation is indeed also possible under the malicious adversary model. The following results show that any learner will, in some situations, be forced by the adversary to return a model that is optimal in terms of accuracy, but exhibits unnecessarily high unfairness in terms of demographic parity.

Theorem 10. *Let $0 \leq \alpha < 0.5, 0 < P_0 \leq 0.5$. For any input set $\mathcal{X}$ with at least four distinct points, there exists a finite hypothesis space $\mathcal{H}$, such that for any learning algorithm $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$, there exists a distribution $\mathbb{P}$ for which $\mathbb{P}(A = 0) = P_0$, a malicious adversary $\mathfrak{A}$ of power α and a hypothesis $h^* \in \mathcal{H}$, such that with probability at least 0.5*

$$\mathcal{R}(\mathcal{L}(S^p), \mathbb{P}) = \mathcal{R}(h^*, \mathbb{P}) = \min_{h\in\mathcal{H}} \mathcal{R}(h, \mathbb{P})$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}) - \Gamma^{DP}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{2P_0}, 1\right\}.$$

We also present a corresponding result for equality of opportunity.

Theorem 11. *Let $0 \leq \alpha < 0.5, P_{10} \leq P_{11} < 1$ be such that $P_{10} + P_{11} < 1$. For any input set $\mathcal{X}$ with at least five distinct points, there exists a finite hypothesis space $\mathcal{H}$, such that for any learning algorithm $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$, there exists a distribution $\mathbb{P}$ for which $\mathbb{P}(A = a, Y = 1) = P_{1a}$ for $a \in \{0, 1\}$, a malicious adversary $\mathfrak{A}$ of power α and a hypothesis $h^* \in \mathcal{H}$, such that with probability at least 0.5*

$$\mathcal{R}(\mathcal{L}(S^p), \mathbb{P}) = \mathcal{R}(h^*, \mathbb{P}) = \min_{h\in\mathcal{H}} \mathcal{R}(h, \mathbb{P})$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}) - \Gamma^{EOp}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{2(1-\alpha)P_{10}}\left(1 - \frac{P_{10}}{P_{11}}\right), 1 - \frac{P_{10}}{P_{11}}\right\}.$$

Once again the error terms on the fairness notions are inversely proportional to P_0 and P_{10} respectively, indicating that datasets in which one of the subgroups is underrepresented are particularly vulnerable to data manipulations.

5.5 Upper bounds

We now prove that the (sample-size-independent) lower bounds from the previous section are tight up to constant factors, by providing matching upper bounds for the same problem. We do so by studying the performance of two natural types of fairness-aware learning algorithms under the malicious adversary model. We find that these algorithms achieve order-optimal performance in the large data regime.

Complete proofs of all results in this section can be found in Appendix C.2. A sketch of the proofs is also presented in Section 5.5.3.

5.5.1 Upper bounds on the λ-weighted objectives

The first type of algorithms we study simply minimize an empirical estimate of the λ-weighted objective L_λ. We show that with high probability such learners achieve an order-optimal deviation from $\min_{h\in\mathcal{H}} L_\lambda(h)$ in the large data regime, as long as $\mathcal{H}$ has a finite VC dimension.

Bound for demographic parity Let $h \in \mathcal{H}$ be a fixed hypothesis. We consider the following natural estimate of $\Gamma^{DP}(h)$, as given in equation (5.4), based on the corrupted dataset $S^p = \{(x_i^p, a_i^p, y_i^p)\}_{i=1}^n$:

$$\widehat{\Gamma}^{DP}(h) = \left| \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = 0\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = 0\}} - \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = 1\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = 1\}} \right|, \tag{5.16}$$

with the convention that $\frac{0}{0} = 0$ for the purposes of this definition. We also denote the empirical risk of h on S^p by $\widehat{R}^p(h) = \frac{1}{n}\sum_{i=1}^n \mathbb{1}\{h(x_i^p) \neq y_i^p\}$.

Now let $\lambda \geq 0$ be a fixed trade-off parameter. Suppose that the learner $\mathcal{L}_\lambda^{DP} : \cup_{n=1}^\infty (\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$ is such that

$$\mathcal{L}_\lambda^{DP}(S^p) \in \operatorname*{argmin}_{h \in \mathcal{H}} (\widehat{R}^p(h) + \lambda \widehat{\Gamma}^{DP}(h)), \quad \text{for all } S^p.$$

That is, $\mathcal{L}_\lambda^{DP}$ always returns a minimizer of the λ-weighted empirical objective. Then the following result holds.

Theorem 12. *Let $\mathcal{H}$ be any hypothesis space with $d = VC(\mathcal{H}) < \infty$. Let $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be a fixed distribution and let $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Denote by $\mathbb{P}^{\mathfrak{A}}$ the probability distribution of the corrupted data S^p, under the random sampling of the clean data, the marked points and the randomness of the adversary. Then for any $\delta \in (0,1)$ and $n \geq \max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_0}, \frac{12\log(12/\delta)}{\alpha}, \frac{d}{2}\right\}$, we have:*

$$\mathbb{P}^{\mathfrak{A}}\left(L_\lambda^{DP}(\mathcal{L}_\lambda^{DP}(S^p)) \leq \min_{h \in \mathcal{H}} L_\lambda^{DP}(h) + \Delta_\lambda^{DP}\right) > 1 - \delta,$$

where[2]

$$\Delta_\lambda^{DP} = 3\alpha + \lambda(2\Delta^{DP}) + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}} + \lambda\sqrt{\frac{d}{P_0 n}}\right)$$

and

$$\Delta^{DP} = \frac{2\alpha}{\frac{P_0}{3} + \alpha} = \mathcal{O}\left(\frac{\alpha}{P_0}\right).$$

This result shows that for any $\mathcal{H}$ of finite VC dimension, any distribution $\mathbb{P}$ and against any malicious adversary $\mathfrak{A}$ of power α, the learner $\mathcal{L}_\lambda^{DP}$ is able, for sufficiently large values of the sample size $n \geq \Omega((P_0/\alpha)^2)$, to return with high probability a hypothesis such that

$$L_\lambda^{DP}(\mathcal{L}_\lambda^{DP}(S^p)) - \min_{h \in \mathcal{H}} L_\lambda^{DP}(h) \leq \mathcal{O}\left(\alpha + \lambda\frac{\alpha}{P_0}\right). \tag{5.17}$$

Note that these rates on the irreducible error term match our lower bound from Theorem 8 and Inequality (5.12). Indeed, the hardness result shows that no algorithm can guarantee better error rates than those in (5.17) on the family of finite hypothesis sets and hence also on the hypothesis sets with finite VC dimension.

[2]The $\widetilde{\mathcal{O}}$-notation hides constant and logarithmic factors.

Bound for equality of opportunity Similarly, we consider the following estimate for the equality of opportunity deviation measure:

$$\widehat{\Gamma}^{EOp}(h) = \left| \frac{\sum_{i=1}^{n} \mathbb{1}\{h(x_i^p) = 1, a_i^p = 0, y_i^p = 1\}}{\sum_{i=1}^{n} \mathbb{1}\{a_i^p = 0, y_i^p = 1\}} - \frac{\sum_{i=1}^{n} \mathbb{1}\{h(x_i^p) = 1, a_i^p = 1, y_i^p = 1\}}{\sum_{i=1}^{n} \mathbb{1}\{a_i^p = 1, y_i^p = 1\}} \right|, \tag{5.18}$$

with the convention that $\frac{0}{0} = 0$ for the purposes of this definition. Suppose that a learner $\mathcal{L}_\lambda^{EOp} : \cup_{n=1}^{\infty}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$ is such that

$$\mathcal{L}_\lambda^{EOp}(S^p) \in \operatorname*{argmin}_{h \in \mathcal{H}} (\widehat{R}^p(h) + \lambda \widehat{\Gamma}^{EOp}(h)), \quad \text{for all } S^p,$$

that is, always returns a minimizer of the λ-weighted empirical objective. Then:

Theorem 13. *Let $\mathcal{H}$ be any hypothesis space with $d = VC(\mathcal{H}) < \infty$. Let $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be a fixed distribution and let $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Then for any $\delta \in (0,1)$ and $n \geq \max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(12/\delta)}{\alpha}, \frac{d}{2}\right\}$*

$$\mathbb{P}^{\mathfrak{A}}\left(L_\lambda^{EOp}(\mathcal{L}_\lambda^{EOp}(S^p)) \leq \min_{h \in \mathcal{H}} L_\lambda^{EOp}(h) + \Delta_\lambda^{EOp}\right) > 1 - \delta,$$

where

$$\Delta_\lambda^{EOp} = 3\alpha + \lambda(2\Delta^{EOp}) + \tilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}} + \lambda\sqrt{\frac{d}{P_{10}n}}\right)$$

and

$$\Delta^{EOp} = \frac{2\alpha}{\frac{P_{10}}{3} + \alpha} = \mathcal{O}\left(\frac{\alpha}{P_{10}}\right).$$

Again, for a sufficiently large sample size, this result implies an upper bound on the excess loss of the learner $\mathcal{L}_\lambda^{EOp}$ in terms of the weighted objective

$$L_\lambda^{EOp}(\mathcal{L}_\lambda^{EOp}(S^p)) - \min_{h \in \mathcal{H}} L_\lambda^{EOp}(h) \leq \mathcal{O}\left(\alpha + \lambda\frac{\alpha}{P_{10}}\right), \tag{5.19}$$

which is again order optimal, according to Theorem 9 and Inequality (5.14).

5.5.2 Component-wise upper bounds

We now introduce a second type of algorithms, which return a hypothesis that achieves both a small loss and a small fairness deviation measure on the training data, or, if no such hypothesis exists, a random hypothesis. We show that, in the case when there exists a classifier that is optimal in both accuracy and fairness, with high probability such learners return a hypothesis $h \in \mathcal{H}$ that is order-optimal in both elements of the objective vector $\mathbf{L}(h)$, as long as $\mathcal{H}$ is of finite VC dimension and n is sufficiently large. Finally, in the case of realizable PAC learning with equality of opportunity fairness, we are able to provide an algorithm that achieves such order-optimal guarantees with *fast statistical rates*, for any finite hypothesis space.

Throughout the section, we assume that there exists a classifier $h^* \in \mathcal{H}$, such that $\mathfrak{V}(h^*) \preceq \mathfrak{V}(h)$ for all $h \in \mathcal{H}$. That is, $\mathcal{R}(h^*) \leq \mathcal{R}(h)$ and $\Gamma(h^*) \leq \Gamma(h)$ for all $h \in \mathcal{H}$. We also assume that $d = VC(\mathcal{H}) < \infty$.

We note that the algorithms studied in this section require the knowledge of α and of P_0 and P_{10} for demographic parity and equality of opportunity respectively, since they explicitly use these quantities when selecting a hypothesis. Estimates of these two quantities can often be obtained in practice, for example by having the quality of a small random subset of the data S^p verified by a trusted authority, or via conducting an additional survey/crowdsourcing experiment.

Bound for demographic parity Given a corrupted dataset $S^p = \{(x_i^p, a_i^p, y_i^p)\}$, let $\widehat{h}^r \in \operatorname{argmin}_{h\in\mathcal{H}} \widehat{\mathcal{R}}^p(h)$ and $\widehat{h}^{DP} \in \operatorname{argmin}_{h\in\mathcal{H}} \widehat{\Gamma}^{DP}(h)$. Further, we define the sets

$$\mathcal{H}_1 = \left\{ h \in \mathcal{H} : \widehat{\mathcal{R}}^p(h) - \widehat{\mathcal{R}}^p(\widehat{h}^r) \leq 3\alpha + 4\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}} \right\}$$

$$\mathcal{H}_2 = \left\{ h \in \mathcal{H} : \widehat{\Gamma}^{DP}(h) - \widehat{\Gamma}^{DP}(\widehat{h}^{DP}) \leq 2\Delta^{DP} + 32\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_0 n}} \right\}.$$

That is, $\mathcal{H}_1$ and $\mathcal{H}_2$ are the sets of classifiers that are not far from optimal on the train data, in terms of their risk and their fairness respectively. The upper bound terms are selected according to the concentration properties of the two measures and describe the amount of expected variability of those, due to the randomness of the training data. Now define the *component-wise learner*:

$$\mathcal{L}_{cw}^{DP}(S^p) = \begin{cases} \text{any } h \in \mathcal{H}_1 \cap \mathcal{H}_2, & \text{if } \mathcal{H}_1 \cap \mathcal{H}_2 \neq \emptyset \\ \text{any } h \in \mathcal{H}, & \text{otherwise,} \end{cases}$$

that returns a classifier that is good in both metrics, if such exists, or an arbitrary classifier otherwise. Then the following result holds.

Theorem 14. *Let $\mathcal{H}$ be any hypothesis space with $d = VC(\mathcal{H}) < \infty$. Let $\mathbb{P} \in \mathcal{P}(\mathcal{X}\times\mathcal{A}\times\mathcal{Y})$ be a fixed distribution and let $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Suppose that there exists a hypothesis $h^* \in \mathcal{H}$, such that $\mathfrak{V}(h^*) \preceq \mathfrak{V}(h)$ for all $h \in \mathcal{H}$. Then for any $\delta \in (0,1)$ and $n \geq \max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_0}, \frac{12\log(12/\delta)}{\alpha}, \frac{d}{2}\right\}$, with probability at least $1-\delta$:*

$$\mathbf{L}^{DP}(\mathcal{L}_{cw}^{DP}(S^p)) \preceq \left(6\alpha + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}}\right), 4\Delta^{DP} + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{P_0 n}}\right)\right).$$

Since $\Delta^{DP} = \mathcal{O}\left(\frac{\alpha}{P_0}\right)$, in the large data limit we obtain that

$$\mathbf{L}^{DP}(\mathcal{L}_{cw}^{DP}) \preceq \left(\mathcal{O}(\alpha), \mathcal{O}\left(\frac{\alpha}{P_0}\right)\right). \tag{5.20}$$

Note that this bound is order-optimal for the class of finite hypothesis spaces, and hence also for the class of hypothesis spaces with finite VC dimension, according to Theorem 8 and Inequality (5.13).

Bound for equality of opportunity Similarly, let $\widehat{h}^{EOp} \in \operatorname{argmin}_{h\in\mathcal{H}} \widehat{\Gamma}^{EOp}(h)$. Further, we define the set

$$\mathcal{H}_3 = \left\{ h \in \mathcal{H} : \widehat{\Gamma}^{EOp}(h) - \widehat{\Gamma}^{EOp}(\widehat{h}^{EOp}) \leq 2\Delta^{EOp} + 32\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_{10} n}} \right\}.$$

That is, $\mathcal{H}_3$ is the set of classifiers that are not far from optimal on the train data, in terms of equality of opportunity fairness. Now define the *component-wise learner* for equality of opportunity:

$$\mathcal{L}_{cw}^{EOp}(S^p) = \begin{cases} \text{any } h \in \mathcal{H}_1 \cap \mathcal{H}_3, & \text{if } \mathcal{H}_1 \cap \mathcal{H}_3 \neq \emptyset \\ \text{any } h \in \mathcal{H}, & \text{otherwise,} \end{cases}$$

that returns a classifier that is good in both metrics, if such exists, or an arbitrary classifier otherwise. Then the following result holds.

Theorem 15. *Let $\mathcal{H}$ be any hypothesis space with $d = VC(\mathcal{H}) < \infty$. Let $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be a fixed distribution and let $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Suppose that there exists a hypothesis $h^* \in \mathcal{H}$, such that $\mathfrak{V}(h^*) \preceq \mathfrak{V}(h)$ for all $h \in \mathcal{H}$. Then for any $\delta \in (0,1)$ and $n \geq \max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(12/\delta)}{\alpha}, \frac{d}{2}\right\}$, with probability at least $1-\delta$*

$$\mathbf{L}^{EOp}(\mathcal{L}_{cw}^{EOp}(S^p)) \preceq \left(6\alpha + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}}\right), 4\Delta^{EOp} + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{P_{10}n}}\right)\right).$$

Since $\Delta^{EOp} = \mathcal{O}\left(\frac{\alpha}{P_{10}}\right)$, in the large data limit we obtain that

$$\mathbf{L}^{EOp}(\mathcal{L}_{cw}^{EOp}) \preceq \left(\mathcal{O}(\alpha), \mathcal{O}\left(\frac{\alpha}{P_{10}}\right)\right). \tag{5.21}$$

Note that this bound is order-optimal for the class of finite hypothesis spaces, and hence also for the class of hypothesis spaces with finite VC dimension, according to Theorem 9 and Inequality (5.15).

Upper bound with fast rates Finally, we study learning with the equality of opportunity fairness notion, in the realizable PAC learning framework, where a perfectly accurate classifier exists. Given this additional assumption, we are able to certify *convergence to an order-optimal error in both fairness and accuracy at fast statistical rates*. For simplicity we assume that $\mathcal{H}$ is finite here.

Specifically, note that while the results presented already achieve order-optimal guarantees in the limit as $n \to \infty$, for a finite amount of samples they incur an additional loss of $\widetilde{\mathcal{O}}\left(\frac{1}{\sqrt{n}}\right)$. Regarding P_0 (for demographic parity) or P_{10} (for equality of opportunity) as fixed, all previous algorithms need $\widetilde{\mathcal{O}}\left(\frac{1}{\alpha^2}\right)$ samples to achieve an excess risk and fairness deviation measure of $\widetilde{\mathcal{O}}(\alpha)$. In contrast, the algorithm we present now only requires $\mathcal{O}\left(\frac{1}{\alpha}\right)$ samples.

Formally, assume that the underlying clean distribution $\mathbb{P}$ is such that there exists a $h^* \in \mathcal{H}$, for which $\mathbb{P}(h^*(X) = Y) = 1$. This implies that $L(h^*) = 0$ and $\Gamma^{EOp}(h^*) = 0$.

Key to the design of an algorithm that achieves fast statistical rates for the objective $\mathbf{L}$ are the following empirical estimates:

$$\bar{\gamma}_{1a}^p(h) = \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 0, a_i^p = a, y_1^p = 1\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a, y_i^p = 1\}} \tag{5.22}$$

of $\bar{\gamma}_{1a}(h) \coloneqq \mathbb{P}(h(X) = 0 | A = a, Y = 1) = 0$ for $a \in \{0,1\}$. Given a (corrupted) training set S^p, denote by

$$\mathcal{H}^*(S^p) := \left\{h \in \mathcal{H} \middle| \max_a \bar{\gamma}_{1a}^p(h) \leq \Delta^{EOp} \wedge \widehat{\mathcal{R}}^p(h) \leq \frac{3\alpha}{2}\right\} \tag{5.23}$$

the set of all classifiers that have a small loss and small values of $\bar{\gamma}^p_{1a}$ for both $a \in \{0,1\}$ on S^p. Consider the learner $\mathcal{L}^{fast}$ defined by

$$\mathcal{L}^{fast}(S^p) = \begin{cases} \text{any } h \in \mathcal{H}^*, & \text{if } \mathcal{H}^* \neq \emptyset \\ \text{any } h \in \mathcal{H}, & \text{otherwise.} \end{cases} \tag{5.24}$$

Then the following result holds.

Theorem 16. *Let $\mathcal{H}$ be finite and $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be such that for some $h^* \in \mathcal{H}$, $\mathbb{P}(h^*(X) = Y) = 1$. Let $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Then for any $\delta, \eta \in (0,1)$ and any*

$$\begin{aligned} n &\geq \max\left\{ \frac{8\log(16|\mathcal{H}|/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(12/\delta)}{\alpha}, \frac{2\log(8|\mathcal{H}|/\delta)}{3\eta^2\alpha}, \frac{2\log(\frac{16|\mathcal{H}|}{\delta})}{3\eta^2(1-\alpha)P_{10}\alpha} \right\} \\ &= \Omega\left(\frac{\log(|\mathcal{H}|/\delta)}{\eta^2 P_{10}\alpha}\right) \end{aligned}$$

with probability at least $1-\delta$

$$\mathbf{L}^{EOp}(\mathcal{L}^{fast}(S^p)) \preceq \left(\frac{3\alpha}{1-\eta}, \frac{2\Delta^{EOp}}{1-\eta}\right).$$

As an immediate consequence of Theorem 16, setting $\eta = \frac{1}{2}$, say, yields that for large n, with high probability

$$\mathbf{L}^{EOp}(\mathcal{L}^{fast}(S^p)) \preceq \left(\mathcal{O}(\alpha), \mathcal{O}\left(\frac{\alpha}{P_{10}}\right)\right). \tag{5.25}$$

Again, this bound is order-optimal for finite hypothesis sets, according to Theorem 9 and Inequality (5.15). In addition, regarding P_{10} as a constant, the number of samples needed for achieving this order-optimal element-wise error is indeed $\mathcal{O}(\frac{1}{\alpha})$, according to Theorem 16, which is faster than the $\widetilde{\mathcal{O}}(\frac{1}{\alpha^2})$ we obtained with the previous results.

5.5.3 Sketch of the upper bounds proofs

Here we present a sketch of the proofs of the upper bounds. The complete proofs can be found in Appendix C.2.

The proofs of Theorems 12, 13, 14, 15 rely on a series of results that describe the deviations of the corrupted fairness estimates $\widehat{\Gamma}(h)$ from the true underlying population values $\Gamma(h)$, uniformly over the hypothesis space $\mathcal{H}$. Key to this is bounding the effect of the data corruption, as expressed by the maximum achievable gap between the corrupted fairness estimates and the corresponding estimates based on the clean (but unknown) subset of the data. Then the large deviation properties of these clean data estimates are studied instead.

Here we make this specific for the case of demographic parity, with the analysis for equality of opportunity being similar. We denote

$$\gamma^p_a(h) = \frac{\sum_{i=1}^n \mathbb{1}\{h(x^p_i) = 1, a^p_i = a\}}{\sum_{i=1}^n \mathbb{1}\{a^p_i = a\}}$$

and

$$\gamma_a(h) = \mathbb{P}(h(X) = 1 | A = a),$$

so that $\widehat{\Gamma}^{DP}(h) = |\gamma_0^p(h) - \gamma_1^p(h)|$ and $\Gamma^{DP}(h) = |\gamma_0(h) - \gamma_1(h)|$. Note that $\gamma_a^p(h)$ is an estimate of a conditional probability *based on the corrupted data*. We now introduce the corresponding estimate that only uses the *unknown clean subset* of the training set S^p

$$\gamma_a^c(h) = \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a, i \notin \mathfrak{P}\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a, i \notin \mathfrak{P}\}}.$$

Bounding the effect of the adversary First, we bound how far the corrupted estimates $\gamma_a^p(h)$ of $\gamma_a(h)$ are from the clean estimates $\gamma_a^c(h)$, uniformly over the hypothesis space $\mathcal{H}$:

Proposition 2. *If* $n \geq \max\left\{\frac{8\log(4/\delta)}{(1-\alpha)P_0}, \frac{12\log(3/\delta)}{\alpha}\right\}$, *we have*

$$\mathbb{P}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}\left(|\gamma_0^p(h) - \gamma_0^c(h)| + |\gamma_1^p(h) - \gamma_1^c(h)|\right) \geq \frac{2\alpha}{\frac{P_0}{3} + \alpha}\right) < \delta.$$

Informally, this proposition allows us to connect the corrupted estimate $\widehat{\Gamma}^{DP}(h)$ with the corresponding ideal clean estimate $\widehat{\Gamma}^c(h) = |\gamma_0^c(h) - \gamma_1^c(h)|$.

Bounding the deviation of the clean data estimate Secondly, a technique used by [WGOS17] and [ABD+18] for proving concentration of fairness measures is used to derive a concentration result for the clean estimates $\gamma_a^c(h)$, around the true population values $\gamma_a(h)$. This, together with Proposition 2, allows us to bound the gap between the corrupted estimate $\widehat{\Gamma}^{DP}(h)$ and the true population value $\Gamma^{DP}(h)$, for a single hypothesis.

Making the bound uniform over $\mathcal{H}$ Finally, the bound obtained is made uniform over $\mathcal{H}$. For this, we use the classic symmetrization technique [Vap13] for proving bounds uniformly over hypothesis spaces of finite VC dimension. However, since the objective is different from the 0-1 loss, care is needed to ensure that the argument goes through, so the proof is given in full detail in the supplementary material.

Once a uniform bound on the deviations of the corrupted fairness estimates from the true underlying population values is obtained, the results of Theorems 12, 13, 14, 15 follow similarly to most classic ERM results.

Proof of Theorem 16 Similarly to the other results, the proof of Theorem 16 first links the corrupted estimates $\bar{\gamma}_{1a}^p$ to their clean counterparts and then uses the clean data concentration to study the behavior of the corrupted estimates. However, an important tool that allows us to obtain the fast statistical rates, is a set of *multiplicative concentration bounds* on the $\bar{\gamma}_{1a}^p$ estimates. Full details and a complete proof can be found in the supplementary material.

5.6 Summary and discussion

In this chapter we explored the statistical limits of fairness-aware learning algorithms on corrupted data, under the malicious adversary model. Our results show that data manipulations

can have an inevitable negative effect on model fairness and that this effect is even more expressed for problems where a subgroup in the population is underrepresented. We also provided upper bounds that match our hardness results up to constant factors, in the large data regime.

While the strong adversarial model and the statistical PAC learning analysis we have considered are mostly of theoretical interest, we believe that the hardness results have several important implications. Indeed, crucial to increasing the trust in learned decision making systems is the ability to guarantee that they exhibit a high amount of fairness, regardless of any known or unforeseen biases in the training data. In contrast, we have shown that this is provably impossible under a strong adversarial model for the data corruption.

We believe that these results stress on the importance of developing and studying further data corruption models in the context of fairness-aware learning. As discussed in the related work section, previous research has shown that it can be possible to recover a fair model under corruptions of the labels or the protected attributes only. While real-world data is likely to contain more subtle manipulations, one may hope that for certain applications there will be models of data corruption that are, on the one hand, sufficiently broad to cover the data issues and, on the other hand, specific enough so that fair learning becomes possible.

Our results can also be seen as an indication that strict data collection practices may in fact be necessary for designing provably fair machine learning models. Indeed, our bounds hold under the assumption that the learner can only access one dataset of unknown quality. In contrast, it has been shown that the use of even a small trusted dataset (that is, a certified clean subset of the data) can greatly improve the performance of machine learning models under corruption, both in the context of classic PAC learning [HMWG18b, KL19] and in the context of fairness-aware learning [RLWS20]. Such data can also be helpful for the sake of validating the fairness of a model as a precautionary step before its real-world adoption.

In summary, understanding and accounting for the types of biases present in machine learning datasets is crucial for addressing the issues brought up in this chapter and for the development of certifiably fair learning models.

CHAPTER 6

Fairness through Regularization for Learning to Rank

In this chapter we move away from the topic of data corruption and study another issue that hinders the practicability of fairness-aware learning, namely that most existing techniques and algorithms in this field are designed to work for binary classification only. In contrast, fairness issues also arise in situations where machine learning models are used in far more sophisticated domains, in particular in ranking. Here we study fairness in the context of learning to rank. We start by motivating why fairness is important to consider in this context. We then show how numerous fairness notions can be translated from the language of classification to learning to rank and can be integrated in this new context with minor computational cost and with theoretical guarantees.

6.1 Motivation and outline

Ranking problems are abundant in many contemporary subfields of machine learning and artificial intelligence, including web search, question answering, reviewer allocation, recommender systems and bid phrase suggestions [MSR08]. Decisions taken by ranking systems affect our everyday life, leading to concerns about the fairness of ranking algorithms.

Indeed, ranking systems are typically designed to maximize utility and return the results most likely relevant for each query [Rob77]. This can have potentially harmful down-stream effects. For example, in 2015 Google became the target of heavy criticism after news reports that when searching for "CEO" in Google's image search, images of women were massively underrepresented [Lo15]. Similar problems still exist in other ranking applications, such as product recommendations or online dating.

To remedy such problems, a number of fair ranking methods have been proposed. However, these suffer from two main shortcomings. Firstly, most existing works introduce *specific fairness notions*, for example by demanding that two groups of items receive an amount of exposure proportional to the groups' utilities [BGW18, SJ18]. While well-suited for a certain range of applications, these notions may not be applicable in other scenarios, be it due to specific ethical considerations, legislation or other reasons that require a different notion of fairness to be considered. For example, when selecting a set of k applicants for a job interview, one may

be required to invite an equal number of people from disadvantaged and non-disadvantaged backgrounds, rendering the above fairness notion inapplicable.

Secondly, few existing fair ranking methods are suited for the increasingly adopted *learning to rank (LTR)* paradigm [Liu11, MC18], where a machine learning model is trained to predict the relevances of the items for any query at test time. A LTR algorithm has to learn to be fair based only on the training data, bringing up the problem of generalization for fairness: a ranking method could appear fair on the training data, but turn out unfair at prediction time. To be best of our knowledge, no previous fair ranking algorithm provides generalization guarantees.

Outline In this chapter, we address the aforementioned challenges and develop fairness-aware algorithms for LTR that can incorporate numerous fairness notions and provably generalize. To this end, we exploit connections between ranking and classification, where both the algorithmic and theoretic challenges of learning fair models are rather well understood [BHN19, CGJ+19, WGOS17]. Importantly, many different notions of classification fairness have been proposed, which describe different properties that are desirable in various applications [MMS+19].

We provide a formalism for translating such well-established and well-understood fairness notions from classification to ranking by phrasing LTR as a binary classification problem for every query-item pair. We exemplify our approach on three fairness notions that fit naturally in the ranking setting and correspond the popular concepts in classification, discussed in Section 2.3: *demographic parity*, *equalized odds*, and *equality of opportunity*. We then formulate corresponding *fairness regularization terms*, which can be incorporated with minor overhead into many standard LTR algorithms.

Besides its flexibility, another advantage of our approach is that it makes the task of fair LTR readily amendable to a learning-theoretic analysis. Specifically, we show generalization bounds for the three considered fairness notions, using a chromatic concentration bound for sums of dependent random variables [Jan04] to overcome the challenge that training samples for the same query are not independent.

Finally, we demonstrate the practical usefulness of our method for training fair models. Experiments on two ranking datasets confirm that training with our regularizers indeed yields models with greatly improved fairness at test time, often with little to no reduction of ranking quality. In contrast, prior fair ranking methods are unable to consistently improve our fairness notions.

6.2 Related work

Fairness in classification Algorithmic fairness is well explored in the context of binary classification, see Section 2.3 and [BHN19]. In this work we show how to extend the three popular group fairness notions from Section 2.3 to the ranking setting. In principle, our formalism is applicable to other group fairness notions, as well as to individual [DHP+12] and causal [KLRS17] fairness. We defer the exploration of these to future work. On the methodological level, we opt for a highly adaptive and scalable regularization approach, inspired by successful similar methods for fair classification [KAS11, KAAS12b, ZVRG17]. Our regularization technique can be integrated into any standard gradient-based LTR system with only a minor additional computational cost. More generally other fair classification

methods, e.g. [BNBR19, CHS20, KCT20, RFMZ20, TYFT20, ZLM18], may be applicable to our framework.

Fairness in ranking Fairness in ranking has so far received less attention than fairness in classification. For an overview of recent techniques, see [Cas19]. Most existing works introduce *a particular fairness notion*, often inspired by a specific application. One popular concept is *fairness of exposure* [BGW18, GDL20, MSHJ20, SZER+19, SJ18, SJ19, YDJ19, ZC20]. It states that the exposure received by a group of items or an individual item should be proportional to its utility. Other works aim at ensuring sufficient representation of items from different groups in the top-k positions of a ranking [CSV18, CMV20, GAK19, YS17, ZBC+17]. Besides group fairness also fair treatment of individuals has been studied in the context of ranking [BEYS21, YGS19]. In contrast to these works, we do not introduce a specific new fairness criterion, but instead provide a formalism that allows for transferring existing fairness notions from classification to ranking.

Perhaps closest to our work is the one of [SJ17], who introduce a number of fair ranking definitions and draw parallels to equalized odds and demographic parity from fair classification. However, they do not provide a formal framework for studying the correspondence between the two setups, and do not study how to optimize these measures in a learning to rank context. Moreover, their fairness measures concern fair rankings for a fixed query, which also holds for the causal fairness notion of [WZW18]. In contrast, our notion of ranking fairness is amortized across queries, similarly to [BGW18].

Another line of work that adapts fair classification techniques to ranking is the one of *pairwise fairness* [BCD+19, KVR19, NCGW20]. However, their classification task is the proxy commonly employed by pairwise ranking methods, namely predicting which one of two items is more relevant than the other for a given query. In contrast, we define fairness in relation to the end decision of whether to return an item for a query or not, which is a more direct measure of the real-world impact of the ranking system. [KZ19] and [VBC20] also study pairwise fairness notions, but with the aim of learning fair continuous risks scores. Among other papers considering broader notions of fairness in ranking, [AJSD19] design learning algorithms that can work with any fairness oracle. The framework however is limited to linear classifiers and the authors do not propose any new fairness notions.

Two further distinctions between our work and previous methods are as follows. Firstly, few prior works consider fairness in the context of learning, and those who do usually propose new training techniques. Instead, the fairness regularizers we introduce can be combined with any existing LTR procedure that can be formulated as learning a score function by minimizing a cost function. Secondly, no prior work provides generalization guarantees for fair ranking as we do.

Fairness in recommender systems For recommender systems, fairness can be studied with respect to the consumers/users (known as C-fairness) or with respect to the providers/items (known as P-fairness) [Bur17]. [Ste18, TPT19] consider calibration and bias disparity within recommender systems with respect to recommended items. In [BSOG18, FKT+18, ZHC18, CPG+19, PK19] various hybrid approaching for achieving both C-fairness and P-fairness are presented. In contrast to our paper, these works are specific to collaborative filtering or tensor-based algorithms and do not carry over to approaches based on supervised learning.

A concept from recommender systems related to demographic parity fairness is that of *neutrality* [KAAS12a], in which one aims to provide recommendations that are independent of a certain

viewpoint. In particular, [KAAS12a, KAAS14] apply a neutrality enhancing regularizer to a recommender system model. The focus of these works, however, lies on dealing with filter bubble problems and no formal links to classification or fairness are made.

Diversity in ranking Another related topic is the one of diversifying the output set of ranking system, see,e.g., [RBCJ09]. However, diversifying rankings generally has the goal of improving the user experience, not a fair treatment of items. A discussion on the relationship between fairness and ranking diversity can be found in [SJ18].

6.3 Preliminaries

In this section we introduce some background information on the learning to rank (LTR) task. For a thorough introduction see [Liu11, MC18].

Learning to rank Let $\mathcal{Q}$ be a set of possible *queries* to a ranking system, and let $\mathcal{I}$ be a set of *items* (or *documents*) that are meant to be ranked according to their relevance for any query. A dataset in the LTR setting typically has the form $S = \{(q_i, d_j^i, r_j^i)\}_{i\in[N],j\in[m_i]}$, i.e. for each of N queries, $q_1, \dots, q_N \in \mathcal{Q}$, a subset of the items $\mathcal{I}_{q_i} = \{d_1^i, d_2^i, \dots, d_{m_i}^i\} \subset \mathcal{I}$ are annotated with binary labels $r_j^i = r(q_i, d_j^i) \in \{0, 1\}$ that indicate if item d_j^i is relevant to query q_i or not. In practice m_i is often much smaller than $|\mathcal{I}|$, since it is typically impractical to determine the relevance of every item for a query.

The goal of LTR is to use a given training set to learn a *ranking procedure* that, for any future query, can return a set of items as well as their order. That is, the learner has to construct a *subset selection function*,

$$R : \mathcal{Q} \to \mathfrak{P}(\mathcal{I}), \tag{6.1}$$

where $\mathfrak{P}$ denotes the powerset operation, as well as an ordering of the predicted item set. The size of the predicted set will depend on the application and may in general be smaller than the total number of relevant items. For example, for ads selection only a limited number of slots may be available on a website.

In practice, R is typically constructed by learning a *score function*, $s : \mathcal{Q} \times \mathcal{I} \to \mathbb{R}$. For any fixed q, $s(q, \cdot)$ induces a total ordering of $\mathcal{I}$, and the set of predicted items is obtained by thresholding or top-k prediction. The function s is usually learned by minimizing a loss function on the quality of the resulting ranking on the train data. Classic examples of this construction are SVMRank [Joa02] or WSABIE [WBU11]. Most other pointwise, pairwise and listwise ranking methods can also be phrased in this way, with differences mainly in how the loss is defined and how the score function is learned numerically [Liu11].

Evaluation measures Many measures exist for evaluating the quality of a ranking system, arguably the simplest being *precision at* k.

Definition 14. *Let S be a test set in the format introduced above. For any query q_i, let $d_1^i, d_2^i, \dots$ be a ranking of the items in $\mathcal{I}_{q_i}$ with associated ground-truth values $r(q_i, d_j^i)$. For any $k \in \mathbb{N} \setminus \{0\}$, the* precision at k *is defined as* $\mathrm{P@k} = \frac{1}{N}\sum_{i=1}^{N} P@k(q_i)$ *with*

$$\mathrm{P@k}(q_i) = \frac{1}{k}\sum_{j=1}^{k} r(q_i, d_j^i). \tag{6.2}$$

For any k, the $\mathrm{P@k}$ value reflects which items appear in the top-k list, but not their ordering. Moreover, $\mathrm{P@k}$ is automatically small for datasets in which queries have few relevant documents. To mitigate these shortcomings, one can add position-dependent weights and normalize by the score of a *best-possible* ranking.

Definition 15. *In the same setting as for Definition 14, the* normalized discounted cumulative gain at k *is defined as* $\mathrm{NDCG@}k = \frac{1}{N}\sum_{i=1}^{N}\mathrm{NDCG@}k(q_i)$ *for*

$$\mathrm{NDCG@}k(q_i) = \Big(\sum_{j=1}^{k}\frac{r(q_i, d_j^i)}{\log_2(j+1)}\Big) \Big/ \Big(\sum_{j=1}^{\min(k,K_i)}\frac{1}{\log_2(j+1)}\Big), \tag{6.3}$$

where $K_i = |\{d \in \mathcal{I}_{q_i} \ : \ r(q_i, d) = 1\}|$ *is the number of relevant items for query* q_i*. Queries with no relevant items are excluded from the average, as the measure is not well-defined for these.*

The role of k in these definitions is twofold. Firstly, in most applications only a fixed number of items, k, can be retrieved in total and so one is only interested in the performance of the ranker up to the k-th document. Secondly, multiple values of k can be considered for the quality of the ranking system to be better assessed.

6.4 Fairness in learning-to-rank

We now introduce our framework for group fairness in ranking. The main step is to exploit a correspondence between ranking and multi-label learning, a view that has previously been employed for practical tasks, e.g., in *extreme classification* [BDJ+19], but not –to our knowledge– to make LTR benefit from prior work on classification fairness.

Specifically, we study how the fairness of the *subset selection function* (6.1) can be evaluated. The objects for which we want to impose fairness, the items, occur as outputs of the learned function. This makes it hard to leverage fairness notions from classification, which are defined with respect to the inputs.

We advocate an orthogonal viewpoint: for any fixed query q, we treat the items not as elements of the predictor's output, but as the inputs to a query-dependent classifier: $f_q : \mathcal{I} \to \{0, 1\}$, where $f_q(d) = 1$, if item d is should be returned for query q, and $f_q(d) = 0$ otherwise. As the query is a priori unknown, this means one ultimately has to find an *item selection function*

$$f : \mathcal{Q} \times \mathcal{I} \to \{0, 1\}. \tag{6.4}$$

While, of course, both views are equivalent, the latter one allows us to readily integrate notions of classification fairness into the LTR paradigm.

Note that even though the item selection function $f(q, d)$ and the relevance label $r(q, d)$ have the same signature, their roles are different. r specifies if an item is relevant for a query or not. f indicates if the item should be returned as a result and hence characterizes the impact of the decisions made by the ranking system. While f will in general be an approximation of r, as learned by the LTR model, f will also likely depend on other factors. For example, if at most k items can be retrieved, but more than k are relevant, some relevant items will end up not being included. Additionally, the choice of f may be influenced by diversity, or, as we argue, fairness considerations.

6.4.1 Group fairness in learning to rank

Group fairness notions in classification are typically based on an underlying probabilistic framework that allows statements about conditional independence relations [BHN19]. Similarly, we assume that $\mathcal{D} \in \mathcal{P}(\mathcal{Q} \times \mathcal{I} \times \{0,1\})$ is an unknown fixed distribution over query/document/relevance triplets. For the rest of our work, all statements about probabilities of events will be with respect to samples $(q, d, r(q,d)) \sim \mathcal{D}$.

Analogously to the situation of classification, we assume that any item $d \in \mathcal{I}$ has a *protected attribute*, $A(d)$, which denotes the group membership for which fairness should be ensured. For example, $A(d)$ can correspond to gender, when the retrieved items are images of people, or to the country of origin of an Amazon product. For simplicity, we assume binary-valued protected attributes, but extensions are easily possible.

A plausible notion of fairness in the context of ranking is: **For any relevant item the probability of being included in the ranker's output should be independent of its protected attribute**. This intuition is easy to formulate in our formalism, resulting in a direct analog of the *equality of opportunity* principle from fair classification [HPS16].

Definition 16 (Equality of opportunity for LTR). *An item selection function* $f : \mathcal{Q} \times \mathcal{I} \to \{0,1\}$ *fulfills the* equality of opportunity *condition, if*

$$\mathbb{P}(f(q,d) = 1 | A(d) = 0, r(q,d) = 1) = \mathbb{P}(f(q,d) = 1 | A(d) = 1, r(q,d) = 1). \tag{6.5}$$

In practice, a ranker will rarely achieve perfect fairness, so we also introduce a quantitative version of Definition 16 in the form of *a fairness deviation measure* [CV10, WGOS17, WM19], that reports a ranking procedure's *amount of unfairness*:

Definition 17. *The* equality of opportunity (EOp) violation *of* $f : \mathcal{Q} \times \mathcal{I} \to \{0,1\}$ *is*

$$\Gamma^{EOp}(f) = \Big| \mathbb{P}(f(q,d) = 1 | A(d) = 0, r(q,d) = 1) - \mathbb{P}(f(q,d) = 1 | A(d) = 1, r(q,d) = 1) \Big|.$$

Clearly, f is fair in the sense of Definition 16 if and only if it fulfills $\Gamma^{\mathrm{EOp}}(f) = 0$.

Other fairness measures As discussed extensively in the literature, different notions of fairness are appropriate under different circumstances. For example, to check the *equality of opportunity* condition one needs to know which items are relevant for a query, and this can be problematic, e.g., if the available data itself exhibits a bias in this respect. A major advantage of our formalism compared to prior fair ranking methods is that it is not partial to a specific fairness measure. Besides *equality of opportunity*, many other notions of group fairness can be expressed by simply translating the corresponding expressions from classification.

For example, one can avoid the problem of a data bias by demanding: **The probability of any item to be selected should be independent of its protected attribute** (disregarding its relevance to the query). In our formalism, this condition is a direct analog of *demographic parity* [CKP09].

Definition 18 (Demographic Parity for LTR). *An item selection function* $f : \mathcal{Q} \times \mathcal{I} \to \{0,1\}$ *fulfills the* demographic parity *condition, if*

$$\mathbb{P}(f(q,d) = 1 | A(d) = 0) = \mathbb{P}(f(q,d) = 1 | A(d) = 1). \tag{6.6}$$

As a corresponding quantitative measure we define the demographic parity (DP) violation *of* f *as*

$$\Gamma^{DP}(f) = \Big|\mathbb{P}(f(q,d)=1|A(d)=0) - \mathbb{P}(f(q,d)=1|A(d)=1)\Big|.$$

Another meaningful notion of fairness in ranking is: **The probability of any item to be selected should be independent of its protected attribute, individually for all relevant and for all irrelevant items.** This condition yields the ranking analog of *equality odds* [HPS16].

Definition 19 (Equalized Odds for LTR). *An item selection function* $f : \mathcal{Q} \times \mathcal{I} \to \{0,1\}$ *fulfills the* equalized odds *condition, if for all* $r \in \{0,1\}$*:*

$$\mathbb{P}(f(q,d)=1|A(d)=0, r(q,d)=r) = \mathbb{P}(f(q,d)=1|A(d)=1, r(q,d)=r) \tag{6.7}$$

The equalized odds (EOd) violation *of* f *is*

$$\Gamma^{EOd}(f) = \frac{1}{2}\sum_{r\in\{0,1\}} \Big|\mathbb{P}(f(q,d)=1|A(d)=0, r(q,d)=r) - \mathbb{P}(f(q,d)=1|A(d)=1, r(q,d)=r)\Big|.$$

6.4.2 Training fair rankers

In order to enforce the fairness of a LTR system during the training phase, we create empirical variants of the fairness violation measures and add them as a regularizer during the training step [ABD+18, KAS11]. For this construction to make sense, we have to answer two questions: *Can we solve the resulting optimization efficiently?* and *Does the inclusion of a regularizer generalize, i.e. ensure fairness also on future predictions?* In rest of this section, we will answer the first question. The second question we will address in Section 6.4.3.

To allow for gradient-based optimization, we parametrize the binary-valued item selection function in a differentiable way using a real-valued score function $s : \mathcal{Q} \times \mathcal{I} \to [0,1]$, similarly to the discussion in Section 6.3. Our inspiration, however, comes from the classification setting, such as logistic regression, and we assume that s is not arbitrary real-valued, but that it parameterizes the probability that d is selected for q, i.e. $s(q,d) = \mathbb{P}(f(q,d)=1)$.

Empirical fairness measures For a given training set, S, in the format discussed in Section 6.3, we obtain empirical estimates of the previously introduced fairness violation measures. For any $a \in \{0,1\}$, $r \in \{0,1\}$, denote by S_a the subset of data points $(q,d,r(q,d))$ in S with $A(d)=a$, and by $S_{a,r}$ the subset of data points in S with $A(d)=a$ and $r(q,d)=r$.

Definition 20 (Empirical fairness violation measures). *For a function* $s : \mathcal{Q} \times \mathcal{I} \to [0,1]$*, its* empirical equality of opportunity violation on a dataset S *is*

$$\Gamma^{EOp}(s;S) = \Big|\frac{1}{|S_{0,1}|}\sum_{(q,d)\in S_{0,1}} s(q,d) - \frac{1}{|S_{1,1}|}\sum_{(q,d)\in S_{1,1}} s(q,d)\Big|.$$

The empirical demographic parity violation of s on S *is*

$$\Gamma^{DP}(s;S) = \Big|\frac{1}{|S_0|}\sum_{(q,d)\in S_0} s(q,d) - \frac{1}{|S_1|}\sum_{(q,d)\in S_1} s(q,d)\Big|.$$

and the empirical equalized odds violation of s on S *is*

$$\Gamma^{EOd}(s;S) = \frac{1}{2}\sum_{r\in\{0,1\}} \Big|\frac{1}{|S_{0,r}|}\sum_{(q,d)\in S_{0,1}} s(q,d) - \frac{1}{|S_{1,r}|}\sum_{(q,d)\in S_{1,1}} s(q,d)\Big|.$$

These expressions can be derived readily as approximations of the conditional probabilities of the individual fairness measures by fractions of the corresponding examples in S. This is done by assuming that the marginal probability of any data point in S is $\mathbb{P}$, and inserting the assumed relation $s(p,q) = \mathbb{P}(f(p,q) = 1)$. Note that Definition 20 applies also to binary-valued functions, so it can also be used to evaluate the fairness of a learned item selection function f on a dataset.

Learning with fairness regularization Let $L(s,S)$ be any loss function ordinarily used to train an LTR model. Instead of optimizing this fairness-agnostic loss, we propose a fairness-regularized objective:

$$L^{\mathsf{fair}}(s;S) = L(s,S) + \alpha\Gamma(s,S) \tag{6.8}$$

for $\alpha \geq 0$, where $\Gamma(s;S)$ is any of the empirical measures of fairness violation. The larger the value of α, the more the resulting rankers will take the fairness of their decisions into account. However, as our experiments in Section 6.5 show, the relation between fairness and ranking quality is not necessarily adversarial.

Optimization The fairness regularization terms, $\alpha\Gamma(s,S)$ and their gradients can be computed efficiently using standard numerical frameworks. In large-scale settings, where memory and computational concerns may arise, the regularized objective (6.8) can also be optimized by stochastic gradient steps over mini-batches, as long as the unregularized loss function $L(s,S)$ supports this as well. The resulting per-batch gradient updates are not unbiased estimators of the full gradient, though, so the characteristics of the fairness notion change depending on the batch size. For example, if each batch is formed of a single query with all associated documents, fairness would be enforced individually for each query, instead of on average across all queries. In our experiments, however, we did not observe any deleterious effect of stochastic training when using a moderate batch size of 100.

6.4.3 Generalization

We now show that, given enough data, our train-time regularization procedure will also ensure fairness at prediction time. Our results are similar to the ones in [WGOS17] for the classification setting. However, in the case of ranking there is additional dependence between the samples, which complicates the analysis and influences the complexity term.

Data generation process To study the generalization properties of our fairness measures, we first formally define the statistical properties of the training data. We assume the following data generation process which is consistent with the structure of LTR datasets, with the only simplifying assumption that the item sets for all queries are of equal size m. For a given data distribution $\mathcal{D} \in \mathcal{P}(\mathcal{Q} \times \mathcal{I} \times \{0,1\})$, a dataset $S = \{(q_i, d_j^i, r_j^i)\}_{i\in[N],j\in[m]}$, is sampled as follows: 1) queries, $q_1,\ldots,q_N$, are sampled i.i.d. from the marginal distribution $\mathcal{D}(q)$; 2) for each query q_i independently a set of items, $D_{q_i} = \{d_1^i,\ldots,d_m^i\}$, is sampled *in an arbitrary way* with the only restriction that the marginal distribution of each individual d_j^i should be $\mathcal{D}(d|q_i)$; 3) for each pair (q_i, d_j^i) independently, the relevance r_j^i is sampled from $\mathcal{D}(r|q_i, d_j^i)$.

Note that while each resulting data point has marginal distribution $\mathcal{D}$, a lot of flexibility remains about how the actual items per query are chosen. In particular, the item set can have dependencies, e.g. avoiding repetitions or diversity constraints. While incorporating

dependencies complicates the analysis, we believe that it is necessary, so as to make sure that real-world ranking data, which typically is far from i.i.d., is covered.

We now characterize the generalization properties of the fairness regularizers. Let $\mathcal{F} \subset \{f : \mathcal{Q} \times \mathcal{I} \to \{0,1\}\}$ be a set of item selection functions that make independent deterministic decisions per item (e.g., by thresholding a learned score function). Then, the following holds:

Theorem 17. *Let S be a dataset sampled as described above with $2Nm > v$ for $v = VCdim(\mathcal{F})$. Let $\tau = \min_{r,a}\Big(\mathbb{P}(r(q,d) = r \wedge A(d) = a)\Big)$ and $\upsilon = \min_{a}\Big(\mathbb{P}(A(d) = a)\Big)$. Then, for any $\delta > 0$, each of the following inequalities holds with probability at least $1-\delta$ over the sampling of S, uniformly for all $f \in \mathcal{F}$:*

$$\Gamma^{\mathsf{EOp}}(f) \leq \Gamma^{\mathsf{EOp}}(f,S) + 8\sqrt{2\frac{v\log(\frac{2eNm}{v}) + \log(\frac{24}{\delta})}{N\tau^2}},$$
$$\Gamma^{\mathsf{EOd}}(f) \leq \Gamma^{\mathsf{EOd}}(f,S) + 8\sqrt{2\frac{v\log(\frac{2eNm}{v}) + \log(\frac{48}{\delta})}{N\tau^2}},$$
$$\Gamma^{\mathsf{DP}}(f) \leq \Gamma^{\mathsf{DP}}(f,S) + 8\sqrt{2\frac{v\log(\frac{2eNm}{v}) + \log(\frac{24}{\delta})}{N\upsilon^2}}.$$

Proof sketch. The proof consists of two parts. First, for any fixed item selection function a bound is shown on the gap between the conditional probabilities contributing to a fairness measure and their empirical estimates. For this, we build on the same technique of [WGOS17] for showing concentration of fairness quantities as the one we used in Chapter 5. We combine this with the large deviations bounds for sums of dependent random variables in terms of the chromatic number of their dependence graph of [Jan04], see below. Secondly, the bounds are extended to hold uniformly over the full hypothesis space by evoking a variant of the classic symmetrization argument (e.g. [Vap13]), while carefully accounting for the dependence between the samples. A complete proof can be found in the supplementary material.

Dealing with the between-sample dependence To deal with the dependence between the samples, we use the following framework from [Jan04]. Let Y_α be a set of random variables, with α ranging over some index set $\mathcal{A}$. Let $X = \sum_{\alpha\in\mathcal{A}} Y_\alpha$. To derive concentration bounds for X, the following notions are useful:

Definition 21 ([Jan04])**.** *Given $\mathcal{A}$ and $\{Y_\alpha\}_{\alpha\in\mathcal{A}}$:*

- *A subset $\mathcal{A}' \subset \mathcal{A}$ is independent if the random variables $\{Y_\alpha\}_{\alpha\in\mathcal{A}'}$ are (jointly) independent.*
- *A family $\{\mathcal{A}_j\}_j$ is a cover of $\mathcal{A}$ if $\cup_j \mathcal{A}_j = \mathcal{A}$. A cover is proper if each set $\mathcal{A}_j$ is independent.*
- *$\chi(\mathcal{A})$ is the size of the smallest proper cover of $\mathcal{A}$, that is the smallest integer m, such that $\mathcal{A}$ can be written as the union of m independent subsets.*

Then the following result holds, similar to the Hoeffding inequality, but accounting for the amount of dependence between the random variables $\{Y_\alpha\}_{\alpha\in\mathcal{A}}$:

Theorem 18 ([Jan04]). *Let Y_α and X be as above, with $a_\alpha \leq Y_\alpha \leq b_\alpha$ for every $\alpha \in \mathcal{A}$, for some real numbers a_α and b_α. Then, for every $t > 0$:*

$$\mathbb{P}(X \geq \mathbb{E}(X) + t) \leq \exp\left(-2\frac{t^2}{\chi(\mathcal{A})\sum_{\alpha\in\mathcal{A}}(b_\alpha - a_\alpha)^2}\right). \tag{6.9}$$

The same upper bound holds for $\mathbb{P}(X \leq \mathbb{E}(X) - t)$.

If instead one considers the mean of $\{Y_\alpha\}_{\alpha\in\mathcal{A}}$, namely $\bar{X} = \frac{1}{|\mathcal{A}|}\sum_{\alpha\in\mathcal{A}} Y_\alpha$, then the following holds:

$$\mathbb{P}(\bar{X} \geq \mathbb{E}(\bar{X}) + t) \leq \exp\left(-2\frac{t^2|\mathcal{A}|^2}{\chi(\mathcal{A})\sum_{\alpha\in\mathcal{A}}(b_\alpha - a_\alpha)^2}\right). \tag{6.10}$$

Specifically, if the Y_α are Bernoulli random variables:

$$\mathbb{P}(\bar{X} \geq \mathbb{E}(\bar{X}) + t) \leq \exp\left(-2\frac{t^2|\mathcal{A}|}{\chi(\mathcal{A})}\right). \tag{6.11}$$

For the case of the ranking dataset $S = \{(q_i, d_j^i, r_j^i)\}_{i\in[N],j\in[m]}$ and a fixed function $f : \mathcal{Q}\times\mathcal{I} \to \{0,1\}$, we study the set of random variables $Y_{(i,j)} = f(q_i, d_j^i)$, for the indexes $\mathcal{A} = \{(i,j) : i \in [N], j \in [m]\}$. Examining the assumptions we made about the way that the data is sampled, it is easy to see that $\chi(\mathcal{A}) = m$. Therefore, Theorem 18 can be used to understand the concentration properties of the random variables $Y_{(i,j)} = f(q_i, d_j^i)$.

Discussion Theorem 17 bounds the fairness violation on future data by the fairness on the training set plus an explicit complexity term, uniformly over all item selection functions. Consequently, any item selection function with low fairness violation on the training set will have a similarly low fairness violation on new data, provided that enough data was used for training. Indeed, the complexity term decreases like $\sqrt{\log N/N}$ as $N \to \infty$, which is the expected behavior for a VC-based bound.

The same scaling behavior does not hold with respect to the number of items per query, m. This is unfortunate, but unavoidable, given the weak assumptions we make on the data generation process: because we do not restrict how the per-query item sets are created, each of them could simply consist of many copies of a single item. In that case, even arbitrary large m would provide only as much information as $m = 1$. In the current form, m appears even logarithmically in the numerator of the complexity term. We believe this to be an artifact of our proof technique, and expect that a more refined analysis will allow us to remove this dependence in the future.

Note that for real data, we do expect larger m to be have a beneficial effect on generalization. This is the reason that we prefer to present the bound as it is in the theorem, i.e. with the empirical fairness estimated from all available data, rather than any alternative formulation, e.g. subsampling the training set to $m = 1$, which would recover an i.i.d. setting. Finding an assumption on the generating process of real-world LTR data that does allow bounds that decrease with respect to m is an interesting topic for future work.

In addition, we expect that more advanced techniques from learning theory, e.g. analysis based on Rademacher complexities [BM02], can be applied to obtain sharper, data-dependent guarantees. Indeed, there has been work on extending the classic Rademacher complexity generalization bounds to the case of dependent data, e.g. [UAG05], and we deem the application of such techniques in the context of fair LTR an interesting direction for future work.

6.5 Experiments

We conduct experiments to validate the practicality and performance of our method for training fair LTR systems, including in a large-scale setting. Our emphasis lies on studying the interaction between model quality and fairness, the effectiveness of our proposed method for optimizing these notions on real data and on the comparison to previous fair ranking algorithms.

6.5.1 Datasets and experimental setup

We report experiments on two datasets. As a measure of ranking quality we use $\mathrm{NDCG@}k$ for $k \in \{1, 2, 3, 4, 5\}$, with results for $\mathrm{P@k}$ in the supplemental material. To quantify fairness, we evaluate the three different empirical measures of fairness violation.

TREC Fairness data We use the training data of the TREC 2019 Fairness track dataset [BDEK19]. It consists of 652 queries taken from the Semantic Scholar search engine, together with a set of scientific papers for each query and binary labels for the relevance of every query-paper pair. The average number of labeled papers per query is 7.1, out of which 3.4 are relevant on average. Because of the rather small number of queries, we use five-fold cross-validation to evaluate our method and report averages and standard errors across the folds. As an exemplary *protected attribute* we use a proxy of the authors' seniority. We split the set of documents into two groups based on whether the mean of their authors' $i10$-index proxies (as provided in the TREC data) exceed a threshold t or not. For $t \in \{3, 4, 5\}$ we get different amounts of group imbalance, with the minority group consisting of around $46\%, 26\%$ and 9% of all papers, respectively.

Inspired by the learning to rank approach for the TREC track of [Bon19], we pre-compute 9-dimensional embeddings of every query-paper pair by using the following hardcrafted features:

- the BM25 score of the query with the title, abstract, authors, topics and publication venue of the paper (5 values),
- the number of in- and out-citations (2 values),
- the publication year of the paper (1 value)
- the character length of the query (1 value).

Each feature is normalized by substracting the mean of the feature over the dataset and dividing by its standard deviation.

MSMARCO We also perform experiments on the passage ranking dataset v2.1 of MS-MARCO [NRS+16]. It consists of approximately one million natural language questions, which serve as queries, associated sets of potentially relevant passages from Internet sources, and binary relevance labels for all provided query-document pairs. On average, there are 8.8 passages per question, and the average number of relevant ones is 0.65. We use the default train-development split and report average performance and standard deviation over 10 random seeds. To create a *protected attribute*, we split the passages into two groups based on their top-level domains, as a proxy of the answers' geographic origin. Specifically, we split by `".com`

vs other" (denoted by *com*) and by ".com/.org/.gov/.edu/.net vs other" (denoted by *ext*). Their minority groups are of size 32% and 5% of all passages, respectively.

We use pretrained 768-dimensional BERT feature embeddings [DCLT19] for representing the query-passage pairs. Specifically, we follow the embedding procedure described in [NC19, HWBN20], where each query-passage pair is represented as the following token sequence:

$$[CLS] \text{ query text } [SEP] \text{ passage text } [CLS]$$

This sequence is then processed through a pre-trained BERT model from Tensorflow Hub [AAB+15], with maximum sequence length set to 200, and the hidden units of the first $[CLS]$ token are used as a representation of the query-passage pair.

6.5.2 Learning to rank models

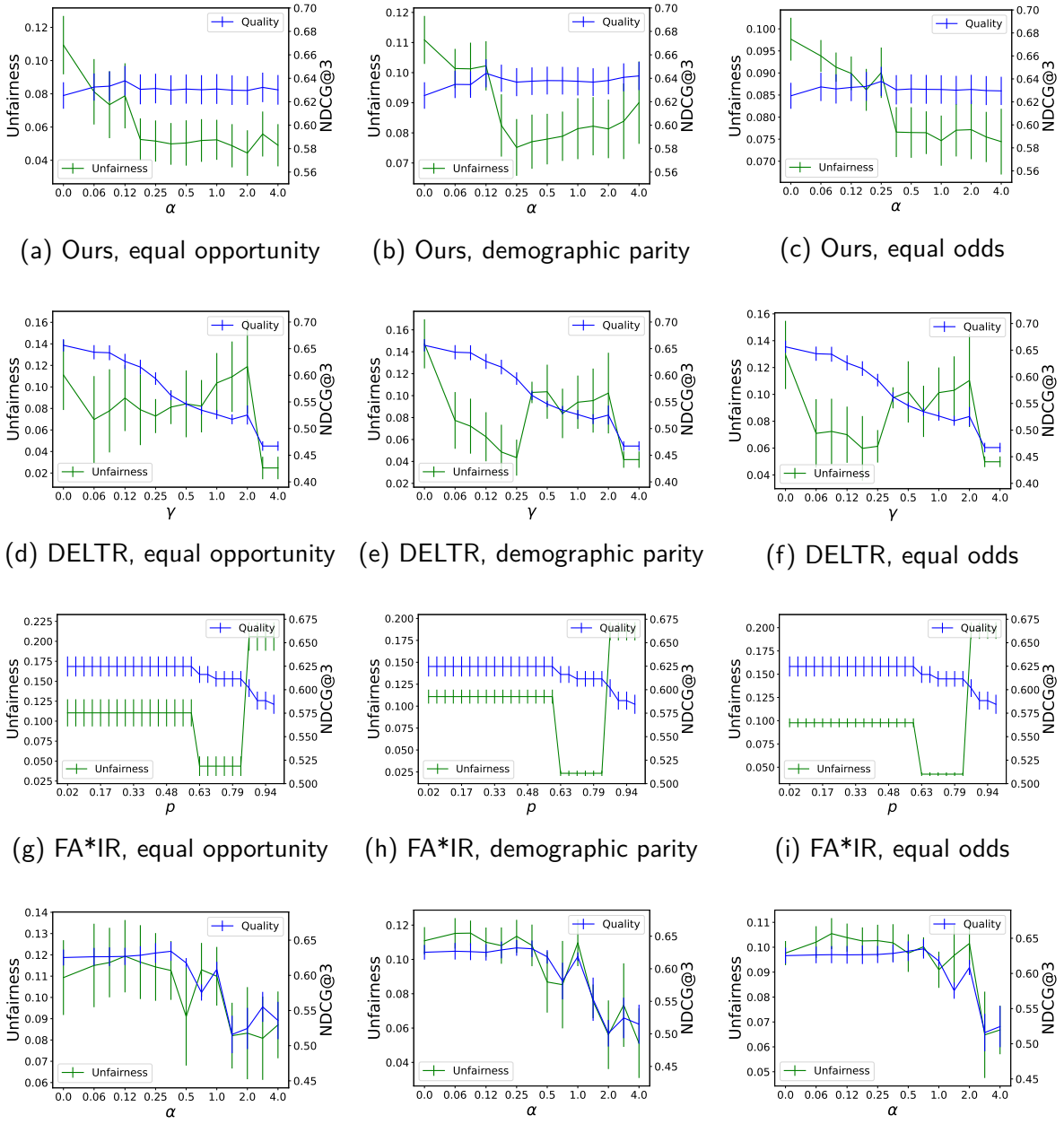
Our algorithm We adopt a classic pointwise LTR approach with a generalized linear score function, $s(d, q) = \langle \theta, \phi(q, d) \rangle$, for a predefined feature function, $\phi : \mathcal{Q} \times \mathcal{I} \to \mathbb{R}^D$. as described in the previous section. As loss function of ranking quality, $L(s, S)$, we use the squared loss between the relevance labels and the predictions of s over all data. To optimize for both ranking quality and fairness, we train with a weighted loss, as in equation (6.8). For TREC we train all models by 1500 steps of gradient descent with a learning rate of 0.003. In the MSMARCO experiments we train with 5 epochs of SGD with a batch size of 100 queries and 10 passages per query and a learning rate of 0.0001.

Baselines and ablation studies Our work provides a novel way of converting well-established fairness notions from classification to a LTR setting. While previously developed methods for fair ranking aim to optimize for other fairness notions, it is informative to see how such algorithms perform against our method, in order to understand the relationship between our and previous works. Therefore, we consider two recent methods for fair ranking, DELTR [ZC20] and FA*IR [ZBC+17], using their public implementation [ZSCK20].

DELTR is a state-of-the-art algorithm for fair LTR. At train time, a linear version of ListNet is trained, together with a regularizer tailored to a notion of disparate exposure [CKP09, SJ18]. We use the same feature representations as for our method, as well as the same range for their regularization parameter γ, to ensure a fair comparison. Unfortunately, the implementation of [ZSCK20] does not scale to MSMARCO.

FA*IR, on the other hand, is an algorithm that *changes the ranking query by query, at prediction time*, by ensuring that whenever k items are retrieved, the proportion of retrieved items from a protected group is not smaller that the β-th quantile of a binomial distribution $Bin(k, p)$, for fixed parameters $p, \beta \in [0, 1]$. We use $\beta = 0.1$ and $p \in [0.02, 0.04, \ldots, 0.98]$. Since FA*IR requires access to the items relevances at prediction time, we first train via our method without a fairness regularizer and then, at test time, use the relevances predicted by our method as inputs to FA*IR.

We also perform an ablation study by considering a version of our algorithm that learns to enforce fairness on the *per-query level*. This is inspired by [SJ17], who, however, do not propose an algorithm for enforcing such per-query fairness notions. Within our framework this is achieved by regularizing with a separate term for every query in a batch and then averaging over the batch afterwards.



(a) Ours, equal opportunity (b) Ours, demographic parity (c) Ours, equal odds

(d) DELTR, equal opportunity (e) DELTR, demographic parity (f) DELTR, equal odds

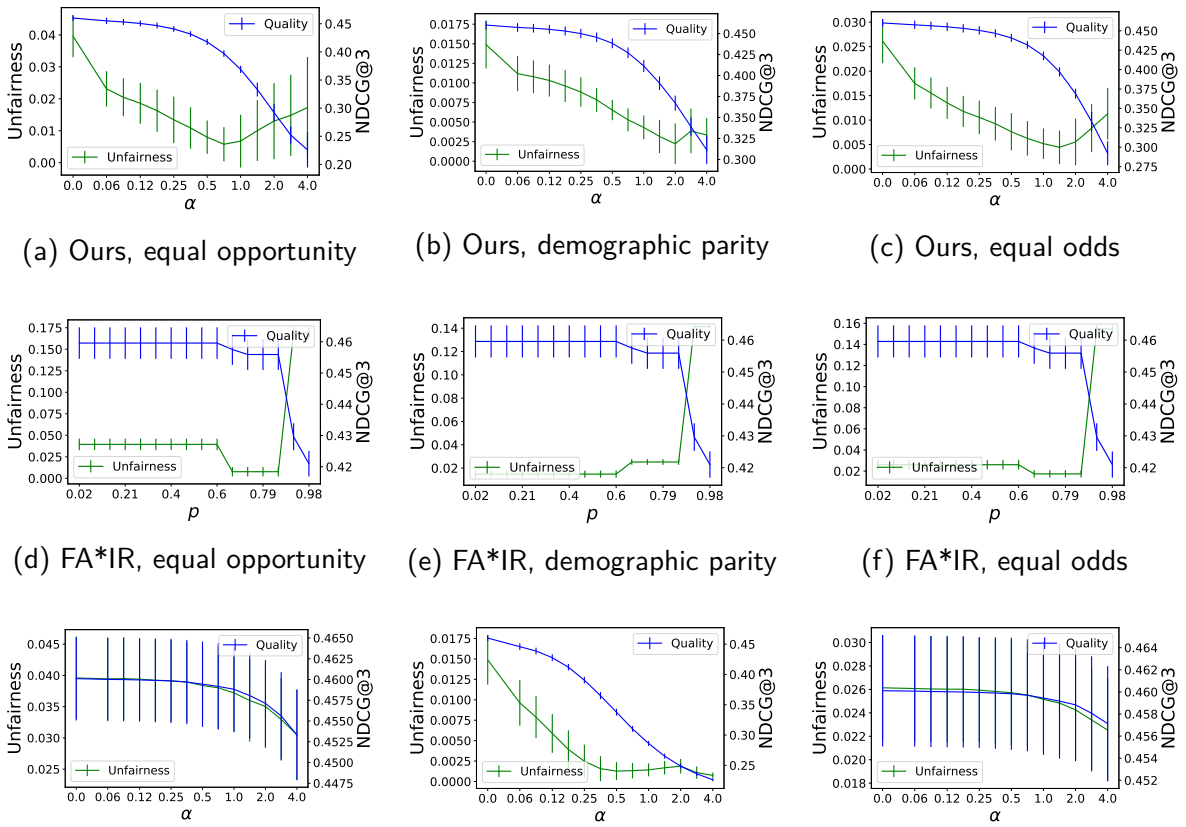
(g) FA*IR, equal opportunity (h) FA*IR, demographic parity (i) FA*IR, equal odds

(j) Per-query, equal opportunity (k) Per-query, demographic parity (l) Per-query, equal odds

Figure 6.1: TREC: Test-time performance of fair rankers with equal opportunity, demographic parity and equalized odds fairness, achieved by our algorithm and the baselines: unfairness (left y-axes) and NDCG@3 ranking quality (right y-axes); after training with different regularization strengths (x-axis).

6.5.3 Results

Figures 6.1 and 6.2 show the ranking quality and the unfairness, achieved by our method and the three baselines, when imposing different amounts of each fairness measure in typical settings for TREC ($t = 3, k = 3$) and MSMARCO (*com*, $k = 3$) respectively. As one can see, our method is able to consistently improve fairness. For TREC, this comes at no loss in ranking quality (here NDCG). For MSMARCO the loss is quite small for small to medium values of α. As the figure shows, these observations are robust across the different amounts



(a) Ours, equal opportunity (b) Ours, demographic parity (c) Ours, equal odds

(d) FA*IR, equal opportunity (e) FA*IR, demographic parity (f) FA*IR, equal odds

(g) Per-query, equal opportunity (h) Per-query, demographic parity (i) Per-query, equal odds

Figure 6.2: MSMARCO: Test-time performance of fair rankers with equal opportunity, demographic parity and equalized odds fairness, achieved by our algorithm and the baselines: unfairness (left y-axes) and NDCG@3 ranking quality (right y-axes); after training with different regularization strengths (x-axis).

of regularization and across the studied fairness measures. In contrast, the fairness curves of the baselines behave erratically with respect to the trade-off parameters.

The possibility of increasing the fairness of learning models without damaging their accuracy has been previously observed in the context of supervised learning [WPT19], but not, to our knowledge, in a ranking context. This effect is more expressed in the experiment on the TREC data than for MSMARCO, possibly due to the higher number of relevant items per query in TREC, which results in more flexibility to rearrange items without decreasing the ranking quality.

We obtained similar results for the other setups, e.g. different values of k, fairness measures and protected attributes and for $\mathrm{P@k}$. *Table 6.1 summarizes some of the results in a compact form.* For different fairness notions and splits into protected groups (rows), it reports the maximal and mean reduction of the fairness violation measure over the range of values of the trade-off parameter for which the corresponding model's prediction quality is not significantly worse than for a model trained without a fairness regularizer (i.e. $\alpha = 0$). Here we call a model significantly worse than another if the difference of the mean quality values of the two models is larger than the sum of the standard errors/deviations, for TREC/MSMARCO respectively, around those averages (i.e. if the error bars, as in Figures 6.1 and 6.2, would not intersect). The results are averaged over $k \in \{1, 2, 3, 4, 5\}$.

Intuitively, the max values quantify how much an algorithm can improve fairness without decreasing the ranking quality, while the mean values report the average improvement of fairness over the values of the trade-off parameters, as a more robust measure. The results confirm that in all cases our proposed training method is able to greatly reduce the unfairness in the test time ranking without majorly damaging ranking quality. In comparison, the baselines behave inconsistently between experiments and are less robust to the choice of the trade-off parameter, indicating that training with the right regularization, as integrated in our method, is indeed beneficial for test time fairness.

Table 6.1: Maximal and mean relative fairness increase, achievable without a significant decrease of ranking quality, for our algorithm and the baselines. See main text for details.

TREC		Ours		DELTR		FA*IR		Per-query	
		Max	Mean	Max	Mean	Max	Mean	Max	Mean
equality of opportunity	$t=3$	48%	34%	39%	23%	56%	18%	19%	-5%
	$t=4$	46%	37%	2%	-8%	56%	11%	14%	-4%
	$t=5$	46%	32%	18%	7%	6%	-17%	8%	-13%
demographic parity	$t=3$	27%	17%	55%	36%	83%	24%	15%	-1%
	$t=4$	44%	32%	12%	6%	56%	10%	27%	5%
	$t=5$	57%	40%	26%	14%	11%	-35%	24%	1%
equalized odds	$t=3$	20%	13%	48%	31%	57%	16%	14%	-3%
	$t=4$	30%	21%	9%	4%	40%	3%	13%	-1%
	$t=5$	29%	21%	21%	12%	0%	-35%	7%	-4%
average		39%	27%	26%	14%	41%	-1%	16%	-3%

MSMARCO		Ours		DELTR		FA*IR		Per-query	
		Max	Mean	Max	Mean	Max	Mean	Max	Mean
equality of opportunity	*com*	55%	36%	NA	NA	64%	11%	24%	6%
	ext	19%	10%	NA	NA	0%	-112%	2%	0%
demographic parity	*com*	42%	27%	NA	NA	39%	-50%	0%	0%
	ext	20%	13%	NA	NA	0%	-168%	7%	3%
equalized odds	*com*	61%	41%	NA	NA	45%	-12%	14%	4%
	ext	28%	17%	NA	NA	0%	-142%	1%	0%
average		37%	24%	NA	NA	25%	-79%	8%	2%

6.6 Summary

We introduced a rigorous framework for transferring classification fairness notions to the context of LTR, by rephrasing ranking as a collection of query-dependent classification problems. This simple viewpoint allows for expanding the optimization methods and proof techniques from fair classification to ranking and multi-label learning. We report the first, to our knowledge, generalization bounds for group fairness in ranking and show that including a suitable regularizer during training can greatly improve the fairness of rankings with no or minor reduction in model quality. This effect seems even more pronounced than in classification tasks, especially if the number of relevant items per query is large. We hypothesize that the multi-label nature of the ranking task naturally allows for more fairness without adverse effects on accuracy.

CHAPTER 7

Discussion and future work

In this thesis we explored several topics in the field of trustworthy machine learning. In particular, we studied adversarial multi-source learning as a model for learning from unreliable data sources. We also explored the topic of fairness, in particular in the presence of data corruption and in the challenging setup of learning to rank. A major focus of this work was on designing machine learning algorithms that come with theoretical guarantees, even under severe problems with the data, i.e. worst-case data corruption, and even in contexts beyond binary classification, such as ranking. We believe that establishing such performance guarantees is a necessary step towards increasing the trust in real-world machine learning algorithms among the general public.

A distinctive feature of the results presented in this thesis, as compared to other works in the field of trustworthy machine learning, is the use of PAC learning techniques for obtaining guarantees for machine learning models. In the context of robustness to training data corruption, a substantial amount of recent research effort [BNL12b, CLL+17, SBC20] has been focused on designing practical gradient-based attacks and defenses for learned systems, resulting in a cat-and-mouse game between the learner and the adversary. In contrast, we studied the fundamental limits of PAC learning in the presence of worst-case data corruption, similarly to [KL93]. In the case of fairness-aware learning, the vast majority of works addressing the problem from a theoretical perspective have focused on understanding fair decision-making when the set of *all possible* binary classifiers is used and when full information about the data distribution is available, disregarding learnability considerations, e.g. [HPS16, MW18a]. In contrast, we accounted for the finite-sample effects when studying fairness-aware learning, similarly to [WGOS17, ABD+18].

We hope that our analysis of the learning-theoretic aspects of robust and fair machine learning serves as a useful complement to the on-going effort in making learned models more trustworthy and reliable.

While this thesis has focused on robustness to training data corruption and on fairness, there are many other desirable properties of machine learning algorithms that are important to satisfy in order to ensure their positive impact in real-world applications. Already in the context of robustness, an orthogonal aspect to the one studied in this thesis is that of protecting learned models from test-time data manipulations. This topic has received a lot of recent attention, for example in the context of image recognition, where it is known that deep neural network models can easily be fooled by inconceivable, but carefully chosen pixel perturbations [SZS+14].

At the same time, there are many real-world applications where the presence of strategic entities altering test time instances to their own advantage is a significant practical concern [HMPW16, AEIK18]. Therefore, dealing with potentially worst-case data perturbations is crucial also in the context of test time manipulations.

Beyond the scope of this thesis are also the issues of privacy and interpretability. Indeed, as machine learning models are increasingly trained on private data, for example via federated learning [MMR+17], it is important to ensure that knowledge about a model or its training process cannot serve as a basis for reconstructing sensitive information [MTV+20]. In addition, with the increasing amounts of available data and computing power, larger and larger models, e.g. deep neural networks, are being adopted for solving various real-world prediction tasks. While excelling in performance, these models often act as black boxes, as they are too large to be analyzed by standard mathematical techniques. Methods for interpreting and explaining the decisions made by such massive models are highly desirable [GBY+18], in particular for the sake of ensuring fairness, detecting spurious correlations or providing feedback to entities that receive an unfavorable classification decision.

Within the context of robust and fair machine learning, there are a number of possible extensions of the work presented in this thesis that we see as interesting directions for *future work*. Below we present these, in an order that aligns with the chapters of the thesis.

Adversarial multi-source learning from heterogeneous data In Chapters 3 and 4 we have studied the problem of learning from multiple unrealiable data sources, where a subset of the sources can contribute arbitrary, even adversarially perturbed, data. However, a recurring assumption is that the clean sources contain data that is sampled from the same distribution, or at least from distributions that are very similar to each other. In contrast, there are many situations where this assumption may not hold. In fact, in setups such as federated learning [MR17] and collaborative learning [BHPQ17], the point of using multiple data sources is exactly to obtain data from different modes of an underlying meta-distribution and some data heterogeneity is not only expected, but also desirable.

In such contexts, an additional challenge towards achieving robustness is that the natural variability in the distributions of the clean sources makes it difficult to distinguish between-source differences that are benign (due to the sources containing different types of clean data) from those that are due to data corruptions and other issues with the datasets. For this distinction to be possible, one would need to have an appropriate model of what "natural" changes of the distribution can be expected between the clean sources.

To our awareness, robustness in the heterogeneous multi-source setting is a problem that has largely remained unaddressed in the literature and we deem this an exciting direction for future work.

Learning theory for fairness-aware learning In Chapter 5 we showed that PAC learning guarantees for fairness can be obtained as long as the VC dimension of the hypothesis class that is used is finite. However, we also argued that in many situations fairness can be certified trivially (and provably), for example in cases where the hypothesis class contains a constant classifier. This means that while the VC dimension being finite is a sufficient condition for "fairness learnability", it is certainly not a necessary one.

Therefore, a natural question is what are the properties of a hypothesis space that precisely characterize learnability in terms of various popular fairness measures. Understanding these

properties can be useful not only for studying the limits of learning in terms of fairness, but also for quantifying the fairness-accuracy trade-off in the context of binary classification.

Fairness-accuracy trade-off in ranking and multi-label learning The fairness-accuracy trade-off is important to understand not only for classification, but also in the context of ranking. As discussed in Chapter 6, our experimental findings suggest that achieving fairness does not necessarily need to come at the price of utility in LTR settings. We hypothesize that this is due to the multi-label nature of ranking problems: for a given query, it is often the case that many items are relevant and so the exact set of returned items, as well as their order, can often be adapted in multiple ways without a substantial drop in utility.

An interesting open question is whether this intuition can be formalized and whether our experimental findings will also hold for real-world recommender systems applications.

Bibliography

[AAB+15] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[AAZL18] Dan Alistarh, Zeyuan Allen-Zhu, and Jerry Li. Byzantine stochastic gradient descent. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[ABD+18] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International Conference on Machine Learing (ICML)*, 2018.

[ABHM17] Pranjal Awasthi, Avrim Blum, Nika Haghtalab, and Yishay Mansour. Efficient PAC learning from the crowd. *Conference on Computational Learning Theory (COLT)*, 2017.

[ADSK18] Dan Alistarh, Christopher De Sa, and Nikola Konstantinov. The convergence of stochastic gradient descent in asynchronous shared memory. In *ACM Symposium on Principles of Distributed Computing (PODC)*, 2018.

[AEIK18] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International Conference on Machine Learing (ICML)*, 2018.

[AHJ+18] Dan Alistarh, Torsten Hoefler, Mikael Johansson, Sarit Khirirat, Nikola Konstantinov, and Cédric Renggli. The convergence of sparsified gradient methods. *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[AJSD19] Abolfazl Asudeh, HV Jagadish, Julia Stoyanovich, and Gautam Das. Designing fair ranking schemes. In *International Conference on Management of Data (COMAD)*, 2019.

[AKM20] Pranjal Awasthi, Matthäus Kleindessner, and Jamie Morgenstern. Equalized odds postprocessing under imperfect group information. In *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2020.

[AL88] Dana Angluin and Philip Laird. Learning from noisy examples. *Machine Learning*, 2(4):343–370, 1988.

[ALM17] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. In *International Conference on Learning Representations (ICLR)*, 2017.

[BBL03] Olivier Bousquet, Stéphane Boucheron, and Gábor Lugosi. Introduction to statistical learning theory. In *Summer School on Machine Learning*, 2003.

[BBL04] Olivier Bousquet, Stéphane Boucheron, and Gábor Lugosi. Introduction to statistical learning theory. In *Advanced lectures on machine learning*, pages 169–207. Springer, 2004.

[BCD+19] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H Chi, et al. Fairness in recommendation ranking through pairwise comparisons. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2019.

[BCMC19] Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. Analyzing federated learning through an adversarial lens. In *International Conference on Machine Learing (ICML)*, 2019.

[BDBC+10] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010.

[BDEK19] Asia J. Biega, Fernando Diaz, Michael D. Ekstrand, and Sebastian Kohlmeier. Overview of the TREC 2019 fair ranking track. In *The Twenty-Eighth Text REtrieval Conference (TREC 2019) Proceedings*, 2019.

[BDJ+19] Samy Bengio, Krzysztof Dembczynski, Thorsten Joachims, Marius Kloft, and Manik Varma. Extreme classification. In *Dagstuhl Reports 18291*. Schloss Dagstuhl – Leibniz Center for Informatics, 2019.

[BEK02] Nader H Bshouty, Nadav Eiron, and Eyal Kushilevitz. Pac learning with nasty noise. *Theoretical Computer Science (TCC)*, 2002.

[BEY05] Allan Borodin and Ran El-Yaniv. *Online computation and competitive analysis*. Cambridge University Press, 2005.

[BEYS21] Amanda Bower, Hamid Eftekhari, Mikhail Yurochkin, and Yuekai Sun. Individually fair rankings. In *International Conference on Learning Representations (ICLR)*, 2021.

[BGS+17] Peva Blanchard, Rachid Guerraoui, Julien Stainer, et al. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Conference on Neural Information Processing Systems (NIPS)*, 2017.

[BGW18] Asia J Biega, Krishna P Gummadi, and Gerhard Weikum. Equity of attention: Amortizing individual fairness in rankings. In *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2018.

[BHN19] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning*. fairmlbook.org, 2019. http://www.fairmlbook.org.

[BHPQ17] Avrim Blum, Nika Haghtalab, Ariel D Procaccia, and Mingda Qiao. Collaborative pac learning. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

[BIK+17] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017.

[BM02] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research (JMLR)*, 2002.

[BNBR19] Sina Baharlouei, Maher Nouiehed, Ahmad Beirami, and Meisam Razaviyayn. Rényi fair inference. In *International Conference on Learning Representations (ICLR)*, 2019.

[BNL12a] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *International Conference on Machine Learing (ICML)*, 2012.

[BNL12b] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *International Conference on Machine Learing (ICML)*, 2012.

[Bon19] Malte Bonart. Fair ranking in academic search, 2019.

[BR18] Battista Biggio and Fabio Roli. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 2018.

[BS20] Avrim Blum and Kevin Stangl. Recovering from biased data: Can fairness constraints improve accuracy? In *Foundations of Responsible Computing*, volume 156. Schloss Dagstuhl – Leibniz Center for Informatics, 2020.

[BSOG18] Robin Burke, Nasim Sonboli, and Aldo Ordonez-Gauger. Balanced neighborhoods for multi-sided fairness in recommendation. In *Conference on Fairness, Accountability and Transparency (FAccT)*, 2018.

[Bur17] Robin Burke. Multisided fairness for recommendation. *arXiv preprint arXiv:1707.00093*, 2017.

[BWKT14] Wei Bi, Liwei Wang, James T Kwok, and Zhuowen Tu. Learning to predict from crowdsourced data. In *UAI*, pages 82–91, 2014.

[Cas19] Carlos Castillo. Fairness and transparency in ranking. In *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2019.

[CBDF+99] Nicolo Cesa-Bianchi, Eli Dichterman, Paul Fischer, Eli Shamir, and Hans Ulrich Simon. Sample-efficient strategies for learning in the presence of noise. *Journal of the ACM (JACM)*, 1999.

[CGJ+19] Andrew Cotter, Maya Gupta, Heinrich Jiang, Nathan Srebro, Karthik Sridharan, Serena Wang, Blake Woodworth, and Seungil You. Training well-generalizing classifiers for fairness metrics and other data-dependent constraints. In *International Conference on Machine Learing (ICML)*, 2019.

[CHKV21] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. Fair classification with noisy protected attributes: A framework with provable guarantees. In *International Conference on Machine Learing (ICML)*, 2021.

[CHS20] Jaewoong Cho, Gyeongjo Hwang, and Changho Suh. A fair classifier using kernel density estimation. *Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[CKP09] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. In *International Conference on Data Mining Workshops (IDCMW)*, 2009.

[CKW08] Koby Crammer, Michael Kearns, and Jennifer Wortman. Learning from multiple sources. *Journal of Machine Learning Research (JMLR)*, 9(Aug):1757–1774, 2008.

[CLL+17] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

[CLM19] Sitan Chen, Jerry Li, and Ankur Moitra. Efficiently learning structured distributions from untrusted batches. In *ACM Symposium on Theory of Computing (STOC)*, 2019.

[CMM09] Liqun Chen, Chris J. Mitchell, and Andrew P. Martin, editors. *Trusted Computing, Second International Conference, Trust 2009, Oxford, UK, April 6-8, 2009, Proceedings*, volume 5471 of *Lecture Notes in Computer Science*, 2009.

[CMV20] L Elisa Celis, Anay Mehrotra, and Nisheeth K Vishnoi. Interventions for ranking in the presence of implicit bias. In *Conference on Fairness, Accountability and Transparency (FAccT)*, 2020.

[CMV21] L Elisa Celis, Anay Mehrotra, and Nisheeth K Vishnoi. Fair classification with adversarial perturbations. *arXiv preprint arXiv:2106.05964*, 2021.

[CNM+20] Hongyan Chang, Ta Duy Nguyen, Sasi Kumar Murakonda, Ehsan Kazemi, and Reza Shokri. On adversarial bias and the robustness of fair machine learning. *arXiv preprint arXiv:2006.08669*, 2020.

[CPG+19] Abhijnan Chakraborty, Gourab K Patro, Niloy Ganguly, Krishna P Gummadi, and Patrick Loiseau. Equality of coice: Towards fair representation in crowdsourced top-k recommendations. In *Conference on Fairness, Accountability and Transparency (FAccT)*, 2019.

[CS04] Andreas Christmann and Ingo Steinwart. On robustness properties of convex risk minimization methods for pattern recognition. *Journal of Machine Learning Research*, 2004.

[CSV17] Moses Charikar, Jacob Steinhardt, and Gregory Valiant. Learning from untrusted data. In *ACM Symposium on Theory of Computing (STOC)*, 2017.

[CSV18] L Elisa Celis, Damian Straszak, and Nisheeth K Vishnoi. Ranking with fairness constraints. In *International Colloquium on Automata, Languages, and Programming (ICALP)*. Schloss Dagstuhl – Leibniz Center for Informatics, 2018.

[CSX17] Yudong Chen, Lili Su, and Jiaming Xu. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems (POMACS)*, 1(2):1–25, 2017.

[CV10] Toon Calders and Sicco Verwer. Three naive Bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery (DMKD)*, 2010.

[CŽ13] Toon Calders and Indrė Žliobaitė. Why unbiased computational processes can lead to discriminative decision procedures. In *Discrimination and privacy in the information society*, pages 43–57. Springer, 2013.

[DADC18] Maria De-Arteaga, Artur Dubrawski, and Alexandra Chouldechova. Learning under selective labels in the presence of expert consistency. In *Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*, 2018.

[DCLT19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.

[DCM+12] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V Le, et al. Large scale distributed deep networks. In *Conference on Neural Information Processing Systems (NIPS)*, 2012.

[DHP+12] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Innovations in Theoretical Computer Science Conference (ITCS)*, 2012.

[DKK+16] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 655–664. IEEE, 2016.

[DKK+19a] Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high-dimensions without the computational intractability. *SIAM Journal on Computing*, 48(2):742–864, 2019.

[DKK+19b] Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. In *International Conference on Machine Learing (ICML)*, 2019.

[DSZO+15] Christopher M De Sa, Ce Zhang, Kunle Olukotun, Christopher Ré, and Christopher Ré. Taming the wild: A unified analysis of hogwild-style algorithms. In *Conference on Neural Information Processing Systems (NIPS)*. 2015.

[Fen17] Jiashi Feng. On fundamental limits of robust learning. *arXiv preprint arXiv:1703.10444*, 2017.

[FGC20] Riccardo Fogliato, Max G'Sell, and Alexandra Chouldechova. Fairness evaluation in presence of biased noisy labels. In *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2020.

[FKT+18] Golnoosh Farnadi, Pigi Kouki, Spencer K Thompson, Sriram Srinivasan, and Lise Getoor. A fairness-aware hybrid recommender system. *arXiv preprint arXiv:1809.09030*, 2018.

[FV13] Benoît Frénay and Michel Verleysen. Classification in the presence of label noise: a survey. *IEEE transactions on neural networks and learning systems*, 2013.

[FXM14] Jiashi Feng, Huan Xu, and Shie Mannor. Distributed robust learning. *arXiv preprint arXiv:1409.5937*, 2014.

[FYB18] Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. Mitigating sybils in federated learning poisoning. *arXiv preprint arXiv:1808.04866*, 2018.

[GAK19] Sahin Cem Geyik, Stuart Ambler, and Krishnaram Kenthapadi. Fairness-aware ranking in search & recommendation systems with application to linkedin talent search. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2019.

[GBY+18] Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. In *International Conference on Data Science and Advanced Analytics (DSAA)*, 2018.

[GDL20] Sruthi Gorantla, Amit Deshpande, and Anand Louis. Ranking for individual and group fairness simultaneously. *arXiv preprint arXiv:2010.06986*, 2020.

[HG17] Dan Hendrycks and Kevin Gimpel. Improving the generalization of adversarial training with domain adaptation. In *International Conference on Learning Representations (ICLR)*, 2017.

[HK20] Steve Hanneke and Samory Kpotufe. A no-free-lunch theorem for multi-task learning. *arXiv preprint arXiv:2006.15785*, 2020.

[HMPW16] Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Innovations in Theoretical Computer Science Conference (ITCS)*, 2016.

[HMWG18a] Dan Hendrycks, Mantas Mazeika, Duncan Wilson, and Kevin Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[HMWG18b] Dan Hendrycks, Mantas Mazeika, Duncan Wilson, and Kevin Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[HMZ18] Judy Hoffman, Mehryar Mohri, and Ningshan Zhang. Algorithms and theory for multiple-source adaptation. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[Hoe63] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58, 1963.

[HPS16] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2016.

[HSNL18] Tatsunori B Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learing (ICML)*, 2018.

[Hub64] Peter J Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 1964.

[Hub11] Peter J Huber. *Robust statistics*. Springer, 2011.

[Hun16] Elle Hunt. Tay, microsoft's ai chatbot, gets a crash course in racism from twitter. `https://www.theguardian.com/technology/2016/mar/24/tay-microsofts-ai-chatbot-gets-a-crash-course-in-racism-from-t` 2016. Accessed: 2021-11-06.

[HWBN20] Shuguang Han, Xuanhui Wang, Mike Bendersky, and Marc Najork. Learning-to-rank with BERT in TF-ranking. *arXiv preprint arXiv:2004.08476*, 2020.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[ICS+20] Alexey Ignatiev, Martin C Cooper, Mohamed Siala, Emmanuel Hebrard, and Joao Marques-Silva. Towards formal fairness in machine learning. In *International Conference on Principles and Practice of Constraint Programming*, 2020.

[IKL21] Eugenia Iofinova, Nikola Konstantinov, and Christoph H. Lampert. Flea: Provably fair multisource learning from unreliable training data. *arXiv preprint arXiv:2106.11732*, 2021.

[IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learing (ICML)*, 2015.

[Jan04] Svante Janson. Large deviations for sums of partly dependent random variables. *Random Structures & Algorithms*, 24(3):234–248, 2004.

[JEP+21] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 2021.

[JN20] Heinrich Jiang and Ofir Nachum. Identifying and correcting label bias in machine llearning. In *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2020.

[JO19] Ayush Jain and Alon Orlitsky. Robust learning of discrete distributions from batches. *arXiv preprint arXiv:1911.08532*, 2019.

[JO20] Ayush Jain and Alon Orlitsky. A general method for robust learning from batches. *arXiv preprint arXiv:2002.11099*, 2020.

[Joa02] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2002.

[KAAS12a] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Enhancement of the neutrality in recommendation. In *Decisions@ RecSys*, 2012.

[KAAS12b] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *European Conference on Machine Learning and Data Mining (ECML PKDD)*, 2012.

[KAAS14] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Correcting popularity bias by enhancing recommendation neutrality. In *RecSys Posters*, 2014.

[KAS11] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. Fairness-aware learning through regularization approach. In *11th International Conference on Data Mining Workshops*, 2011.

[KBDG04] Daniel Kifer, Shai Ben-David, and Johannes Gehrke. Detecting change in data streams. In *VLDB*, 2004.

[KCT20] Joon Sik Kim, Jiahao Chen, and Ameet Talwalkar. Fact: A diagnostic for group fairness trade-offs. In *International Conference on Machine Learing (ICML)*, 2020.

[KFAL20] Nikola Konstantinov, Elias Frantar, Dan Alistarh, and Christoph H. Lampert. On the sample complexity of adversarial multi-source pac learning. In *International Conference on Machine Learing (ICML)*, 2020.

[KL93] Michael Kearns and Ming Li. Learning in the presence of malicious errors. *SIAM Journal on Computing (SICOMP)*, 1993.

[KL17] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learing (ICML)*, 2017.

[KL19] Nikola Konstantinov and Christoph H. Lampert. Robust learning from untrusted sources. In *International Conference on Machine Learing (ICML)*, 2019.

[KL21a] Nikola Konstantinov and Christoph H. Lampert. Fairness-aware learning from corrupted data. *arXiv preprint arXiv:2102.06004*, 2021.

[KL21b] Nikola Konstantinov and Christoph H. Lampert. Fairness through regularization for learning to rank. *arXiv preprint arXiv:2102.05996*, 2021.

[KLRS17] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Conference on Neural Information Processing Systems (NIPS)*, 2017.

[KMR17] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. *Innovations in Theoretical Computer Science Conference (ITCS)*, 2017.

[KMZ20] Nathan Kallus, Xiaojie Mao, and Angela Zhou. Assessing algorithmic fairness with unobserved protected class using data combination. In *Conference on Fairness, Accountability and Transparency (FAccT)*, 2020.

[KTK12] Hiroshi Kajino, Yuta Tsuboi, and Hisashi Kashima. A convex formulation for learning from crowds. *Transactions of the Japanese Society for Artificial Intelligence*, 27(3):133–142, 2012.

[KVR19] Caitlin Kuhlman, MaryAnn VanValkenburg, and Elke Rundensteiner. FARE: Diagnostics for fair ranking using pairwise error metrics. In *International World Wide Web Conference (WWW)*, 2019.

[KZ18] Nathan Kallus and Angela Zhou. Residual unfairness in fair machine learning from prejudiced data. In *International Conference on Machine Learing (ICML)*, 2018.

[KZ19] Nathan Kallus and Angela Zhou. The fairness of risk scores beyond classification: Bipartite ranking and the xAUC metric. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[LBC+20] Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed H Chi. Fairness without demographics through adversarially reweighted learning. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[LBSS21] Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. Tilted empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2021.

[Liu11] Tie-Yan Liu. *Learning to Rank for Information Retrieval*. Springer Science & Business Media, 2011.

[Lo15] Danica Lo. When you Google image CEO, the first female photo on the results page is Barbie. `https://www.glamour.com/story/google-search-ceo`, 2015. Accessed: 2021-05-26.

[LZMV19] Alex Lamy, Ziyuan Zhong, Aditya K Menon, and Nakul Verma. Noise-tolerant fair classification. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[MAA+16] Frank McKeen, Ilya Alexandrovich, Ittai Anati, Dror Caspi, Simon Johnson, Rebekah Leslie-Hurd, and Carlos Rozas. Intel® software guard extensions (intel® sgx) support for dynamic memory management inside an enclave. In *Proceedings of the Hardware and Architectural Support for Security and Privacy 2016*, page 10, 2016.

[MC18] Bhaskar Mitra and Nick Craswell. An introduction to neural information retrieval. *Foundations and Trends in Information Retrieval*, 2018.

[MC21] Anay Mehrotra and L Elisa Celis. Mitigating bias in set selection with noisy protected attributes. In *Conference on Fairness, Accountability and Transparency (FAccT)*, 2021.

[MM12] Mehryar Mohri and Andres Munoz Medina. New analysis and algorithm for learning with drifting distributions. In *International Conference on Algorithmic Learning Theory (ALT)*, 2012.

[MMM19] Saeed Mahloujifar, Mohammad Mahmoody, and Ameer Mohammed. Universal multi-party poisoning attacks. In *International Conference on Machine Learing (ICML)*, 2019.

[MMR09] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation with multiple sources. In *Conference on Neural Information Processing Systems (NIPS)*, pages 1041–1048, 2009.

[MMR+17] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2017.

[MMS+19] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 2019.

[MNMG20] Ninareh Mehrabi, Muhammad Naveed, Fred Morstatter, and Aram Galstyan. Exacerbating algorithmic bias through fairness attacks. *arXiv preprint arXiv:2012.08723*, 2020.

[MOS20] Hussein Mozannar, Mesrob I Ohannessian, and Nathan Srebro. Fair learning with private demographic data. In *International Conference on Machine Learing (ICML)*, 2020.

[MPL15] Julian McAuley, Rahul Pandey, and Jure Leskovec. Inferring networks of substitutable and complementary products. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015.

[MR17] H. Brendan McMahan and Daniel Ramage. Federated learning: Collaborative machine learning without centralized training data. https://research.googleblog.com/2017/04/federated-learning-collaborative.html, 2017.

[MRT18] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT press, 2018.

[MS14] Yishay Mansour and Mariano Schain. Robust domain adaptation. *Annals of Mathematics and Artificial Intelligence*, 71(4):365–380, 2014.

[MSHJ20] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. Controlling fairness and bias in dynamic learning-to-rank. In *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2020.

[MSR08] Christopher D Manning, Hinrich Schütze, and Prabhakar Raghavan. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[MTSVDH15] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, 2015.

[MTV+20] Fatemehsadat Mireshghallah, Mohammadkazem Taram, Praneeth Vepakomma, Abhishek Singh, Ramesh Raskar, and Hadi Esmaeilzadeh. Privacy in deep learning: A survey. *arXiv preprint arXiv:2004.12254*, 2020.

[MW18a] Aditya Krishna Menon and Robert C Williamson. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency (FAccT)*, 2018.

[MW18b] Aditya Krishna Menon and Robert C Williamson. The cost of fairness in binary classification. In *Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*, 2018.

[NC19] Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with BERT. *arXiv preprint arXiv:1901.04085*, 2019.

[NCGW20] Harikrishna Narasimhan, Andrew Cotter, Maya R Gupta, and Serena Wang. Pairwise fairness for ranking and regression. In *AAAI Conference on Artificial Intelligence*, 2020.

[NRS+16] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human-generated machine reading comprehension dataset. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2016.

[PK19] Alexander Peysakhovich and Christian Kroer. Fair division without disparate impact. *arXiv preprint arXiv:1906.02775*, 2019.

[PL17] Anastasia Pentina and Christoph H. Lampert. Multi-task learning with labeled and unlabeled tasks. In *International Conference on Machine Learing (ICML)*, 2017.

[Pre82] Daryl Pregibon. Resistant fits for some commonly used logistic models with medical applications. *Biometrics*, pages 485–498, 1982.

[PSBR18] Adarsh Prasad, Arun Sai Suggala, Sivaraman Balakrishnan, and Pradeep Ravikumar. Robust estimation via robust gradient estimation. *arXiv preprint arXiv:1802.06485*, 2018.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[Qia18] Mingda Qiao. Do outliers ruin collaboration? In *International Conference on Machine Learing (ICML)*, 2018.

[QV18] Mingda Qiao and Gregory Valiant. Learning discrete distributions from untrusted batches. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 94. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[RBCJ09] Filip Radlinski, Paul N Bennett, Ben Carterette, and Thorsten Joachims. Redundancy, diversity and interdependent document relevance. In *International Conference on Research and Development in Information Retrieval (SIGIR)*, 2009.

[RDS+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[RFMZ20] Ashkan Rezaei, Rizal Fathony, Omid Memarrast, and Brian Ziebart. Fairness for robust log loss classification. In *AAAI Conference on Artificial Intelligence*, 2020.

[RLWS20] Yuji Roh, Kangwook Lee, Steven Euijong Whang, and Changho Suh. FR-Train: A mutual information-based approach to fair and robust training. In *International Conference on Machine Learing (ICML)*, 2020.

[Rob77] Stephen E Robertson. The probability ranking principle in ir. *Journal of Documentation*, 33(4):294–304, 1977.

[SBC20] David Solans, Battista Biggio, and Carlos Castillo. Poisoning attacks on algorithmic fairness. In *European Conference on Machine Learning and Data Mining (ECML PKDD)*, 2020.

[SCST17] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. In *Conference on Neural Information Processing Systems (NIPS)*, 2017.

[SHWH19] Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. Improving the generalization of adversarial training with domain adaptation. In *International Conference on Learning Representations (ICLR)*, 2019.

[SJ17] Ashudeep Singh and Thorsten Joachims. Equality of opportunity in rankings. In *Workshop on Prioritizing Online Content (WPOC) at NeurIPS*, 2017.

[SJ18] Ashudeep Singh and Thorsten Joachims. Fairness of exposure in rankings. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2018.

[SJ19] Ashudeep Singh and Thorsten Joachims. Policy learning for fairness in ranking. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[SKL17] Jacob Steinhardt, Pang Wei W Koh, and Percy S Liang. Certified defenses for data poisoning attacks. In *Conference on Neural Information Processing Systems (NIPS)*, pages 3517–3529, 2017.

[SL18] Remy Sun and Christoph H. Lampert. KS(conf): A light-weight test if a convnet operates outside of its specifications. In *German Conference on Pattern Recognition (GCPR)*, 2018.

[SNT+20] Muhammad Shafique, Mahum Naseer, Theocharis Theocharides, Christos Kyrkou, Onur Mutlu, Lois Orosa, and Jungwook Choi. Robust machine learning systems: Challenges, current trends, perspectives, and the road ahead. *IEEE Design & Test*, 2020.

[SS15] Reza Shokri and Vitaly Shmatikov. Privacy-preserving deep learning. In *ACM SIGSAC conference on computer and communications security*, 2015.

[SSBD14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge university press, 2014.

[Ste18] Harald Steck. Calibrated recommendations. In *Conference on Recommender Systems (RecSys)*, 2018.

[SZER+19] Piotr Sapiezynski, Wesley Zeng, Ronald E Robertson, Alan Mislove, and Christo Wilson. Quantifying the impact of user attentionon fair group representation in ranked lists. In *International World Wide Web Conference (WWW)*, 2019.

[SZS+14] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *International Conference on Learning Representations (ICLR)*, 2014.

[TNKB20] Bahar Taskesen, Viet Anh Nguyen, Daniel Kuhn, and Jose Blanchet. A distributionally robust approach to fair classification. *arXiv preprint arXiv:2007.09530*, 2020.

[TPT19] Virginia Tsintzou, Evaggelia Pitoura, and Panayiotis Tsaparas. Bias disparity in recommendation systems. In *Workshop on Recommendation in Multi-stakeholder Environments at RecSys*, 2019.

[TRO+19] Ki Hyun Tae, Yuji Roh, Young Hun Oh, Hyunsu Kim, and Steven Euijong Whang. Data cleaning for accurate, fair, and robust models: A big data-AI integration approach. In *International Workshop on Data Management for End-to-End Machine Learning (DEEM)*, 2019.

[Tuk60] John W Tukey. A survey of sampling from contaminated distributions. *Contributions to probability and statistics*, pages 448–485, 1960.

[TYFT20] Zilong Tan, Samuel Yeom, Matt Fredrikson, and Ameet Talwalkar. Learning fair representations for kernel models. In *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2020.

[UAG05] Nicolas Usunier, Massih R Amini, and Patrick Gallinari. Generalization error bounds for classifiers trained with interdependent data. *Conference on Neural Information Processing Systems (NIPS)*, 2005.

[Val84] Leslie G Valiant. A theory of the learnable. *Communications of the ACM*, 1984.

[Val85] Leslie G Valiant. Learning disjunction of conjunctions. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 1985.

[Vap13] Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer, 2013.

[VBC20] Robin Vogel, Aurélien Bellet, and Stéphan Clémençon. Learning fair scoring functions: Bipartite ranking under roc-based fairness constraints. *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2020.

[WBU11] Jason Weston, Samy Bengio, and Nicolas Usunier. WSABIE: Scaling up to large vocabulary image annotation. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2011.

[WGN+20] Serena Wang, Wenshuo Guo, Harikrishna Narasimhan, Andrew Cotter, Maya Gupta, and Michael I Jordan. Robust optimization for fairness with noisy protected groups. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[WGOS17] Blake Woodworth, Suriya Gunasekar, Mesrob I Ohannessian, and Nathan Srebro. Learning non-discriminatory predictors. In *Conference on Computational Learning Theory (COLT)*, 2017.

[WLC+10] Paul Wais, Shivaram Lingamneni, Duncan Cook, Jason Fennell, Benjamin Goldenberg, Daniel Lubarov, David Marin, and Hari Simons. Towards building a high-quality workforce with mechanical turk. In *NIPS Workshop on Computational Social Science and the Wisdom of Crowds*, 2010.

[WLL20] Jialu Wang, Yang Liu, and Caleb Levy. Fair classification with group-dependent label noise. *arXiv preprint arXiv:2011.00379*, 2020.

[WM19] Robert Williamson and Aditya Menon. Fairness risk measures. In *International Conference on Machine Learing (ICML)*, 2019.

[WMP+03] Douglas Wahlsten, Pamela Metten, Tamara J Phillips, Stephen L Boehm, Sue Burkhart-Kasch, Janet Dorow, Sharon Doerksen, Chris Downing, Jennifer Fogarty, Kristina Rodd-Henricks, et al. Different data from different labs: Lessons from studies of gene–environment interaction. *Journal of neurobiology*, 54(1):283–311, 2003.

[WPT19] Michael Wick, Swetasudha Panda, and Jean-Baptiste Tristan. Unlocking fairness: a trade-off revisited. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[WZW18] Yongkai Wu, Lu Zhang, and Xintao Wu. On discrimination discovery and removal in ranked data using causal graph. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2018.

[Xie17] Tianpei Xie. *Robust Learning from Multiple Information Sources*. PhD thesis, University of Michigan, 2017.

[XLSA18] Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning - a comprehensive evaluation of the good, the bad and the ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 2018.

[YCKB18] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learing (ICML)*, 2018.

[YCKB19] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Defending against saddle point attack in Byzantine-robust distributed learning. In *International Conference on Machine Learing (ICML)*, 2019.

[YCRB18] Dong Yin, Yudong Chen, Kannan Ramchandran, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. *International Conference on Machine Learing (ICML)*, 2018.

[YDJ19] Himank Yadav, Zhengxiao Du, and Thorsten Joachims. Fair learning-to-rank from implicit feedback. *arXiv preprint arXiv:1911.08054*, 2019.

[YGS19] Ke Yang, Vasilis Gkatzelis, and Julia Stoyanovich. Balanced ranking with diversity constraints. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2019.

[YS17] Ke Yang and Julia Stoyanovich. Measuring fairness in ranked outputs. In *Scientific and Statistical Database Management Conference (SSDBM)*, 2017.

[ZBC+17] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. FA*IR: A fair top-k ranking algorithm. In *Conference on Information and Knowledge Management (CIKM)*, 2017.

[ZC20] Meike Zehlike and Carlos Castillo. Reducing disparate exposure in ranking: A learning to rank approach. In *International World Wide Web Conference (WWW)*, 2020.

[ZHC18] Ziwei Zhu, Xia Hu, and James Caverlee. Fairness-aware tensor-based recommendation. In *Conference on Information and Knowledge Management (CIKM)*, 2018.

[ZIK17] Guoxi Zhang, Tomoharu Iwata, and Hisashi Kashima. Robust multi-view topic modeling by incorporating detecting anomalies. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2017.

[ZL17] Alexander Zimin and Christoph H. Lampert. Learning theory for conditional risk minimization. In *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2017.

[ZLM18] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 2018.

[ZSCK20] Meike Zehlike, Tom Sühr, Carlos Castillo, and Ivan Kitanovski. Fairsearch: A tool for fairness in ranked search results. In *International World Wide Web Conference (WWW)*, 2020.

[ZVRG17] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Conference on Uncertainty in Artificial Intelligence (AISTATS)*, 2017.

[ZXXS17] Jing Zhao, Xijiong Xie, Xin Xu, and Shiliang Sun. Multi-view learning overview: Recent progress and new challenges. *Information Fusion*, 38:43–54, 2017.

[ZZY12] Chao Zhang, Lei Zhang, and Jieping Ye. Generalization bounds for domain adaptation. In *Conference on Neural Information Processing Systems (NIPS)*, 2012.

[ZZY13] Chao Zhang, Lei Zhang, and Jieping Ye. Generalization bounds for domain adaptation. *arXiv preprint arXiv:1304.1574*, 2013.

APPENDIX A

Proofs from Chapter 3

In this chapter we present the proofs of all results from Chapter 3. In particular, in Section A.1 we prove our upper bound result and its corollaries. We then move on to the lower bounds and prove Theorems 5 and 6 in Sections A.2 and A.3 respectively.

A.1 Proof of Theorem 4 and its corollaries

Theorem 4. *Let $N, m, k \in \mathbb{N}$ be integers, such that $k \in (N/2, N]$. Let $\alpha = \frac{N-k}{N} < \frac{1}{2}$ be the proportion of corrupted sources. Assume that $\mathcal{H}$ has the uniform convergence property with rate function s. Then there exists a learner $\mathcal{L} : (\mathcal{X} \times \mathcal{Y})^{N \times m} \to \mathcal{H}$ with the following two properties.*

(a) Let G be a fixed subset of $[N]$ of size $|G| = k$. For $S' = \{S'_1, \dots, S'_N\} \overset{i.i.d.}{\sim} \mathcal{D}$, with probability at least $1 - \delta$ over the sampling of S':

$$\mathcal{R}(\mathcal{L}(\mathfrak{A}(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) \leq 2s\Big(km, \frac{\delta}{2}, S_G\Big) + 6\alpha \max_{i \in [N]} s\Big(m, \frac{\delta}{2N}, S_i\Big), \tag{A.1}$$

uniformly against all fixed-set adversaries with preserved set G, where $S = \{S_1, \dots, S_N\} = \mathfrak{A}(S')$ is the dataset modified the adversary and $S_G = \cup_{i \in G} S_i$ is the set of all uncorrupted data.

(b) For $S' = \{S'_1, \dots, S'_N\} \overset{i.i.d.}{\sim} \mathcal{D}$, with probability at least $1 - \delta$ over the sampling of S':

$$\mathcal{R}(\mathcal{L}(\mathfrak{A}(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) \leq 2s\Big(km, \frac{\delta}{2\binom{N}{k}}, S_G\Big) + 6\alpha \max_{i \in [N]} s\Big(m, \frac{\delta}{2N}, S_i\Big), \tag{A.2}$$

uniformly against all flexible-set adversaries with preserved size k, where $S = \{S_1, \dots, S_N\} = \mathfrak{A}(S')$ is the dataset returned by the adversary, G is the set of sources not modified by the adversary and $S_G = \cup_{i \in G} S_i$ is the set of all uncorrupted data.

Proof. Denote by $S'_i = \{(x'_{i,1}, y'_{i,1}), \dots, (x'_{i,m}, y'_{i,m})\}$ for $i = 1, \dots, N$ the initial datasets and by $S_i = \{(x_{i,1}, y_{i,1}), \dots, (x_{i,m}, y_{i,m})\}$ for $i = 1, \dots, N$ the datasets after the modifications of the adversary. As explained in the main body of the paper, we denote by:

$$\widehat{\mathcal{R}}_i(h) = \frac{1}{m} \sum_{j=1}^{m} \ell(h(x_{i,j}), y_{i,j}) \tag{A.3}$$

the empirical risk of any hypothesis $h \in \mathcal{H}$ on the dataset S_i and by:

$$d_{\mathcal{H}}(S_i, S_j) = \sup_{h \in \mathcal{H}} |\widehat{\mathcal{R}}_i(h) - \widehat{\mathcal{R}}_j(h)| \tag{A.4}$$

the empirical discrepancy between the datasets S_i and S_j.

We show that a learner that first runs a certain filtering algorithm (Algorithm A.1) based on the discrepancy metric and then performs empirical risk minimization on the remaining data to compute a hypothesis satisfies the properties stated in the theorem. The full algorithm for the learner is therefore given in Algorithm A.2.

(a) The key idea of the proof is that the clean sources are close to each other with high probability, so they get selected when running Algorithm A.1. On the other hand, if a bad source has been selected, it must be close to *at least one* of the good sources, so it can not have too bad an effect on the empirical risk.

For all $i \in G$, let $\mathcal{E}_i$ be the event that:

$$\sup_{h \in \mathcal{H}} \left| \mathcal{R}(h) - \widehat{\mathcal{R}}_i(h) \right| \leq s\left(m, \frac{\delta}{2N}, S_i\right). \tag{A.5}$$

Further, let $\mathcal{E}_G$ be the event that:

$$\sup_{h \in \mathcal{H}} \left| \mathcal{R}(h) - \widehat{\mathcal{R}}_G(h) \right| \leq s\left(km, \frac{\delta}{2}, S_G\right), \tag{A.6}$$

where

$$\widehat{\mathcal{R}}_G(h) = \frac{1}{km} \sum_{i \in G} \sum_{j=1}^{m} \ell(h(x_{i,j}), y_{i,j}).$$

Denote by $\mathcal{E}_i^c$ and $\mathcal{E}_G^c$ the complements of these events. Then we know that $\mathbb{P}\left(\mathcal{E}_G^c\right) \leq \frac{\delta}{2}$, and $\mathbb{P}\left(\mathcal{E}_i^c\right) \leq \frac{\delta}{2N}$ for all $i \in G$. Therefore, if $\mathcal{E} = \mathcal{E}_G \wedge (\wedge_{i \in G} \mathcal{E}_i)$, we have:

$$\mathbb{P}\left(\mathcal{E}^c\right) = \mathbb{P}\left(\mathcal{E}_G^c \vee (\vee_{i \in G} \mathcal{E}_i^c)\right) \leq \mathbb{P}\left(\mathcal{E}_G^c\right) + \sum_{i \in G} \mathbb{P}\left(\mathcal{E}_i^c\right) \leq \frac{\delta}{2} + k\frac{\delta}{2N} \leq \delta. \tag{A.7}$$

Hence, the probability of the event $\mathcal{E}$ that all of (A.5) and (A.6) hold, is at least $1 - \delta$. We now show that under the event $\mathcal{E}$, Algorithm A.2 returns a hypothesis that satisfies the condition in (a).

Algorithm A.1: Dataset filtering for robust multi-source learning

Inputs: Datasets $S_1, \ldots, S_N$
Initialize $\mathrm{T} = \{\}$ // trusted sources
for $i = 1, \ldots, N$ **do**
  if $d_{\mathcal{H}}\Big(S_i, S_j\Big) \leq s\left(m, \frac{\delta}{2N}, S_i\right) + s\left(m, \frac{\delta}{2N}, S_j\right)$,
    for at least $\lfloor \frac{N}{2} \rfloor$ values of $j \neq i$, **then**
    $\mathrm{T} = \mathrm{T} \cup \{i\}$
  end if
end for
Return: $\bigcup_{i \in \mathrm{T}} S_i$ // indices of the trusted sources

Algorithm A.2: ERM on the trusted sources

Inputs: Datasets $S_1, \ldots, S_N$
Run Algorithm A.1 to compute $S_{\mathrm{T}} = \bigcup_{i \in \mathrm{T}} S_i$
Compute $h^{\mathfrak{A}} = \operatorname{argmin}_{h \in \mathcal{H}} \frac{1}{|S_{\mathrm{T}}|} \sum_{(x,y) \in S_{\mathrm{T}}} \ell(h(x), y)$
Return: $h^{\mathfrak{A}}$

Indeed, fix an arbitrary fixed-set adversary $\mathfrak{A}$ with preserved set G. Whenever $\mathcal{E}$ holds, for all $i, j \in G$ we have:

$$\begin{aligned} d_{\mathcal{H}}\left(S_i, S_j\right) = \sup_{h \in \mathcal{H}}(|\widehat{\mathcal{R}}_i(h) - \widehat{\mathcal{R}}_j(h)|) &\leq \sup_{h \in \mathcal{H}}\left(|\widehat{\mathcal{R}}_i(h) - \mathcal{R}(h)|\right) + \sup_{h \in \mathcal{H}}\left(|\mathcal{R}(h) - \widehat{\mathcal{R}}_j(h)|\right) \\ &\leq s\left(m, \frac{\delta}{2N}, S_i\right) + s\left(m, \frac{\delta}{2N}, S_j\right). \end{aligned} \tag{A.8}$$

Now since $k \geq \lfloor \frac{N}{2} \rfloor + 1$, we get that $G \subseteq \mathrm{T}$. Moreover, for any $i \in \mathrm{T} \setminus G$, there exists *at least one* $j \in G$, such that $d_{\mathcal{H}}(S_i, S_j) \leq s\left(m, \frac{\delta}{2N}, S_i\right) + s\left(m, \frac{\delta}{2N}, S_j\right)$. For any $i \in \mathrm{T} \setminus G$, denote by $f(i)$ the smallest such j. Therefore, for any $i \in (\mathrm{T} \setminus G)$:

$$\begin{aligned} |\widehat{\mathcal{R}}_i(h) - \mathcal{R}(h)| &\leq |\widehat{\mathcal{R}}_i - \widehat{\mathcal{R}}_{f(i)}(h)| + |\widehat{\mathcal{R}}_{f(i)}(h) - \mathcal{R}(h)| \\ &\leq d_{\mathcal{H}}\left(S_i, S_{f(i)}\right) + s\left(m, \frac{\delta}{2N}, S_{f(i)}\right) \\ &\leq s\left(m, \frac{\delta}{2N}, S_i\right) + 2s\left(m, \frac{\delta}{2N}, S_{f(i)}\right) \end{aligned} \tag{A.9}$$

Denote by

$$\widehat{\mathcal{R}}_{\mathrm{T}}(h) = \frac{1}{|\mathrm{T}|} \sum_{i \in \mathrm{T}} \widehat{\mathcal{R}}_i(h) = \frac{1}{|S_{\mathrm{T}}|} \sum_{(x,y) \in S_{\mathrm{T}}} \ell(h(x), y) \tag{A.10}$$

the loss over all the trusted data. Then for any $h \in \mathcal{H}$ we have:

$$\begin{aligned} \left|\widehat{\mathcal{R}}_{\mathrm{T}}(h) - \mathcal{R}(h)\right| &\leq \frac{1}{|\mathrm{T}|m}\left(\left|\sum_{i \in G}\sum_{l=1}^{m}\left(\ell(h(x_{i,l}), y_{i,l}) - \mathcal{R}(h)\right)\right|\right. \\ &\quad \left. + \sum_{i \in (\mathrm{T} \setminus G)}\left|\sum_{l=1}^{m}\left(\ell(h(x_{i,l}), y_{i,l}) - \mathcal{R}(h)\right)\right|\right) \\ &= \frac{k}{|\mathrm{T}|}\left|\widehat{\mathcal{R}}_G(h) - \mathcal{R}(h)\right| + \frac{1}{|\mathrm{T}|}\sum_{i \in (\mathrm{T} \setminus G)}\left|\widehat{\mathcal{R}}_i(h) - \mathcal{R}(h)\right| \\ &\leq \frac{k}{|\mathrm{T}|}s\left(km, \frac{\delta}{2}, S_G\right) + \frac{1}{|\mathrm{T}|}\sum_{i \in (\mathrm{T} \setminus G)}\left|\widehat{\mathcal{R}}_i(h) - \mathcal{R}(h)\right| \\ &\leq \frac{k}{|\mathrm{T}|}s\left(km, \frac{\delta}{2}, S_G\right) + \frac{1}{|\mathrm{T}|}\sum_{i \in (\mathrm{T} \setminus G)}\left(s\left(m, \frac{\delta}{2N}, S_i\right) + 2s\left(m, \frac{\delta}{2N}, S_{f(i)}\right)\right) \\ &\leq \frac{k}{|\mathrm{T}|}s\left(km, \frac{\delta}{2}, S_G\right) + 3\frac{|\mathrm{T}| - k}{|\mathrm{T}|}\max_{i \in [N]} s\left(m, \frac{\delta}{2N}, S_i\right) \\ &\leq s\left(km, \frac{\delta}{2}, S_G\right) + 3\frac{N - k}{N}\max_{i \in [N]} s\left(m, \frac{\delta}{2N}, S_i\right) \end{aligned}$$

Finally, let $h^* = \operatorname{argmin}_{h\in\mathcal{H}} \mathcal{R}(h)$ and $h^{\mathfrak{A}} = \mathcal{L}(\mathfrak{A}(S')) = \operatorname{argmin}_{h\in\mathcal{H}} \widehat{\mathcal{R}}_{\mathrm{T}}(h)$. Then:

$$\begin{aligned}\mathcal{R}(h^{\mathfrak{A}}) - \mathcal{R}(h^*) &= \left(\mathcal{R}(h^{\mathfrak{A}}) - \widehat{\mathcal{R}}_{\mathrm{T}}(h^{\mathfrak{A}})\right) + \left(\widehat{\mathcal{R}}_{\mathrm{T}}(h^{\mathfrak{A}}) - \mathcal{R}(h^*)\right)\\ &\leq \left(\mathcal{R}(h^{\mathfrak{A}}) - \widehat{\mathcal{R}}_{\mathrm{T}}(h^{\mathfrak{A}})\right) + \left(\widehat{\mathcal{R}}_{\mathrm{T}}(h^*) - \mathcal{R}(h^*)\right)\\ &\leq 2\sup_{h\in\mathcal{H}}\left|\widehat{\mathcal{R}}_{\mathrm{T}}(h) - \mathcal{R}(h)\right|. \end{aligned} \tag{A.11}$$

Since we showed this result for an arbitrary fixed-set adversary with preserved set G, the result follows.

(b) The crucial difference in the case of the flexible-set adversary is that the set G is chosen after the clean data is observed. We thus need concentration results for *all* of the subsets of $[N]$ of size k, as well as all individual sources.

For all $i \in [N]$, let $\mathcal{E}_i$ be the event that:

$$\sup_{h\in\mathcal{H}}\left|\mathcal{R}(h) - \widehat{\mathcal{R}}'_i(h)\right| \leq s\left(m, \frac{\delta}{2N}, S'_i\right), \tag{A.12}$$

where

$$\widehat{\mathcal{R}}'_i = \frac{1}{m}\sum_{j=1}^{m} \ell(h(x'_{i,j}), y'_{i,j}) \tag{A.13}$$

Further, for *any* $A \subseteq [N]$ of size $|A| = k$, let $\mathcal{E}_A$ be the event that:

$$\sup_{h\in\mathcal{H}}\left|\mathcal{R}(h) - \widehat{\mathcal{R}}'_A(h)\right| \leq s\left(km, \frac{\delta}{2\binom{N}{k}}, S'_A\right), \tag{A.14}$$

where $S'_A = \cup_{i\in A} S'_i$ and

$$\widehat{\mathcal{R}}'_A(h) = \frac{1}{km}\sum_{i\in A}\sum_{l=1}^{m} \ell(h(x'_{i,l}), y'_{i,l}). \tag{A.15}$$

Then we know that $\mathbb{P}\left(\mathcal{E}_i^c\right) \leq \frac{\delta}{2N}$ for all $i \in [N]$ and $\mathbb{P}\left(\mathcal{E}_G^c\right) \leq \frac{\delta}{2\binom{N}{k}}$ for all $A \subseteq [N]$ with $|A| = k$. Therefore, if $\mathcal{E} = (\wedge_A \mathcal{E}_A) \wedge \left(\wedge_{i\in[N]}\mathcal{E}_i\right)$, we have:

$$\mathbb{P}\left(\mathcal{E}^c\right) = \mathbb{P}\left((\vee_A \mathcal{E}_A^c) \vee \left(\vee_{i\in[N]}\mathcal{E}_i^c\right)\right) \leq \sum_A \mathbb{P}\left(\mathcal{E}_A^c\right) + \sum_{i\in[N]} \mathbb{P}\left(\mathcal{E}_i^c\right) \leq \binom{N}{k}\frac{\delta}{2\binom{N}{k}} + N\frac{\delta}{2N} = \delta. \tag{A.16}$$

Hence, the probability of the event $\mathcal{E}$ that all of (A.12) and (A.14) hold, is at least $1-\delta$. In particular, under $\mathcal{E}$:

$$\sup_{h\in\mathcal{H}}\left|\mathcal{R}(h) - \widehat{\mathcal{R}}_G(h)\right| = \sup_{h\in\mathcal{H}}\left|\mathcal{R}(h) - \widehat{\mathcal{R}}'_G(h)\right| \leq s\left(km, \frac{\delta}{2\binom{N}{k}}, S'_G\right) = s\left(km, \frac{\delta}{2\binom{N}{k}}, S_G\right) \tag{A.17}$$

and

$$\sup_{h\in\mathcal{H}}\left|\mathcal{R}(h) - \widehat{\mathcal{R}}_i(h)\right| = \sup_{h\in\mathcal{H}}\left|\mathcal{R}(h) - \widehat{\mathcal{R}}'_i(h)\right| \leq s\left(m, \frac{\delta}{2N}, S'_i\right) = s\left(m, \frac{\delta}{2N}, S_i\right), \tag{A.18}$$

for all $i \in G$.

Now, for any flexible-set adversary with preserved size k, the same argument as in (a) shows that:

$$\mathcal{R}(h^{\mathfrak{A}}) - \mathcal{R}(h^*) \leq 2s\left(km, \frac{\delta}{2\binom{N}{k}}, S_G\right) + 6\frac{N-k}{N}\max_{i\in[N]} s\left(m, \frac{\delta}{2N}, S_i\right) \tag{A.19}$$

holds under the event $\mathcal{E}$. □

We now show how to obtain data-dependent guarantees, via the notion of Rademacher complexity. Let

$$\mathfrak{R}_S\left(\ell \circ \mathcal{H}\right) = \mathbb{E}_\sigma\left(\sup_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i \ell(h(x_i), y_i)\right) \tag{A.20}$$

be the (empirical) Rademacher complexity of $\mathcal{H}$ with respect to the loss function ℓ on a sample $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. Here $\{\sigma_i\}_{i=1}^n$ are i.i.d. Rademacher random variables. Let $S_G = \bigcup_{i\in G} S_i$, $\mathfrak{R}_i = \mathfrak{R}_{S_i}(\ell \circ \mathcal{H})$ and $\mathfrak{R}_G = \mathfrak{R}_{S_G}(\ell \circ \mathcal{H})$. Assume also that the loss function ℓ is bounded, so that for some constant $M > 0$, $\ell(y_1, y_2) \leq M$ for all $y_1, y_2 \in \mathcal{Y}$. Then we have:

Corollary 1. *In the setup of Theorem 4, against a fixed-set adversary, it holds that*

$$\mathcal{R}(\mathcal{L}(\mathfrak{A}(S^{'}))) - \min_{h\in\mathcal{H}} \mathcal{R}(h) \leq 4\mathfrak{R}_G + 6M\sqrt{\frac{\log(\frac{4}{\delta})}{2km}} + \alpha\Big(18M\sqrt{\frac{\log\left(\frac{4N}{\delta}\right)}{2m}} + 12\max_{i\in[N]}\mathfrak{R}_i\Big). \tag{A.21}$$

Proof. We use the standard generalization bound based on Rademacher complexity. Assume that $S = \{(x_1, y_1), \ldots, (x_n, y_n)\} \sim \mathcal{D}$, then with probability at least $1 - \delta$ over the data [MRT18]:

$$\sup_{h\in\mathcal{H}} |\mathbb{E}\left(\ell(h(x), y)\right) - \frac{1}{n}\sum_{i=1}^{n} \ell(h(x_i), y_i)| \leq 2\mathfrak{R}_S\left(\ell \circ \mathcal{H}\right) + 3M\sqrt{\frac{\log\left(\frac{2}{\delta}\right)}{2n}}. \tag{A.22}$$

Substituting into the result of Theorem 4 gives the result. □

Corollary 2. *In the setup of Theorem 4, against a flexible-set adversary, it holds that*

$$\mathcal{R}(\mathcal{L}(\mathfrak{A}(S^{'}))) - \min_{h\in\mathcal{H}} \mathcal{R}(h) \leq 4\mathfrak{R}_G + 12\alpha\max_{i\in[N]}\mathfrak{R}_i + \widetilde{\mathcal{O}}\left(\frac{\sqrt[4]{\alpha}}{\sqrt{m}}\right). \tag{A.23}$$

Proof. Using the concentration result from Corollary 1 and $\binom{N}{k} = \binom{N}{(1-\alpha)N} = \binom{N}{\alpha N} \leq 2^{H(\alpha)N}$, where $H(p) = -p\log_2(p) - (1-p)\log_2(1-p)$ is the binary entropy function, we obtain:

$$\begin{aligned}\mathcal{R}(\mathcal{L}(\mathfrak{A}(S'))) - \min_{h\in\mathcal{H}} \mathcal{R}(h) &\leq 4\mathfrak{R}_G + 6M\sqrt{\frac{\log\left(\frac{4\binom{N}{k}}{\delta}\right)}{2km}} + \alpha\left(18M\sqrt{\frac{\log\left(\frac{4N}{\delta}\right)}{2m}} + 12\max_{i\in[N]}\mathfrak{R}_i\right) \\ &= 4\mathfrak{R}_G + 6M\sqrt{\frac{\log(\binom{N}{k})}{2km} + \frac{\log(\frac{4}{\delta})}{2km}}\end{aligned}$$

$$
\begin{aligned}
&+ \alpha \left(18M \sqrt{\frac{\log\left(\frac{4N}{\delta}\right)}{2m}} + 12 \max_{i \in [N]} \mathfrak{R}_i \right) \\
&\leq 4\mathfrak{R}_G + 6M \sqrt{\frac{H(\alpha) N \log(2)}{2(1-\alpha)Nm} + \frac{\log(\frac{4}{\delta})}{2(1-\alpha)Nm}} \\
&+ \alpha \left(18M \sqrt{\frac{\log\left(\frac{4N}{\delta}\right)}{2m}} + 12 \max_{i \in [N]} \mathfrak{R}_i \right) \\
&\leq 4\mathfrak{R}_G + 12\alpha \max_{i \in [N]} \mathfrak{R}_i + \widetilde{\mathcal{O}} \left(\frac{\sqrt[4]{\alpha}}{\sqrt{m}} \right)
\end{aligned}
$$

where for the last inequality we used $H(\alpha) \leq 2\sqrt{\alpha(1-\alpha)}$, $1-\alpha \in (\frac{1}{2}, 1]$ and $\sqrt[4]{\alpha} > \alpha$. $\square$

For the case of binary classifiers, we also provide a simpler bound in terms of the VC dimension of $\mathcal{H}$.

Corollary 3. *Assume that $Y = \{-1, 1\}$ and that $\mathcal{H}$ has finite VC-dimension d. Then:*

(a) In the case of the fixed-set adversary there exists a universal constant C, such that:

$$
\mathcal{R}(\mathcal{L}(\mathfrak{A}(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) \leq 2C \sqrt{\frac{d}{km}} + 2\sqrt{\frac{2\log(\frac{4}{\delta})}{km}} + \alpha \left(6C \sqrt{\frac{d}{m}} + 6\sqrt{\frac{2\log\left(\frac{4N}{\delta}\right)}{m}} \right). \tag{A.24}
$$

(b) In the case of the flexible-set adversary:

$$
\mathcal{R}(\mathcal{L}(\mathfrak{A}(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) \leq \mathcal{O} \left(\sqrt{\frac{d}{km}} + \frac{\sqrt[4]{\alpha}}{\sqrt{m}} + \alpha \sqrt{\frac{d}{m}} + \alpha \sqrt{\frac{\log(N)}{m}} \right). \tag{A.25}
$$

Proof. (a) Whenever $\mathcal{H}$ is of finite VC-dimension d, there exists a constant C, such that the following generalization bound holds [BBL04]:

$$
\sup_{h \in \mathcal{H}} \left| \mathbb{E}\left(\ell(h(x), y)\right) - \frac{1}{n} \sum_{i=1}^{n} \ell(h(x_i), y_i) \right| \leq C \sqrt{\frac{d}{n}} + \sqrt{\frac{2\log\left(\frac{2}{\delta}\right)}{n}} \tag{A.26}
$$

and hence $\mathcal{H}$ has the uniform convergence property with rate function $s = C\sqrt{\frac{d}{n}} + \sqrt{\frac{2\log\left(\frac{2}{\delta}\right)}{n}}$. Substituting into the result of Theorem 4 gives the result.

(b) Using the concentration result from (a) and $\binom{N}{k} = \binom{N}{(1-\alpha)N} = \binom{N}{\alpha N} \leq 2^{H(\alpha)N}$, where $H(p) = -p\log_2(p) - (1-p)\log_2(1-p)$ is the binary entropy function, we obtain:

$$
\mathcal{R}(\mathcal{L}(\mathfrak{A}(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) \leq 2C \sqrt{\frac{d}{km}} + 2\sqrt{\frac{2\log(\frac{4\binom{N}{k}}{\delta})}{km}} + \alpha \left(6C \sqrt{\frac{d}{m}} + 6\sqrt{\frac{2\log\left(\frac{4N}{\delta}\right)}{m}} \right)
$$

$$
\begin{aligned}
&= 2C\sqrt{\frac{d}{km}} + 2\sqrt{\frac{2\log(\binom{N}{k})}{km} + \frac{2\log(\frac{4}{\delta})}{km}} \\
&+ \alpha\left(6C\sqrt{\frac{d}{m}} + 6\sqrt{\frac{2\log\left(\frac{4N}{\delta}\right)}{m}}\right) \\
&\leq 2C\sqrt{\frac{d}{km}} + 2\sqrt{\frac{2H(\alpha)N\log(2)}{(1-\alpha)Nm} + \frac{2\log(\frac{4}{\delta})}{(1-\alpha)Nm}} \\
&+ \alpha\left(6C\sqrt{\frac{d}{m}} + 6\sqrt{\frac{2\log\left(\frac{4N}{\delta}\right)}{m}}\right) \\
&\leq \mathcal{O}\left(\sqrt{\frac{d}{km}} + \frac{\sqrt[4]{\alpha}}{\sqrt{m}} + \alpha\sqrt{\frac{d}{m}} + \alpha\sqrt{\frac{\log(N)}{m}}\right),
\end{aligned}
$$

where for the last inequality we used $H(\alpha) \leq 2\sqrt{\alpha(1-\alpha)}$ and $1-\alpha \in (\frac{1}{2}, 1]$. □

A.2 Proof of Theorem 5

Theorem 5. *Let $\mathcal{H}$ be a non-trivial hypothesis space. Let m and N be any positive integers and let G be a fixed subset of $[N]$ of size $k \in \{1, \ldots, N-1\}$. Let $\mathcal{L} : (\mathcal{X} \times \mathcal{Y})^{N\times m} \to \mathcal{H}$ be a multi-source learner that acts by merging the data from all sources and then calling a single-source learner. Let $S' \in (\mathcal{X} \times \mathcal{Y})^{N\times m}$ be drawn i.i.d. from $\mathcal{D}$. Then there exists a distribution $\mathcal{D}$ with $\min_{h\in\mathcal{H}} \mathcal{R}(h) = 0$ and a fixed-set adversary $\mathcal{A}$ with index set G, such that:*

$$
\mathbb{P}_{S'\sim\mathcal{D}}\Big(\mathcal{R}\big(\mathcal{L}(\mathcal{A}(S')\big) > \frac{\alpha}{8(1-\alpha)}\Big) > \frac{1}{20}, \tag{A.27}
$$

where $\alpha = \frac{N-k}{N}$ is the power of the adversary.

We use a similar proof technique as in the lower bound results in [BEK02] and in the classic sample complexity lower bound for binary classification, e.g. Theorem 3.20 in [MRT18]. An overview is as follows. Consider a distribution on $\mathcal{X}$ that has support only at two points - the common point x_1 and the rare point x_2. Take $\mathbb{P}(x_2) = \mathcal{O}(\frac{\alpha}{1-\alpha})$. Then the expected number of occurrences of the point x_2 in G is $\mathcal{O}\left(\frac{\alpha}{1-\alpha}(1-\alpha)Nm\right) = \mathcal{O}\left(\alpha Nm\right)$. Thus, one can show that with constant probability the number of x_2's in G is at most αNm and hence the adversary (that has access to exactly αNm points in total) can insert the same number of x_2's, but wrongly labelled, into the final dataset. Therefore, based on the union of the corrupted datasets, no algorithm can guess with probability greater than $1/2$ what the true label of x_2 was.

Proof. We prove that there exists a distribution $\mathcal{D}$ on $\mathcal{X}$ and a labelling function $f \in \mathcal{H}$, such that the resulting joint distribution on $\mathcal{X} \times \mathcal{Y}$, defined by $x \sim \mathcal{D}$ and $y = f(x)$, satisfies the desired property.

Without loss of generality, let $G = [1, 2, \ldots, k]$. Since $\mathcal{H}$ is non-trivial, there exist $h_1, h_2 \in \mathcal{H}$ and $x_1, x_2 \in \mathcal{X}$, such that $h_1(x_1) = h_2(x_1)$, while $h_1(x_2) = 1$, but $h_2(x_2) = -1$. Consider

the following distribution on $\mathcal{X}$:

$$\mathbb{P}_{\mathcal{D}}(x_1) = 1 - 4\epsilon \quad \text{and} \quad \mathbb{P}_{\mathcal{D}}(x_2) = 4\epsilon, \tag{A.28}$$

where $\epsilon = \frac{1}{8}\frac{\alpha}{1-\alpha}$. Assume that the points are labelled by a function $f \in \mathcal{H}$ (to be chosen later as either h_1 or h_2). Denote the initial uncorrupted collection of datasets by $S' = (S'_1, \ldots, S'_N)$, with $S'_i = \{(x'_{i,1}, f(x'_{i,1})), \ldots, (x'_{i,m}, f(x'_{i,m}))\}$ and $x'_{i,j}$ being i.i.d. samples from $\mathcal{D}$.

First we show that with constant probability the point x_2 appears at most $\alpha N m$ times in G. Indeed, let C be this number of appearances. Then C is a binomial random variable with probability of success 4ϵ and number of trials $(1-\alpha)Nm$. Therefore, by the Chernoff bound:

$$\mathbb{P}_{S'}(C \geq \alpha N m) = \mathbb{P}_{S'}(C \geq (1+1)4\epsilon(1-\alpha)Nm) \leq e^{-\frac{\alpha N m}{6}} \leq e^{-1/6} < \frac{17}{20} \tag{A.29}$$

and so:

$$\mathbb{P}_{S'}(C \leq \alpha N m) > \frac{3}{20}. \tag{A.30}$$

Now consider the following policy for the fixed-set adversary $\mathfrak{A}^s : S' \to S$. For any index $i \in [N]$ the adversary replaces $S'_i = \{(x'_{i,1}, f(x'_{i,1})), \ldots, (x'_{i,m}, f(x'_{i,m}))\}$ with a dataset $S_i = \{(x_{i,1}, y_{i,1}) \ldots, (x_{i,m}, y_{i,m})\}$, such that:

$$(x_{i,j}, y_{i,j}) = \begin{cases} (x'_{i,j}, f(x'_{i,j})), & \text{if } i \in G = [1, 2, \ldots, k] \\ (x_2, -f(x_2)), & \text{if } i \in [k+1, \ldots, N] \text{ and } (i-k-1)m + j \leq C \\ (x_1, f(x_1)), & \text{otherwise} \end{cases} \tag{A.31}$$

Then the adversary returns $S = (S_1, \ldots, S_N)$. That is, the adversary keeps the datasets in G untouched, and fills the datasets in $[N] \backslash G$ with as many x_2's as there are in G, but wrongly labelled.

Crucially, whenever $C \leq \alpha N m$, the union of the data in all N sets will look the same no matter if the original labelling function was h_1 or h_2. In particular, $\mathcal{L}(\mathfrak{A}^s(S'))$ will be identical in both cases.

Finally, we argue that under the event $C \leq \alpha N m$ and the chosen adversary, the learner would incur high loss and show that this implies the result in (A.27). Let $\mathcal{S}$ be the set of all datasets in $(\mathcal{X} \times \mathcal{Y})^{N \times m}$, such that $C \leq \alpha N m$ holds. We just showed that $\mathbb{P}_{S'}(S' \in \mathcal{S}) > \frac{3}{20}$ and that whenever $S' \in \mathcal{S}$, $\mathcal{L}(\mathfrak{A}^s(S'))$ is independent of whether the original labelling function was h_1 or h_2.

Consider a fixed set $S' \in \mathcal{S}$ and let $S = \mathfrak{A}^s(S')$ and $h_S = \mathcal{L}(S)$. Denote by $\mathcal{R}(h_S, f) = \mathbb{P}_{\mathcal{D}}(h_S(x) \neq f(x) \cap x \neq x_1)$ and note that $\mathcal{R}(h_S, f) \leq \mathbb{P}_{\mathcal{D}}(h_S(x) \neq f(x)) = \mathcal{R}(\mathcal{L}(\mathfrak{A}^s(S')))$. Notice that:

$$\begin{aligned} \mathcal{R}(h_S, h_1) + \mathcal{R}(h_S, h_2) &= \sum_{i=1,2} \mathbb{1}_{h_S(x_i) \neq h_1(x_i)} \mathbb{1}_{x_i \neq x_1} \mathbb{P}(x_i) + \sum_{i=1,2} \mathbb{1}_{h_S(x_i) \neq h_2(x_i)} \mathbb{1}_{x_i \neq x_1} \mathbb{P}(x_i) \\ &= \mathbb{1}_{h_S(x_2) \neq h_1(x_2)} 4\epsilon + \mathbb{1}_{h_S(x_2) \neq h_2(x_2)} 4\epsilon \\ &= 4\epsilon, \end{aligned}$$

where we used that $h_1(x_2) = 1 = -h_2(x_2)$ and that h_S is independent of the underlying labelling function.

Since the above holds for any $S' \in \mathcal{S}$, it also holds in expectation, conditioned on $S' \in \mathcal{S}$:

$$\mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, h_1) + \mathcal{R}(h_S, h_2) \right) \geq 4\epsilon. \tag{A.32}$$

Therefore, $\mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, h_i) \right) \geq 2\epsilon$ for at least one of $i = 1, 2$. Take f to be h_1, if h_1 satisfies the inequality, and h_2 otherwise. Conditioning on $\{\mathcal{R}(h_S, f) \geq \epsilon\}$ and using $\mathcal{R}(h_S, f) \leq \mathbb{P}_{\mathcal{D}}(x \neq x_1) = 4\epsilon$:

$$\begin{aligned} 2\epsilon \leq \mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \right) &= \mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) | \mathcal{R}(h_S, f) \geq \epsilon \right) \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) \\ &\quad + \mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) | \mathcal{R}(h_S, f) < \epsilon \right) \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) < \epsilon \right) \\ &\leq 4\epsilon \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) + \epsilon \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) < \epsilon \right) \\ &= \epsilon + 3\epsilon \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right). \end{aligned}$$

Hence,

$$\mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) \geq \frac{1}{3\epsilon} \left(2\epsilon - \epsilon \right) = \frac{1}{3} \tag{A.33}$$

Finally,

$$\begin{aligned} \mathbb{P}_{S'} \left(\mathcal{R}(\mathcal{L}(\mathfrak{A}^s(S'))) \geq \epsilon \right) \geq \mathbb{P}_{S'} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) &\geq \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) \mathbb{P}_{S'} \left(S' \in \mathcal{S} \right) \\ &> \frac{1}{3} \frac{3}{20} \\ &= \frac{1}{20}. \end{aligned}$$

□

A.3 Proof of Theorem 6

Theorem 6. *Let $\mathcal{H} \subset \{h : \mathcal{X} \to \mathcal{Y}\}$ be a hypothesis space, let m and N be any integers and let G be a fixed subset of $[N]$ of size $k \in \{1, \ldots, N-1\}$. Let $S' \in (\mathcal{X} \times \mathcal{Y})^{N \times m}$ be drawn i.i.d. from $\mathcal{D}$. Then the following statements hold for any multi-source learner $\mathcal{L}$:*

(a) Suppose that $\mathcal{H}$ is non-trivial. Then there exists a distribution $\mathcal{D}$ on $\mathcal{X}$ with $\min_{h \in \mathcal{H}} \mathcal{R}(h) = 0$, and a fixed-set adversary $\mathfrak{A}$ with index set G, such that:

$$\mathbb{P}_{S'} \left(\mathcal{R}\Big(\mathcal{L}(\mathfrak{A}(S') \Big) > \frac{\alpha}{8m} \right) > \frac{1}{20}. \tag{A.34}$$

(b) Suppose that $\mathcal{H}$ has VC dimension $d \geq 2$. Then there exists a distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$ and a fixed-set adversary $\mathfrak{A}$ with index set G, such that:

$$\mathbb{P}_{S'} \left(\mathcal{R}\Big(\mathcal{L}(\mathfrak{A}(S') \Big) - \min_{h \in \mathcal{H}} \mathcal{R}(h) > \sqrt{\frac{d}{1280Nm}} + \frac{\alpha}{16m} \right) > \frac{1}{64}. \tag{A.35}$$

In both cases, $\alpha = \frac{N-k}{N}$ is the power of the adversary.

To prove part (a), we use a similar technique as in the lower bound results in [BEK02] and in the classic sample complexity lower bound for binary classification, e.g. Theorem 3.20 in [MRT18]. An overview is as follows. Consider a distribution on $\mathcal{X}$ that has support only at two points - the common point x_1 and the rare point x_2. Take $\mathbb{P}(x_2) = \mathcal{O}(\frac{\alpha}{m})$. Then one can show that with constant probability the number of datasets that contain x_2 is at most αN. We show that in this case there exists an algorithm for the strong adversary that will return the same unordered collection of datasets, regardless of the true label of x_2. Thus no learner can guess with probability greater than $1/2$ what the true label of x_2 was.

Part (b) follows from part (a) and the standard lower bound for agnostic binary classification.

Proof. a) As in Theorem 5, we prove that there exists a distribution $\mathcal{D}$ on $\mathcal{X}$ and a labeling function $f \in \mathcal{H}$, such that the resulting joint distribution on $\mathcal{X} \times \mathcal{Y}$, defined by $x \sim \mathcal{D}$ and $y = f(x)$, satisfies the desired property.

Without loss of generality, let $G = [1, 2, \ldots, k]$. Since $\mathcal{H}$ is non-trivial ($d \geq 2$), there exist $h_1, h_2 \in \mathcal{H}$ and $x_1, x_2 \in \mathcal{X}$, such that $h_1(x_1) = h_2(x_1)$, while $h_1(x_2) = 1$, but $h_2(x_2) = -1$. Consider the following distribution on $\mathcal{X}$:

$$\mathbb{P}_{\mathcal{D}}(x_1) = 1 - 4\epsilon \quad \text{and} \quad \mathbb{P}_{\mathcal{D}}(x_2) = 4\epsilon, \tag{A.36}$$

where $\epsilon = \frac{\alpha}{8m}$. Assume that the points are labelled by a function $f \in \mathcal{H}$ (to be chosen later as either h_1 or h_2). Denote the initial uncorrupted collection of datasets by $S' = (S'_1, \ldots, S'_N)$, with $S'_i = \{(x'_{i,1}, f(x'_{i,1})), \ldots, (x'_{i,m}, f(x'_{i,m}))\}$ and $x'_{i,j}$ being i.i.d. samples from $\mathcal{D}$.

First we show that with constant probability the point x_2 is contained in no more than αN sources. Indeed, let C_b be the number of sources that contain x_2 and let C_p be the number of points (out of the Nm in total) that are equal to x_2. Clearly $C_b \leq C_p$. Note that C_p is a binomial random variable with probability of success 4ϵ and number of trials Nm. Therefore, by the Chernoff bound:

$$\mathbb{P}_{S'}(C_p \geq \alpha N) = \mathbb{P}_{S'}(C_p \geq (1+1)4\epsilon Nm) \leq e^{-\frac{\alpha N}{6}} \leq e^{-1/6} < \frac{17}{20} \tag{A.37}$$

and so:

$$\mathbb{P}_{S'}(C_b \leq \alpha N) \geq \mathbb{P}_{S'}(C_p \leq \alpha N) > \frac{3}{20}. \tag{A.38}$$

Now consider the following policy for the adversary $\mathfrak{A}^s : S' \to S$. Whenever $C_b \leq \alpha N$, let $M \subset G$ be the list of indexes $i \in G$, such that S'_i contains x_2. Let $l = |M|$ and note that $l \leq C_b \leq \alpha N$. For any index $i \in [N]$ the adversary replaces $S'_i = \{x'_{i,1}, f(x'_{i,1}), \ldots, (x'_{i,m}, f(x'_{i,m}))\}$ with a dataset $S_i = \{(x_{i,1}, y_{i,1}) \ldots, (x_{i,m}, y_{i,m})\}$, such that:

$$(x_{i,j}, y_{i,j}) = \begin{cases} (x'_{i,j}, f(x'_{i,j})), & \text{if } i \in G = [1, 2, \ldots, k] \\ (x_1, f(x_1)), & \text{if } i \in [k+1, \ldots, k+l] \text{ and } x'_{M[i-k],j} = x_1 \\ (x_2, -f(x_2)), & \text{if } i \in [k+1, \ldots, k+l] \text{ and } x'_{M[i-k],j} = x_2 \\ (x_1, f(x_1)), & \text{if } i \in [k+l+1, \ldots, N] \end{cases} \tag{A.39}$$

Then the adversary returns $S = (S_1, \ldots, S_N)$. That is, the adversary keeps the datasets in G untouched, copies all of the datasets in M into its own data, flipping the labels of the x_2's, and, in case there are additional sources at its disposal, it fills them with (correctly labelled) x_1's only.

Crucially, the resulting (unordered) collection is the same no matter if the original labelling function was h_1 or h_2. In particular, $\mathcal{L}(S)$ will be the same in both cases.

In the case when $C_b > \alpha N$, the adversary leaves the data unchanged, i.e. $S = S'$.

Finally, we argue that under the event $C_b \leq \alpha N$ and the chosen adversary, the learner would incur high loss and show that this implies the result in (A.34). Let $\mathcal{S}$ be the set of all datasets in $(\mathcal{X} \times \mathcal{Y})^{N \times m}$, such that $C_b \leq \alpha N$ holds. We just showed that $\mathbb{P}_{S'}(S' \in \mathcal{S}) > \frac{3}{20}$ and that whenever $S' \in \mathcal{S}$, $\mathcal{L}(\mathfrak{A}^s(S'))$ is independent of whether the original labelling function was h_1 or h_2.

Now the proof proceeds just as in Theorem 5. Consider a fixed set $S' \in \mathcal{S}$ and let $S = \mathfrak{A}^s(S')$ and $h_S = \mathcal{L}(S)$. Denote by $\mathcal{R}(h_S, f) = \mathbb{P}_{\mathcal{D}}(h_S(x) \neq f(x) \cap x \neq x_1)$ and note that $\mathcal{R}(h_S, f) \leq \mathbb{P}_{\mathcal{D}}(h_S(x) \neq f(x)) = \mathcal{R}(\mathcal{L}(\mathfrak{A}^s(S')))$. Notice that:

$$\begin{aligned}
\mathcal{R}(h_S, h_1) + \mathcal{R}(h_S, h_2) &= \sum_{i=1,2} \mathbb{1}_{h_S(x_i) \neq h_1(x_i)} \mathbb{1}_{x_i \neq x_1} \mathbb{P}(x_i) + \sum_{i=1,2} \mathbb{1}_{h_S(x_i) \neq h_2(x_i)} \mathbb{1}_{x_i \neq x_1} \mathbb{P}(x_i) \\
&= \mathbb{1}_{h_S(x_2) \neq h_1(x_2)} 4\epsilon + \mathbb{1}_{h_S(x_2) \neq h_2(x_2)} 4\epsilon \\
&= 4\epsilon,
\end{aligned}$$

where we used that $h_1(x_2) = 1 = -h_2(x_2)$ and that h_S is independent of the underlying labelling function.

Since the above holds for any $S' \in \mathcal{S}$, it also holds in expectation, conditioned on $S' \in \mathcal{S}$:

$$\mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, h_1) + \mathcal{R}(h_S, h_2) \right) \geq 4\epsilon. \tag{A.40}$$

Therefore, $\mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, h_i) \right) \geq 2\epsilon$ for at least one of $i = 1, 2$. Take f to be h_1, if h_1 satisfies the inequality, and h_2 otherwise. Conditioning on $\{\mathcal{R}(h_S, f) \geq \epsilon\}$ and using $\mathcal{R}(h_S, f) \leq \mathbb{P}_{\mathcal{D}}(x \neq x_1) = 4\epsilon$:

$$\begin{aligned}
2\epsilon \leq \mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \right) &= \mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) | \mathcal{R}(h_S, f) \geq \epsilon \right) \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) \\
&\quad + \mathbb{E}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) | \mathcal{R}(h_S, f) < \epsilon \right) \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) < \epsilon \right) \\
&\leq 4\epsilon \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) + \epsilon \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) < \epsilon \right) \\
&= \epsilon + 3\epsilon \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right).
\end{aligned}$$

Hence,

$$\mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) \geq \frac{1}{3\epsilon} \left(2\epsilon - \epsilon \right) = \frac{1}{3} \tag{A.41}$$

Finally,

$$\begin{aligned}
\mathbb{P}_{S'} \left(\mathcal{R}(\mathcal{L}(\mathfrak{A}^s(S'))) \geq \epsilon \right) &\geq \mathbb{P}_{S'} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) \\
&\geq \mathbb{P}_{S' \in \mathcal{S}} \left(\mathcal{R}(h_S, f) \geq \epsilon \right) \mathbb{P}_{S'} \left(S' \in \mathcal{S} \right) \\
&> \frac{1}{3} \frac{3}{20} \\
&= \frac{1}{20}.
\end{aligned}$$

b) First we argue that there exists a distribution $\mathcal{D}_1$ on $\mathcal{X} \times \mathcal{Y}$ and a fixed-set adversary $\mathfrak{A}_1^s$, such that:

$$\mathbb{P}_{S' \sim \mathcal{D}_1} \left(\mathcal{R}(\mathcal{L}(\mathfrak{A}_1^s(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) > \sqrt{\frac{d}{320Nm}} \right) > \frac{1}{64}. \tag{A.42}$$

This follows directly from the classic lower bound for binary classification in the unrealizable case. Indeed, applying Theorem 3.23 in [MRT18] and setting the adversary to be the identity mapping gives the result.

Now, since any hypothesis space with VC dimension $d \geq 2$ is non-trivial, we also know from a) that there exists an adversary $\mathfrak{A}_2^s$ and a distribution $\mathcal{D}_2$ on $\mathcal{X} \times \mathcal{Y}$, such that:

$$\mathbb{P}_{S' \sim \mathcal{D}_2} \left(\mathcal{R}(\mathcal{L}(\mathfrak{A}_2^s(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) > \frac{\alpha}{8m} \right) > \frac{1}{20}. \tag{A.43}$$

Fix any set of values for N, m, d, k. Then at least one of the pairs $(\mathfrak{A}_1^s, \mathcal{D}_1)$ and $(\mathfrak{A}_2^s, \mathcal{D}_2)$ satisfies:

$$\begin{aligned}
&\mathbb{P}_{S'} \left(\mathcal{R}(\mathcal{L}(\mathfrak{A}^s(S'))) - \min_{h \in \mathcal{H}} \mathcal{R}(h) > \sqrt{\frac{d}{1280Nm}} + \frac{\alpha}{16m} \right) \\
&\geq \mathbb{P}_{S'} \left(\mathcal{R}(\mathcal{L}(\mathfrak{A}^s(S'))) > 2 \max\{\sqrt{\frac{d}{1280Nm}}, \frac{\alpha}{16m}\} \right) \\
&= \mathbb{P}_{S'} \left(\mathcal{R}(\mathcal{L}(\mathfrak{A}^s(S'))) > \max\{\sqrt{\frac{d}{320Nm}}, \frac{\alpha}{8m}\} \right) \\
&> \frac{1}{64}.
\end{aligned}$$

□

APPENDIX B

Proofs from Chapter 4

Here we present a proof of Theorem 7. To this end, we first show a concentration result about the α-weighted empirical risk $\mathcal{R}_\alpha(h)$. Then we use this result, together with a standard discrepancy argument, to prove the statement of the theorem.

First we bound $|\hat{\mathcal{R}}_\alpha(h) - \mathcal{R}_\alpha(h)|$ with high probability and uniformly over $\mathcal{H}$. We adapt the classical proofs of generalization bounds in terms of the Rademacher complexity of a hypothesis class, e.g. [BBL04].

Lemma 1. *Given the setup and assumptions described above, for any $\delta > 0$ with probability at least $1 - \delta$ over the data, for any function $h \in \mathcal{H}$:*

$$|\mathcal{R}_\alpha(h) - \hat{\mathcal{R}}_\alpha(h)| \leq 2\sum_{i=1}^{N} \alpha_i \mathfrak{R}_i(\mathcal{H}) + 3\sqrt{\frac{\log\left(\frac{4}{\delta}\right) M^2}{2}}\sqrt{\sum_{i=1}^{N} \frac{\alpha_i^2}{m_i}}, \tag{B.1}$$

where for each $i = 1, 2, \ldots, N$:

$$\mathfrak{R}_i(\mathcal{H}) = \mathbb{E}_\sigma\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m_i}\sum_{j=1}^{m_i} \sigma_{i,j}\ell(f(x_{i,j}), y_{i,j})\right)\right), \tag{B.2}$$

and where $\sigma_{i,j}$ are independent Rademacher random variables.

Proof. Write:

$$\mathcal{R}_\alpha(h) \leq \hat{\mathcal{R}}_\alpha(h) + \sup_{f\in\mathcal{H}}\left(\mathcal{R}_\alpha(f) - \hat{\mathcal{R}}_\alpha(f)\right) \tag{B.3}$$

To link the second term to its expectation, we prove the following:

Lemma 2. *Define the function $\phi : (\mathcal{X}\times\mathcal{Y})^m \to \mathbb{R}$ by:*

$$\phi(\{x_{1,1}, y_{1,1}\}, \ldots, \{x_{N,m_N}, y_{N,m_N}\}) = \sup_{f\in\mathcal{H}}\left(\mathcal{R}_\alpha(f) - \hat{\mathcal{R}}_\alpha(f)\right).$$

Denote for brevity $z_{i,j} = \{x_{i,j}, y_{i,j}\}$. Then, for any $i \in \{1, 2, \ldots, N\}, j \in \{1, 2, \ldots, m_i\}$:

$$\sup_{z_{1,1},\ldots,z_{N,m_N},z'_{i,j}} |\phi(z_{1,1}, \ldots, z_{i,j}, \ldots, z_{N,m_N}) - \phi\left(z_{1,1}, \ldots, z'_{i,j}, \ldots, z_{N,m_N}\right)| \leq \frac{\alpha_i}{m_i} M \tag{B.4}$$

Proof. Fix any i, j and any $z_{1,1}, \ldots, z_{N,m_N}, z_{i,j}^{'}$. Denote the α-weighted empirical average of the loss with respect to the sample $z_{1,1}, \ldots, z_{i,j}^{'}, \ldots, z_{N,m_N}$ by $\mathcal{R}_{\alpha}^{'}$. Then we have that:

$$\begin{aligned}
|\phi\left(\ldots, z_{i,j}, \ldots\right) - \phi(\ldots, z_{i,j}^{'}, \ldots)| &= |\sup_{f\in\mathcal{H}}\left(\mathcal{R}_{\alpha}\left(f\right) - \hat{\mathcal{R}}_{\alpha}\left(f\right)\right) - \sup_{f\in\mathcal{H}}(\mathcal{R}_{\alpha}\left(f\right) - \hat{\mathcal{R}}_{\alpha}^{'}\left(f\right))| \\
&\leq |\sup_{f\in\mathcal{H}}(\hat{\mathcal{R}}_{\alpha}^{'}\left(f\right) - \hat{\mathcal{R}}_{\alpha}\left(f\right))| \\
&= \frac{\alpha_i}{m_i}\left|\sup_{f\in\mathcal{H}}\left(\ell\left(f(x_{i,j}^{'}), y_{i,j}^{'}\right) - \ell\left(f(x_{i,j}), y_{i,j}\right)\right)\right| \\
&\leq \frac{\alpha_i}{m_i} M
\end{aligned}$$

Note: the inequality we used above holds for bounded functions inside the supremum. □

Let S denote a random sample of size m drawn from a distribution as the one generating out data (i.e. m_i samples from $\mathcal{D}_i$ for each i). Now, using Lemma 2, McDiarmid's inequality gives:

$$\mathbb{P}\left(\phi(S) - \mathbb{E}(\phi(S)) \geq t\right) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{N}\sum_{j=1}^{m_i}\frac{\alpha_i^2}{m_i^2}M^2}\right) = \exp\left(-\frac{2t^2}{M^2\sum_{i=1}^{N}\frac{\alpha_i^2}{m_i}}\right)$$

For any $\delta > 0$, setting the right-hand side above to be $\delta/4$ and using (B.3), we obtain that with probability at least $1 - \delta/4$:

$$\mathcal{R}_{\alpha}\left(h\right) \leq \hat{\mathcal{R}}_{\alpha}\left(h\right) + \mathbb{E}_S\left(\sup_{f\in\mathcal{H}}\left(\mathcal{R}_{\alpha}(f) - \hat{\mathcal{R}}_{\alpha}(f)\right)\right) + \sqrt{\frac{\log\left(\frac{4}{\delta}\right)M^2}{2}}\sqrt{\sum_{i=1}^{N}\frac{\alpha_i^2}{m_i}} \tag{B.5}$$

To deal with the expected loss inside the second term, introduce a ghost sample (denoted by S'), drawn from the same distributions as our original sample (denoted by S). Denoting the weighted empirical loss with respect to the ghost sample by $\mathcal{R}_{\alpha}^{'}$, $\beta_i = m_i/m$ for all i, and using the convexity of the supremum, we obtain:

$$\begin{aligned}
\mathbb{E}_S\left(\sup_{f\in\mathcal{H}}\left(\mathcal{R}_{\alpha}(f) - \hat{\mathcal{R}}_{\alpha}(f)\right)\right) &= \mathbb{E}_S\left(\sup_{f\in\mathcal{H}}\left(\mathbb{E}_{S'}\left(\hat{\mathcal{R}}_{\alpha}^{'}(f)\right) - \hat{\mathcal{R}}_{\alpha}(f)\right)\right) \\
&\leq \mathbb{E}_{S,S'}\left(\sup_{f\in\mathcal{H}}\left(\hat{\mathcal{R}}_{\alpha}^{'}(f) - \hat{\mathcal{R}}_{\alpha}(f)\right)\right) \\
&= \mathbb{E}_{S,S'}\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m}\sum_{i=1}^{N}\sum_{j=1}^{m_i}\frac{\alpha_i}{\beta_i}\left(\ell(f(x_{i,j}^{'}), y_{i,j}^{'}) - \ell(f(x_{i,j}), y_{i,j})\right)\right)\right)
\end{aligned}$$

Introducing m independent Rademacher random variables and noting that $\ell(f(x^{'}), y^{'}) - \ell(f(x), y)$ and $\sigma\left(\ell(f(x^{'}), y^{'}) - \ell(f(x), y)\right)$ have the same distribution, as long as (x, y) and

$(x^{'}, y^{'})$ have the same distribution:

$$\begin{aligned}\mathbb{E}_S\left(\sup_{f\in\mathcal{H}}\left(\mathcal{R}_\alpha(f)-\hat{\mathcal{R}}_\alpha(f)\right)\right) &\leq \mathbb{E}_{S,S',\sigma}\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m}\sum_{i=1}^{N}\sum_{j=1}^{m_i}\frac{\alpha_i}{\beta_i}\sigma_{i,j}\left(\ell(f(x^{'}_{i,j}), y^{'}_{i,j})\right.\right.\right.\\ &\left.\left.\left.-\ell(f(x_{i,j}), y_{i,j})\right)\right)\right)\\ &\leq \mathbb{E}_{S',\sigma}\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m}\sum_{i=1}^{N}\sum_{j=1}^{m_i}\frac{\alpha_i}{\beta_i}\sigma_{i,j}\ell(f(x^{'}_{i,j}), y^{'}_{i,j})\right)\right)\\ &+\mathbb{E}_{S,\sigma}\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m}\sum_{i=1}^{N}\sum_{j=1}^{m_i}\frac{\alpha_i}{\beta_i}\left(-\sigma_{i,j}\right)\ell(f(x_{i,j}), y_{i,j})\right)\right)\\ &= 2\mathbb{E}_{S,\sigma}\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m}\sum_{i=1}^{N}\sum_{j=1}^{m_i}\frac{\alpha_i}{\beta_i}\sigma_{i,j}\ell(f(x_{i,j}), y_{i,j})\right)\right)\end{aligned}$$

We can now link the last term to the empirical analog of the Rademacher complexity, by using the McDiarmid Inequality (with an observation similar to Lemma 1). Putting this together, we obtain that for any $\delta > 0$ with probability at least $1 - \delta/2$:

$$\mathcal{R}_\alpha\left(h\right) \leq \hat{\mathcal{R}}_\alpha\left(h\right) + 2\mathbb{E}_\sigma\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m}\sum_{i=1}^{N}\sum_{j=1}^{m_i}\frac{\alpha_i}{\beta_i}\sigma_{i,j}\ell(f(x_{i,j}), y_{i,j})\right)\right) + 3\sqrt{\frac{\log\left(\frac{4}{\delta}\right)M^2}{2}}\sqrt{\sum_{i=1}^{N}\frac{\alpha_i^2}{m_i}} \tag{B.6}$$

Finally, note that:

$$\begin{aligned}\mathbb{E}_\sigma\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m}\sum_{i=1}^{N}\sum_{j=1}^{m_i}\frac{\alpha_i}{\beta_i}\sigma_{i,j}\ell(f(x_{i,j}), y_{i,j})\right)\right) &\leq \mathbb{E}_\sigma\left(\sum_{i=1}^{N}\alpha_i\sup_{f\in\mathcal{H}}\left(\frac{1}{m_i}\sum_{j=1}^{m_i}\sigma_{i,j}\ell(f(x_{i,j}), y_{i,j})\right)\right)\\ &= \sum_{i=1}^{N}\alpha_i\mathbb{E}_\sigma\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m_i}\sum_{j=1}^{m_i}\sigma_{i,j}\ell(f(x_{i,j}), y_{i,j})\right)\right)\\ &= \sum_{i=1}^{N}\alpha_i\mathfrak{R}_i\left(\mathcal{H}\right)\end{aligned}$$

Bounding $\hat{\mathcal{R}}_\alpha(h) - \mathcal{R}_\alpha(h)$ with the same quantity and with probability at least $1 - \delta/2$ follows by a similar argument. The result then follows by applying the union bound. □

Now we show:

Theorem 7. *Given the setup above, let $\hat{h}_\alpha = argmin_{h\in\mathcal{H}}\hat{\mathcal{R}}_\alpha(h)$ and $h_T^* = argmin_{h\in\mathcal{H}}\mathcal{R}_T(h)$. For any $\delta > 0$, with probability at least $1 - \delta$ over the data:*

$$\mathcal{R}_T(\hat{h}_\alpha) \leq \mathcal{R}_T(h_T^*) + 4\sum_{i=1}^{N}\alpha_i\mathfrak{R}_i\left(\mathcal{H}\right) + 2\sum_{i=1}^{N}\alpha_i d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right) + 6\sqrt{\frac{\log\left(\frac{4}{\delta}\right)M^2}{2}}\sqrt{\sum_{i=1}^{N}\frac{\alpha_i^2}{m_i}}, \tag{4.4}$$

where, for each source $i = 1, \ldots, N$,

$$\mathfrak{R}_i\left(\mathcal{H}\right) = \mathbb{E}_\sigma\left(\sup_{f\in\mathcal{H}}\left(\frac{1}{m_i}\sum_{j=1}^{m_i}\sigma_{i,j}\ell(f(x_{i,j}), y_{i,j})\right)\right)$$

and $\sigma_{i,j}$ are independent Rademacher random variables.

Proof. For any $h \in \mathcal{H}$:

$$|\mathcal{R}_\alpha(h) - \mathcal{R}_T(h)| = |\sum_{i=1}^N \alpha_i \mathcal{R}_i(h) - \mathcal{R}_T(h)| \leq \sum_{i=1}^N \alpha_i |\mathcal{R}_i(h) - \mathcal{R}_T(h)| \leq \sum_{i=1}^N \alpha_i d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right).$$

Now applying this bound twice and using Lemma 1, we get that with probability at least $1-\delta$:

$$\begin{aligned}
\mathcal{R}_T(\hat{h}_\alpha) &\leq \mathcal{R}_\alpha(\hat{h}_\alpha) + \sum_{i=1}^N \alpha_i d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right) \\
&\leq \hat{\mathcal{R}}_\alpha(\hat{h}_\alpha) + 2\sum_{i=1}^N \alpha_i \mathfrak{R}_i\left(\mathcal{H}\right) + 3\sqrt{\frac{\log\left(\frac{4}{\delta}\right) M^2}{2}}\sqrt{\sum_{i=1}^N \frac{\alpha_i^2}{m_i}} + \sum_{i=1}^N \alpha_i d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right) \\
&\leq \hat{\mathcal{R}}_\alpha(h_T^*) + 2\sum_{i=1}^N \alpha_i \mathfrak{R}_i\left(\mathcal{H}\right) + 3\sqrt{\frac{\log\left(\frac{4}{\delta}\right) M^2}{2}}\sqrt{\sum_{i=1}^N \frac{\alpha_i^2}{m_i}} + \sum_{i=1}^N \alpha_i d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right) \\
&\leq \mathcal{R}_\alpha(h_T^*) + 4\sum_{i=1}^N \alpha_i \mathfrak{R}_i\left(\mathcal{H}\right) + 6\sqrt{\frac{\log\left(\frac{4}{\delta}\right) M^2}{2}}\sqrt{\sum_{i=1}^N \frac{\alpha_i^2}{m_i}} + \sum_{i=1}^N \alpha_i d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right) \\
&\leq \mathcal{R}_T(h_T^*) + 4\sum_{i=1}^N \alpha_i \mathfrak{R}_i\left(\mathcal{H}\right) + 6\sqrt{\frac{\log\left(\frac{4}{\delta}\right) M^2}{2}}\sqrt{\sum_{i=1}^N \frac{\alpha_i^2}{m_i}} + 2\sum_{i=1}^N \alpha_i d_{\mathcal{H}}\left(\mathcal{D}_i, \mathcal{D}_T\right)
\end{aligned}$$

□

APPENDIX C

Proofs from Chapter 5

Here we present the proofs of all results from Chapter 5. The chapter is structured as follows.

- **Appendix C.1** contains the proofs of all lower bounds results. Section C.1.1 focuses on the Pareto lower bounds. Section C.1.2 contains the proofs for the lower bounds on fairness, given that accuracy is kept optimal.
- **Appendix C.2** contains the complete proofs of our upper bound results. In particular, Section C.2.1 explains the notation and introduces the classic concentration tools that we will use. In Section C.2.2 a number of concentration results under corrupted data for the demographic parity and equal opportunity fairness notions are shown. Finally, Section C.2.3 gives the formal proofs of all upper bound results, building on the concentration inequalities from the previous section.

C.1 Lower bounds proofs

In the proofs of our hardness results we use a technique from [KL93] called the *method of induced distributions*. The idea is to construct two distributions that are sufficiently different, so that different classifiers perform well on each, yet can be made indistinguishable after the modifications of the adversary. Then no fixed learner with access only to the corrupted data can be "correct" with high probability on both distributions and so any learner will incur excessively high loss and/or exhibit excessively high unfairness on at least one of the two distributions, regardless of the amount of available data.

The proofs of the four results are structured in a similar way, but use different constructions of the underlying learning problem, tailored to the fairness measure and the type of bound we want to show.

C.1.1 Pareto lower bounds proofs

Theorem 8. *Let $0 \leq \alpha < 0.5, 0 < P_0 \leq 0.5$. For any input set $\mathcal{X}$ with at least four distinct points, there exists a finite hypothesis space $\mathcal{H}$, such that for any learning algorithm $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$, there exists a distribution $\mathbb{P}$ for which $\mathbb{P}(A = 0) = P_0$, a malicious adversary $\mathfrak{A}$ of power α and a hypothesis $h^* \in \mathcal{H}$, such that with probability at least*

0.5

$$L(\mathcal{L}(S^p), \mathbb{P}) - L(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{1-\alpha}, 2P_0(1-P_0)\right\}$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}) - \Gamma^{DP}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{2P_0(1-P_0)(1-\alpha)}, 1\right\} \geq \min\left\{\frac{\alpha}{2P_0}, 1\right\}.$$

Proof. Let $\eta = \frac{\alpha}{1-\alpha}$, so that $\alpha = \frac{\eta}{1+\eta}$.

Case 1 Assume that $\eta = \frac{\alpha}{1-\alpha} \leq 2P_0(1-P_0)$. Take four distinct points $\{x_1, x_2, x_3, x_4\} \in \mathcal{X}$. We consider two distributions $\mathbb{P}_0$ and $\mathbb{P}_1$, where each $\mathbb{P}_i$ is defined as

$$\mathbb{P}_i(x, a, y) = \begin{cases} 1 - P_0 - \eta/2 & \text{if } x = x_1, a = 1, y = 1 \\ P_0 - \eta/2 & \text{if } x = x_2, a = 0, y = 0 \\ \eta/2 & \text{if } x = x_3, a = i, y = \neg i \\ \eta/2 & \text{if } x = x_4, a = \neg i, y = i \\ 0 & \text{otherwise} \end{cases}$$

Note that these are valid distributions, since $\eta \leq 2P_0(1-P_0) \leq 2P_0 \leq 2(1-P_0)$ by assumption and also that $P_0 = \mathbb{P}_i(A = 0)$ for both $i \in \{0, 1\}$. Consider the hypothesis space $\mathcal{H} = \{h_0, h_1\}$, with

$$h_0(x_1) = 1 \quad h_0(x_2) = 0 \quad h_0(x_3) = 1 \quad h_0(x_4) = 0$$

and

$$h_1(x_1) = 1 \quad h_1(x_2) = 0 \quad h_1(x_3) = 0 \quad h_1(x_4) = 1.$$

Note that $L(h_i, \mathbb{P}_i) = 0$ for both $i = 0, 1$. Moreover,

$$\begin{aligned} \Gamma^{DP}(h_0, \mathbb{P}_0) &= \left|\mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_0(X) = 1|A = 0) - \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_0(X) = 1|A = 1)\right| \\ &= \left|\frac{\eta/2}{P_0 - \eta/2 + \eta/2} - \frac{1 - P_0 - \eta/2}{1 - P_0 - \eta/2 + \eta/2}\right| \\ &= \left|\frac{\eta}{2P_0} - \frac{2 - 2P_0 - \eta}{2(1-P_0)}\right| \\ &= \left|\frac{\eta}{2P_0(1-P_0)} - 1\right| \\ &= 1 - \frac{\eta}{2P_0(1-P_0)}, \end{aligned}$$

since $\eta \leq 2P_0(1-P_0)$ by assumption. Furthermore,

$$\begin{aligned} \Gamma^{DP}(h_1, \mathbb{P}_0) &= \left|\mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_1(X) = 1|A = 0) - \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_1(X) = 1|A = 1)\right| \\ &= |0 - 1| \\ &= 1 \end{aligned}$$

Therefore, $\Gamma^{DP}(h_1, \mathbb{P}_0) - \Gamma^{DP}(h_0, \mathbb{P}_0) = \frac{\eta}{2P_0(1-P_0)}$. Similarly,

$$\Gamma^{DP}(h_1, \mathbb{P}_1) = \left|\mathbb{P}_{(X,A,Y)\sim\mathbb{P}_1}(h_1(X) = 1|A = 0) - \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_1}(h_1(X) = 1|A = 1)\right|$$

$$= \left| \frac{\eta/2}{P_0 - \eta/2 + \eta/2} - \frac{1 - P_0 - \eta/2}{1 - P_0 - \eta/2 + \eta/2} \right|$$
$$= 1 - \frac{\eta}{2P_0(1-P_0)}$$

and

$$\begin{aligned}\Gamma^{DP}(h_0, \mathbb{P}_1) &= \left| \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_1}(h_0(X) = 1 | A = 0) - \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_1}(h_0(X) = 1 | A = 1) \right| \\ &= |0 - 1| \\ &= 1,\end{aligned}$$

so that $\Gamma^{DP}(h_0, \mathbb{P}_1) - \Gamma^{DP}(h_1, \mathbb{P}_1) = \frac{\eta}{2P_0(1-P_0)}$.

Consider a (randomized) malicious adversary $\mathfrak{A}_i$ of power α, that given a clean distribution $\mathbb{P}_i$, changes every marked point to $(x_3, \neg i, i)$ with probability 0.5 and to $(x_4, i, \neg i)$ otherwise. Under a distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, the probability of seeing a point $(x_3, i, \neg i)$ is $\frac{\eta}{2}(1-\alpha) = \frac{\eta}{2}\frac{1}{1+\eta} = \alpha/2$, which is equal to the probability of seeing a point $(x_3, \neg i, i)$. Therefore, denoting the probability distribution of the corrupted dataset, under a clean distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, by $\mathbb{P}'_i$ (as a shorthand for $\mathbb{P}_i^{\mathfrak{A}_i}$), we have

$$\mathbb{P}'_i(x, a, y) = \begin{cases} (1-\alpha)(1 - P_0 - \eta/2) & \text{if } x = x_1, a = 1, y = 1 \\ (1-\alpha)(P_0 - \eta/2) & \text{if } x = x_2, a = 0, y = 0 \\ \alpha/2 & \text{if } x = x_3, a = i, y = \neg i \\ \alpha/2 & \text{if } x = x_3, a = \neg i, y = i \\ \alpha/2 & \text{if } x = x_4, a = \neg i, y = i \\ \alpha/2 & \text{if } x = x_4, a = i, y = \neg i \\ 0 & \text{otherwise} \end{cases}$$

In particular, $\mathbb{P}'_0 = \mathbb{P}'_1$, so the two initial distributions $\mathbb{P}_0$ and $\mathbb{P}_1$ become indistinguishable under the adversarial manipulation.

Fix an arbitrary learner $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \{h_0, h_1\}$. Note that, if the clean distribution is $\mathbb{P}_0$, the events (in the probability space defined by the sampling of the poisoned train data)

$$\begin{aligned}\{L(\mathcal{L}(S^p), \mathbb{P}_0) - L(h_0, \mathbb{P}_0) \geq \eta\} &= \{\mathcal{L}(S^p) = h_1\} \\ &= \left\{ \Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_0) - \Gamma^{DP}(h_0, \mathbb{P}_0) \geq \frac{\eta}{2P_0(1-P_0)} \right\}\end{aligned}$$

are all the same. Similarly, if the clean distribution is $\mathbb{P}_1$

$$\begin{aligned}\{L(\mathcal{L}(S^p), \mathbb{P}_1) - L(h_1, \mathbb{P}_1) \geq \eta\} &= \{\mathcal{L}(S^p) = h_0\} \\ &= \left\{ \Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_1) - \Gamma^{DP}(h_1, \mathbb{P}_1) \geq \frac{\eta}{2P_0(1-P_0)} \right\}.\end{aligned}$$

Therefore, depending on whether we choose $\mathbb{P}_0$ or $\mathbb{P}_1$ as a clean distribution, we have

$$\begin{aligned}&\mathbb{P}_{S^p\sim\mathbb{P}'_0}\left((L(\mathcal{L}(S^p), \mathbb{P}_0) - L(h_0, \mathbb{P}_0) \geq \eta) \wedge \left(\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_0) - \Gamma^{DP}(h_0, \mathbb{P}_0) \geq \frac{\eta}{2P_0(1-P_0)} \right) \right) \\ &\quad = \mathbb{P}_{S^p\sim\mathbb{P}'_0}\left(\mathcal{L}(S^p) = h_1\right)\end{aligned}$$

and

$$\mathbb{P}_{S^p\sim\mathbb{P}_1'}\left((L(\mathcal{L}(S^p),\mathbb{P}_1)-L(h_1,\mathbb{P}_1)\geq\eta)\wedge\left(\Gamma^{DP}(\mathcal{L}(S^p),\mathbb{P}_1)-\Gamma^{DP}(h_1,\mathbb{P}_1)\geq\frac{\eta}{2P_0(1-P_0)}\right)\right)$$
$$=\mathbb{P}_{S^p\sim\mathbb{P}_1'}\left(\mathcal{L}(S^p)=h_0\right)$$

Finally, note that $\mathbb{P}_0'=\mathbb{P}_1'$, so that either $\mathbb{P}_{S^p\sim\mathbb{P}_0'}\left(\mathcal{L}(S^p)=h_1\right)\geq 1/2$ or $\mathbb{P}_{S^p\sim\mathbb{P}_1'}\left(\mathcal{L}(S^p)=h_0\right)\geq 1/2$. Therefore, for at least one of $i=0,1$, both

$$L(\mathcal{L}(S^p),\mathbb{P}_i)-L(h_i,\mathbb{P}_i)\geq\eta=\frac{\alpha}{1-\alpha}$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p),\mathbb{P}_i)-\Gamma^{DP}(h_i,\mathbb{P}_i)\geq\frac{\eta}{2P_0(1-P_0)}=\frac{\alpha}{2P_0(1-P_0)(1-\alpha)}$$

both hold with probability at least $1/2$ when the choice of distribution and adversary is $\mathbb{P}_i$ and $\mathfrak{A}_i$ respectively. This concludes the proof in the first case.

Case 2 Now suppose that $\eta=\frac{\alpha}{1-\alpha}>2P_0(1-P_0)$. Let $\alpha_1\in(0,0.5)$ be such that $\frac{\alpha_1}{1-\alpha_1}=2P_0(1-P_0)$. Note that since $f(x)=\frac{x}{1-x}$ is monotonically increasing in (0,1), α_1 is unique and $\alpha_1<\alpha$.

Now repeat the same construction as in Case 1, but with $\eta_1=\frac{\alpha_1}{1-\alpha_1}=2P_0(1-P_0)$. For every marked point, the adversary does the same as in Case 1 with probability α_1/α and does not change the point otherwise. Then the same argument as in Case 1 shows that for one $i\in\{0,1\}$, both

$$L(\mathcal{L}(S^p),\mathbb{P}_i)-L(h_i,\mathbb{P}_i)\geq\eta_1=\frac{\alpha_1}{1-\alpha_1}=2P_0(1-P_0)$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p),\mathbb{P}_i)-\Gamma^{DP}(h_i,\mathbb{P}_i)\geq\frac{\eta_1}{2P_0(1-P_0)}=1$$

both hold with probability at least $1/2$. This concludes the proof of Theorem 8. □

Theorem 9. *Let $0\leq\alpha<0.5, P_{10}\leq P_{11}<1$ be such that $P_{10}+P_{11}<1$. For any input set $\mathcal{X}$ with at least five distinct points, there exists a finite hypothesis space $\mathcal{H}$, such that for any learning algorithm $\mathcal{L}:\cup_{n\in\mathbb{N}}(\mathcal{X}\times\mathcal{A}\times\mathcal{Y})^n\to\mathcal{H}$, there exists a distribution $\mathbb{P}$ for which $\mathbb{P}(A=a,Y=1)=P_{1a}$ for $a\in\{0,1\}$, a malicious adversary $\mathfrak{A}$ of power α and a hypothesis $h^*\in\mathcal{H}$, such that with probability at least 0.5*

$$L(\mathcal{L}(S^p),\mathbb{P})-L(h^*,\mathbb{P})>\min\left\{\frac{\alpha}{1-\alpha},2P_{10},2(1-P_{10}-P_{11})\right\}$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p),\mathbb{P})-\Gamma^{EOp}(h^*,\mathbb{P})\geq\min\left\{\frac{\alpha}{2(1-\alpha)P_{10}},1,\frac{1-P_{10}-P_{11}}{P_{10}}\right\}.$$

Proof. Let $\eta=\frac{\alpha}{1-\alpha}$, so that $\alpha=\frac{\eta}{1+\eta}$.

Case 1 Assume that $\eta = \frac{\alpha}{1-\alpha} \leq 2\min\{P_{10}, 1 - P_{10} - P_{11}\}$. Take five distinct points $\{x_1, x_2, x_3, x_4, x_5\} \in \mathcal{X}$. We consider two distributions $\mathbb{P}_0$ and $\mathbb{P}_1$, where each $\mathbb{P}_i$ is defined as

$$\mathbb{P}_i(x, a, y) = \begin{cases} P_{11} & \text{if } x = x_1, a = 1, y = 1 \\ P_{10} - \eta/2 & \text{if } x = x_2, a = 0, y = 1 \\ \eta/2 & \text{if } x = x_3, a = i, y = \neg i \\ \eta/2 & \text{if } x = x_4, a = \neg i, y = i \\ 1 - P_{10} - P_{11} - \eta/2 & \text{if } x = x_5, a = 0, y = 0 \\ 0 & \text{otherwise} \end{cases}$$

Note that these are valid distributions, since $\eta \leq 2P_{10}, \eta \leq 2(1 - P_{10} - P_{11})$ by assumption, and that $P_{1a} = \mathbb{P}_i(A = a, Y = 1)$ for both $a \in \{0, 1\}, i \in \{0, 1\}$. Consider the hypothesis space $\mathcal{H} = \{h_0, h_1\}$, with

$$h_0(x_1) = 1 \quad h_0(x_2) = 1 \quad h_0(x_3) = 1 \quad h_0(x_4) = 0 \quad h_0(x_5) = 0$$

and

$$h_1(x_1) = 1 \quad h_1(x_2) = 1 \quad h_1(x_3) = 0 \quad h_1(x_4) = 1 \quad h_1(x_5) = 0$$

Note that $L(h_i, \mathbb{P}_i) = 0$ and $\Gamma^{EOp}(h_i, \mathbb{P}_i) = 0$ for both $i = 0, 1$. Note also that $L(h_1, \mathbb{P}_0) = L(h_0, \mathbb{P}_1) = \eta$. Moreover,

$$\begin{aligned} \Gamma^{EOp}(h_1, \mathbb{P}_0) &= \left|\mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_1(X) = 1|A = 0, Y = 1) - \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_1(X) = 1|A = 1, Y = 1)\right| \\ &= \left|\frac{P_{10} - \eta/2}{P_{10} - \eta/2 + \eta/2} - 1\right| \\ &= \frac{\eta}{2P_{10}} \end{aligned}$$

and similarly $\Gamma^{EOp}(h_0, \mathbb{P}_1) = \frac{\eta}{2P_{10}}$.

Consider a (randomized) malicious adversary $\mathfrak{A}_i$ of power α, that given a clean distribution $\mathbb{P}_i$, changes every marked point to $(x_3, \neg i, i)$ with probability 0.5 and to $(x_4, i, \neg i)$ otherwise. Under a distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, the probability of seeing a point $(x_3, i, \neg i)$ is $\frac{\eta}{2}(1 - \alpha) = \frac{\eta}{2}\frac{1}{1+\eta} = \alpha/2$, which is equal to the probability of seeing a point $(x_3, \neg i, i)$. Therefore, denoting the probability distribution of the corrupted dataset, under a clean distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, by $\mathbb{P}'_i$, we have

$$\mathbb{P}'_i(x, a, y) = \begin{cases} (1 - \alpha)P_{11} & \text{if } x = x_1, a = 1, y = 1 \\ (1 - \alpha)(P_{10} - \eta/2) & \text{if } x = x_2, a = 0, y = 1 \\ \alpha/2 & \text{if } x = x_3, a = i, y = \neg i \\ \alpha/2 & \text{if } x = x_3, a = \neg i, y = i \\ \alpha/2 & \text{if } x = x_4, a = \neg i, y = i \\ \alpha/2 & \text{if } x = x_4, a = i, y = \neg i \\ (1 - \alpha)(1 - P_{10} - P_{11} - \eta/2) & \text{if } x = x_5, a = 0, y = 0 \\ 0 & \text{otherwise} \end{cases}$$

In particular, $\mathbb{P}'_0 = \mathbb{P}'_1$, so the two initial distributions $\mathbb{P}_0$ and $\mathbb{P}_1$ become indistinguishable under the adversarial manipulation.

Fix an arbitrary learner $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X}\times\mathcal{A}\times\mathcal{Y})^n \to \{h_0, h_1\}$. Note that, if the clean distribution is $\mathbb{P}_0$, the events (in the probability space defined by the sampling of the poisoned train data)

$$\begin{aligned}\{L(\mathcal{L}(S^p), \mathbb{P}_0) - L(h_0, \mathbb{P}_0) \geq \eta\} &= \{\mathcal{L}(S^p) = h_1\} \\ &= \left\{\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_0) - \Gamma^{EOp}(h_0, \mathbb{P}_0) \geq \frac{\eta}{2P_{10}}\right\}\end{aligned}$$

are all the same. Similarly, if the clean distribution is $\mathbb{P}_1$

$$\begin{aligned}\{L(\mathcal{L}(S^p), \mathbb{P}_1) - L(h_1, \mathbb{P}_1) \geq \eta\} &= \{\mathcal{L}(S^p) = h_0\} \\ &= \left\{\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_1) - \Gamma^{EOp}(h_1, \mathbb{P}_1) \geq \frac{\eta}{2P_{10}}\right\}.\end{aligned}$$

Therefore, depending on whether we choose $\mathbb{P}_0$ or $\mathbb{P}_1$ as a clean distribution, we have

$$\begin{aligned}&\mathbb{P}_{S^p\sim\mathbb{P}'_0}\left(L(\mathcal{L}(S^p), \mathbb{P}_0) - L(h_0, \mathbb{P}_0) \geq \eta \wedge \Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_0) - \Gamma^{EOp}(h_0, \mathbb{P}_0) \geq \frac{\eta}{2P_{10}}\right) \\ &\quad = \mathbb{P}_{S^p\sim\mathbb{P}'_0}\left(\mathcal{L}(S^p) = h_1\right)\end{aligned}$$

and

$$\begin{aligned}&\mathbb{P}_{S^p\sim\mathbb{P}'_1}\left(L(\mathcal{L}(S^p), \mathbb{P}_1) - L(h_1, \mathbb{P}_1) \geq \eta \wedge \Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_1) - \Gamma^{EOp}(h_1, \mathbb{P}_1) \geq \frac{\eta}{2P_{10}}\right) \\ &\quad = \mathbb{P}_{S^p\sim\mathbb{P}'_1}\left(\mathcal{L}(S^p) = h_0\right)\end{aligned}$$

Finally, note that $\mathbb{P}'_0 = \mathbb{P}'_1$, so that either $\mathbb{P}_{S^p\sim\mathbb{P}'_0}\left(\mathcal{L}(S^p) = h_1\right) \geq 1/2$ or $\mathbb{P}_{S^p\sim\mathbb{P}'_1}\left(\mathcal{L}(S^p) = h_0\right) \geq 1/2$. Therefore, for at least one of $i = 0, 1$, both

$$L(\mathcal{L}(S^p), \mathbb{P}_i) - L(h_i, \mathbb{P}_i) \geq \eta = \frac{\alpha}{1-\alpha}$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_i) - \Gamma^{EOp}(h_i, \mathbb{P}_i) \geq \frac{\eta}{2P_{10}} = \frac{\alpha}{2P_{10}(1-\alpha)}$$

both hold with probability at least $1/2$. This concludes the proof of the first case.

Case 2 Now assume that $\frac{\alpha}{1-\alpha} > 2\min\{P_{10}, 1 - P_{10} - P_{11}\}$. We distinguish two cases:

Case 2.1 Suppose that $P_{10} \leq 1 - P_{10} - P_{11}$. We have that $\frac{\alpha}{1-\alpha} > 2P_{10}$. Then, denote by α_1 the unique number between $(0, 0.5)$, such that $\frac{\alpha_1}{1-\alpha_1} = 2P_{10} = 2\min\{P_{10}, 1 - P_{10} - P_{11}\}$, and note that $\alpha_1 < \alpha$. Then repeat the same construction as in Case 1, but with $\eta_1 = \frac{\alpha_1}{1-\alpha_1}$ and an adversary that with probability α_1/α does the same as in Case 1 and leaves a marked point untouched otherwise.

Then the same argument as in Case 1 gives that for some $i \in \{0, 1\}$, with probability at least 0.5, both of the following hold

$$L(\mathcal{L}(S^p), \mathbb{P}_i) - L(h_i, \mathbb{P}_i) \geq \frac{\alpha_1}{1-\alpha_1} = 2P_{10}$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_i) - \Gamma^{EOp}(h_i, \mathbb{P}_i) \geq \frac{\eta_1}{2P_{10}} = 1.$$

Case 2.2 In the case when $1 - P_{10} - P_{11} < P_{10}$ we have that $\frac{\alpha}{1-\alpha} > 2(1 - P_{10} - P_{11})$. Then, denote by α_2 the unique number between $(0, 0.5)$, such that $\frac{\alpha_2}{1-\alpha_2} = 2(1 - P_{10} - P_{11}) = 2\min\{P_{10}, 1 - P_{10} - P_{11}\}$, and note that $\alpha_2 < \alpha$. Then repeat the same construction as in Case 1, but with $\eta_2 = \frac{\alpha_2}{1-\alpha_2}$ and an adversary that with probability α_2/α does the same as in Case 1 and leaves a marked point untouched otherwise.

Then the same argument as in Case 1 gives that for some $i \in \{0, 1\}$, with probability at least 0.5, both of the following hold

$$L(\mathcal{L}(S^p), \mathbb{P}_i) - L(h_i, \mathbb{P}_i) \geq \frac{\alpha_2}{1 - \alpha_2} = 2(1 - P_{10} - P_{11})$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_i) - \Gamma^{EOp}(h_i, \mathbb{P}_i) \geq \frac{\eta_2}{2P_{10}} = \frac{1 - P_{10} - P_{11}}{P_{10}}.$$

This concludes the proof of Theorem 9. □

C.1.2 Hurting fairness without affecting accuracy - proofs

Theorem 10. *Let $0 \leq \alpha < 0.5, 0 < P_0 \leq 0.5$. For any input set $\mathcal{X}$ with at least four distinct points, there exists a finite hypothesis space $\mathcal{H}$, such that for any learning algorithm $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$, there exists a distribution $\mathbb{P}$ for which $\mathbb{P}(A = 0) = P_0$, a malicious adversary $\mathfrak{A}$ of power α and a hypothesis $h^* \in \mathcal{H}$, such that with probability at least 0.5*

$$L(\mathcal{L}(S^p), \mathbb{P}) = L(h^*, \mathbb{P}) = \min_{h\in\mathcal{H}} L(h, \mathbb{P})$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}) - \Gamma^{DP}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{2P_0(1 - P_0)(1 - \alpha)}, 1\right\} \geq \min\left\{\frac{\alpha}{2P_0}, 1\right\}.$$

Proof. Let $\eta = \frac{\alpha}{1-\alpha}$, so that $\alpha = \frac{\eta}{1+\eta}$.

Case 1 First assume that $\eta = \frac{\alpha}{1-\alpha} \leq 2P_0(1-P_0)$. Take four distinct points $\{x_1, x_2, x_3, x_4\} \in \mathcal{X}$. We consider two distributions $\mathbb{P}_0$ and $\mathbb{P}_1$, where each $\mathbb{P}_i$ is defined as

$$\mathbb{P}_i(x, a, y) = \begin{cases} 1 - P_0 - \eta/2 & \text{if } x = x_1, a = 1, y = 1 \\ P_0 - \eta/2 & \text{if } x = x_2, a = 0, y = 0 \\ \eta/2 & \text{if } x = x_3, a = i, y = 1 \\ \eta/2 & \text{if } x = x_4, a = \neg i, y = 1 \\ 0 & \text{otherwise} \end{cases}$$

Note that these are valid distributions, since $\eta \leq 2P_0(1 - P_0) \leq 2P_0 \leq 2(1 - P_0)$ by assumption and also that $P_0 = \mathbb{P}_i(A = 0)$ for both $i \in \{0, 1\}$. Consider the hypothesis space $\mathcal{H} = \{h_0, h_1\}$, with

$$h_0(x_1) = 1 \quad h_0(x_2) = 0 \quad h_0(x_3) = 1 \quad h_0(x_4) = 0$$

and

$$h_1(x_1) = 1 \quad h_1(x_2) = 0 \quad h_1(x_3) = 0 \quad h_1(x_4) = 1.$$

Note that $L(h_i, \mathbb{P}_i) = L(h_{\neg i}, \mathbb{P}_i) = \eta/2$ for both $i = 0, 1$. Moreover,

$$\begin{aligned}
\Gamma^{DP}(h_0, \mathbb{P}_0) &= \left|\mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_0(X) = 1|A = 0) - \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_0(X) = 1|A = 1)\right| \\
&= \left|\frac{\eta/2}{P_0 - \eta/2 + \eta/2} - \frac{1 - P_0 - \eta/2}{1 - P_0 - \eta/2 + \eta/2}\right| \\
&= \left|\frac{\eta}{2P_0} - \frac{2 - 2P_0 - \eta}{2(1 - P_0)}\right| \\
&= \left|\frac{\eta}{2P_0(1 - P_0)} - 1\right| \\
&= 1 - \frac{\eta}{2P_0(1 - P_0)},
\end{aligned}$$

since $\eta \leq 2P_0(1 - P_0)$ by assumption. Furthermore, $\Gamma^{DP}(h_1, \mathbb{P}_0) = 1$, so that $\Gamma^{DP}(h_1, \mathbb{P}_0) - \Gamma^{DP}(h_0, \mathbb{P}_0) = \frac{\eta}{2P_0(1-P_0)}$. Similarly,

$$\begin{aligned}
\Gamma^{DP}(h_1, \mathbb{P}_1) &= \left|\mathbb{P}_{(X,A,Y)\sim\mathbb{P}_1}(h_1(X) = 1|A = 0) - \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_1}(h_1(X) = 1|A = 1)\right| \\
&= \left|\frac{\eta/2}{P_0 - \eta/2 + \eta/2} - \frac{1 - P_0 - \eta/2}{1 - P_0 - \eta/2 + \eta/2}\right| \\
&= 1 - \frac{\eta}{2P_0(1 - P_0)}
\end{aligned}$$

and $\Gamma^{DP}(h_0, \mathbb{P}_1) = 1$.

Consider a (randomized) malicious adversary $\mathfrak{A}_i$ of power α, that given a clean distribution $\mathbb{P}_i$, changes every marked point to $(x_3, \neg i, 1)$ with probability 0.5 and to $(x_4, i, 1)$ otherwise. Under a distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, the probability of seeing a point $(x_3, i, 1)$ is $\frac{\eta}{2}(1 - \alpha) = \frac{\eta}{2}\frac{1}{1+\eta} = \alpha/2$, which is equal to the probability of seeing a point $(x_3, \neg i, 1)$. Therefore, denoting the probability distribution of the corrupted dataset, under a clean distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, by $\mathbb{P}'_i$, we have

$$\mathbb{P}'_i(x, a, y) = \begin{cases}
(1 - \alpha)(1 - P_0 - \eta/2) & \text{if } x = x_1, a = 1, y = 1 \\
(1 - \alpha)(P_0 - \eta/2) & \text{if } x = x_2, a = 0, y = 0 \\
\alpha/2 & \text{if } x = x_3, a = i, y = 1 \\
\alpha/2 & \text{if } x = x_3, a = \neg i, y = 1 \\
\alpha/2 & \text{if } x = x_4, a = \neg i, y = 1 \\
\alpha/2 & \text{if } x = x_4, a = i, y = 1 \\
0 & \text{otherwise}
\end{cases}$$

In particular, $\mathbb{P}'_0 = \mathbb{P}'_1$, so the two initial distributions $\mathbb{P}_0$ and $\mathbb{P}_1$ become indistinguishable under the adversarial manipulation.

Fix an arbitrary learner $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \{h_0, h_1\}$. Note that, if the clean distribution is $\mathbb{P}_0$, the events (in the probability space defined by the sampling of the poisoned train data)

$$\{\mathcal{L}(S^p) = h_1\} = \left\{\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_0) - \Gamma^{DP}(h_0, \mathbb{P}_0) \geq \frac{\eta}{2P_0(1 - P_0)}\right\}$$

are all the same. Similarly, if the clean distribution is $\mathbb{P}_1$

$$\{\mathcal{L}(S^p) = h_0\} = \left\{\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_1) - \Gamma^{DP}(h_1, \mathbb{P}_1) \geq \frac{\eta}{2P_0(1-P_0)}\right\}.$$

Therefore, depending on whether we choose $\mathbb{P}_0$ or $\mathbb{P}_1$ as a clean distribution, we have

$$\mathbb{P}_{S^p \sim \mathbb{P}'_0}\left(\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_0) - \Gamma^{DP}(h_0, \mathbb{P}_0) \geq \frac{\eta}{2P_0(1-P_0)}\right) = \mathbb{P}_{S^p \sim \mathbb{P}'_0}\left(\mathcal{L}(S^p) = h_1\right)$$

and

$$\mathbb{P}_{S^p \sim \mathbb{P}'_1}\left(\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_1) - \Gamma^{DP}(h_1, \mathbb{P}_1) \geq \frac{\eta}{2P_0(1-P_0)}\right) = \mathbb{P}_{S^p \sim \mathbb{P}'_1}\left(\mathcal{L}(S^p) = h_0\right)$$

Finally, note that $\mathbb{P}'_0 = \mathbb{P}'_1$, so that either $\mathbb{P}_{S^p \sim \mathbb{P}'_0}\left(\mathcal{L}(S^p) = h_1\right) \geq 1/2$ or $\mathbb{P}_{S^p \sim \mathbb{P}'_1}\left(\mathcal{L}(S^p) = h_0\right) \geq 1/2$. Furthermore, $L(\mathcal{L}(S^p), \mathbb{P}_i) = \eta/2$ holds for both $i \in \{0, 1\}$, for any realization of the randomness. Therefore, for at least one of $i = 0, 1$, both

$$L(\mathcal{L}(S^p), \mathbb{P}_i) = L(h_i, \mathbb{P}_i) = \frac{\eta}{2}$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_i) - \Gamma^{DP}(h_i, \mathbb{P}_i) \geq \frac{\eta}{2P_0(1-P_0)} = \frac{\alpha}{2P_0(1-P_0)(1-\alpha)}$$

both hold with probability at least $1/2$. This concludes the proof in the first case.

Case 2 Now suppose that $\eta = \frac{\alpha}{1-\alpha} > 2P_0(1-P_0)$. Let $\alpha_1 \in (0, 0.5)$ be such that $\frac{\alpha_1}{1-\alpha_1} = 2P_0(1-P_0)$. Note that since $f(x) = \frac{x}{1-x}$ is monotonically increasing in (0,1), α_1 is unique and $\alpha_1 < \alpha$.

Now repeat the same construction as in Case 1, but with $\eta_1 = \frac{\alpha_1}{1-\alpha_1} = 2P_0(1-P_0)$. For every marked point, the adversary does the same as in Case 1 with probability α_1/α and does not change the point otherwise. Then the same argument as in Case 1 shows that for one $i \in \{0, 1\}$, both

$$L(\mathcal{L}(S^p), \mathbb{P}_i) = L(h_i, \mathbb{P}_i) = \frac{\eta_1}{2} = P_0(1-P_0)$$

and

$$\Gamma^{DP}(\mathcal{L}(S^p), \mathbb{P}_i) - \Gamma^{DP}(h_i, \mathbb{P}_i) \geq \frac{\eta_1}{2P_0(1-P_0)} = 1$$

both hold with probability at least $1/2$. This concludes the proof of Theorem 10. □

Theorem 11. *Let $0 \leq \alpha < 0.5, P_{10} \leq P_{11} < 1$ be such that $P_{10} + P_{11} < 1$. For any input set $\mathcal{X}$ with at least five distinct points, there exists a finite hypothesis space $\mathcal{H}$, such that for any learning algorithm $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$, there exists a distribution $\mathbb{P}$ for which $\mathbb{P}(A = a, Y = 1) = P_{1a}$ for $a \in \{0, 1\}$, a malicious adversary $\mathfrak{A}$ of power α and a hypothesis $h^* \in \mathcal{H}$, such that with probability at least 0.5*

$$L(\mathcal{L}(S^p), \mathbb{P}) = L(h^*, \mathbb{P}) = \min_{h\in\mathcal{H}} L(h, \mathbb{P})$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}) - \Gamma^{EOp}(h^*, \mathbb{P}) \geq \min\left\{\frac{\alpha}{2(1-\alpha)P_{10}}\left(1 - \frac{P_{10}}{P_{11}}\right), 1 - \frac{P_{10}}{P_{11}}\right\}.$$

Proof. Let $\eta = \frac{\alpha}{1-\alpha}$, so that $\alpha = \frac{\eta}{1+\eta}$.

Case 1 First assume that $\eta \leq 2P_{10}$. Take five distinct points $\{x_1, x_2, x_3, x_4, x_5\} \in \mathcal{X}$. We consider two distributions $\mathbb{P}_0$ and $\mathbb{P}_1$, where each $\mathbb{P}_i$ is defined as

$$\mathbb{P}_i(x,a,y) = \begin{cases} P_{11} - \eta/2 & \text{if } x = x_1, a = 1, y = 1 \\ P_{10} - \eta/2 & \text{if } x = x_2, a = 0, y = 1 \\ \eta/2 & \text{if } x = x_3, a = i, y = 1 \\ \eta/2 & \text{if } x = x_4, a = \neg i, y = 1 \\ 1 - P_{10} - P_{11} & \text{if } x = x_5, a = 0, y = 0 \\ 0 & \text{otherwise} \end{cases}$$

Note that these are valid distributions, since $\eta \leq 2P_{10} \leq 2P_{11}$ by assumption, and that $P_{1a} = \mathbb{P}_i(A = a, Y = 1)$ for both $a \in \{0,1\}, i \in \{0,1\}$. Consider the hypothesis space $\mathcal{H} = \{h_0, h_1\}$, with

$$h_0(x_1) = 1 \quad h_0(x_2) = 1 \quad h_0(x_3) = 1 \quad h_0(x_4) = 0 \quad h_0(x_5) = 0$$

and

$$h_1(x_1) = 1 \quad h_1(x_2) = 1 \quad h_1(x_3) = 0 \quad h_1(x_4) = 1 \quad h_1(x_5) = 0$$

Note that $L(h_i, \mathbb{P}_i) = L(h_{\neg i}, \mathbb{P}_i) = \eta/2$. Moreover,

$$\begin{aligned} \Gamma^{EOp}(h_0, \mathbb{P}_0) &= \left| \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_1(X) = 1 | A = 0, Y = 1) - \mathbb{P}_{(X,A,Y)\sim\mathbb{P}_0}(h_1(X) = 1 | A = 1, Y = 1) \right| \\ &= \left| 1 - \frac{P_{11} - \eta/2}{P_{11} - \eta/2 + \eta/2} \right| \\ &= \frac{\eta}{2P_{11}} \end{aligned}$$

and similarly $\Gamma^{EOp}(h_1, \mathbb{P}_0) = \frac{\eta}{2P_{10}}$. Since $P_{10} \leq P_{11}$, $\Gamma^{EOp}(h_0, \mathbb{P}_0) \leq \Gamma^{EOp}(h_1, \mathbb{P}_0)$ and

$$\Gamma^{EOp}(h_1, \mathbb{P}_0) - \Gamma^{EOp}(h_0, \mathbb{P}_0) = \frac{\eta}{2P_{10}} \left(1 - \frac{P_{10}}{P_{11}}\right).$$

Similarly $\Gamma^{EOp}(h_0, \mathbb{P}_1) = \frac{\eta}{2P_{10}}$ and $\Gamma^{EOp}(h_1, \mathbb{P}_1) = \frac{\eta}{2P_{11}}$, so that $\Gamma^{EOp}(h_1, \mathbb{P}_1) \leq \Gamma^{EOp}(h_0, \mathbb{P}_1)$ and

$$\Gamma^{EOp}(h_0, \mathbb{P}_1) - \Gamma^{EOp}(h_1, \mathbb{P}_1) = \frac{\eta}{2P_{10}} \left(1 - \frac{P_{10}}{P_{11}}\right).$$

Consider a (randomized) malicious adversary $\mathfrak{A}_i$ of power α, that given a clean distribution $\mathbb{P}_i$, changes every marked point to $(x_3, \neg i, 1)$ with probability 0.5 and to $(x_4, i, 1)$ otherwise. Under a distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, the probability of seeing a point $(x_3, i, 1)$ is $\frac{\eta}{2}(1-\alpha) = \frac{\eta}{2}\frac{1}{1+\eta} = \alpha/2$, which is equal to the probability of seeing a point $(x_3, \neg i, 1)$. Therefore, denoting the probability distribution of the corrupted dataset, under a clean distribution $\mathbb{P}_i$ and an adversary $\mathfrak{A}_i$, by $\mathbb{P}'_i$, we have

$$\mathbb{P}'_i(x,a,y) = \begin{cases} (1-\alpha)(P_{11} - \eta/2) & \text{if } x = x_1, a = 1, y = 1 \\ (1-\alpha)(P_{10} - \eta/2) & \text{if } x = x_2, a = 0, y = 1 \\ \alpha/2 & \text{if } x = x_3, a = i, y = 1 \\ \alpha/2 & \text{if } x = x_3, a = \neg i, y = 1 \\ \alpha/2 & \text{if } x = x_4, a = \neg i, y = 1 \\ \alpha/2 & \text{if } x = x_4, a = i, y = 1 \\ (1-\alpha)(1 - P_{10} - P_{11}) & \text{if } x = x_5, a = 0, y = 0 \\ 0 & \text{otherwise} \end{cases}$$

In particular, $\mathbb{P}'_0 = \mathbb{P}'_1$, so the two initial distributions $\mathbb{P}_0$ and $\mathbb{P}_1$ become indistinguishable under the adversarial manipulation.

Fix an arbitrary learner $\mathcal{L} : \cup_{n\in\mathbb{N}}(\mathcal{X}\times\mathcal{A}\times\mathcal{Y})^n \to \{h_0, h_1\}$. Note that, if the clean distribution is $\mathbb{P}_0$, the events (in the probability space defined by the sampling of the poisoned train data)

$$\{\mathcal{L}(S^p) = h_1\} = \left\{\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_0) - \Gamma^{EOp}(h_0, \mathbb{P}_0) \geq \frac{\eta}{2P_{10}}\left(1 - \frac{P_{10}}{P_{11}}\right)\right\}$$

are all the same. Similarly, if the clean distribution is $\mathbb{P}_1$

$$\{\mathcal{L}(S^p) = h_0\} = \left\{\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_1) - \Gamma^{EOp}(h_1, \mathbb{P}_1) \geq \frac{\eta}{2P_{10}}\left(1 - \frac{P_{10}}{P_{11}}\right)\right\}.$$

Therefore, depending on whether we choose $\mathbb{P}_0$ or $\mathbb{P}_1$ as a clean distribution, we have

$$\mathbb{P}_{S^p\sim\mathbb{P}'_0}\left(\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_0) - \Gamma^{EOp}(h_0, \mathbb{P}_0) \geq \frac{\eta}{2P_{10}}\left(1 - \frac{P_{10}}{P_{11}}\right)\right) = \mathbb{P}_{S^p\sim\mathbb{P}'_0}\left(\mathcal{L}(S^p) = h_1\right)$$

and

$$\mathbb{P}_{S^p\sim\mathbb{P}'_1}\left(\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_1) - \Gamma^{EOp}(h_1, \mathbb{P}_1) \geq \frac{\eta}{2P_{10}}\left(1 - \frac{P_{10}}{P_{11}}\right)\right) = \mathbb{P}_{S^p\sim\mathbb{P}'_1}\left(\mathcal{L}(S^p) = h_0\right)$$

Finally, note that $\mathbb{P}'_0 = \mathbb{P}'_1$, so that either $\mathbb{P}_{S^p\sim\mathbb{P}'_0}\left(\mathcal{L}(S^p) = h_1\right) \geq 1/2$ or $\mathbb{P}_{S^p\sim\mathbb{P}'_1}\left(\mathcal{L}(S^p) = h_0\right) \geq 1/2$. Moreover, $L(\mathcal{L}(S^p), \mathbb{P}_i) = L(h_i, \mathbb{P}_i) = \eta/2$ holds for both $i \in \{0, 1\}$, for any realization of the randomness. Therefore, for at least one of $i = 0, 1$, both

$$L(\mathcal{L}(S^p), \mathbb{P}_i) = L(h_i, \mathbb{P}_i) = \frac{\eta}{2}$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_i) - \Gamma^{EOp}(h_i, \mathbb{P}_i) \geq \frac{\eta}{2P_{10}}\left(1 - \frac{P_{10}}{P_{11}}\right) = \frac{\alpha}{2P_{10}(1-\alpha)}\left(1 - \frac{P_{10}}{P_{11}}\right)$$

both hold with probability at least $1/2$. This concludes the proof in the first case.

Case 2 Now assume that $\frac{\alpha}{1-\alpha} > 2P_{10}$. Then denote by α_1 the unique number between $(0, 0.5)$, such that $\frac{\alpha_1}{1-\alpha_1} = 2P_{10}$, and note that $\alpha_1 < \alpha$. Then repeat the same construction as in Case 1, but with $\eta_1 = \frac{\alpha_1}{1-\alpha_1}$ and an adversary that with probability α_1/α does the same as in Case 1 and leaves a marked point untouched otherwise.

Then the same argument as in Case 1 gives that for some $i \in \{0, 1\}$, with probability at least 0.5, both of the following hold

$$L(\mathcal{L}(S^p), \mathbb{P}_i) = L(h_i, \mathbb{P}_i) = \frac{\eta_1}{2} = P_{10}$$

and

$$\Gamma^{EOp}(\mathcal{L}(S^p), \mathbb{P}_i) - \Gamma^{EOp}(h_i, \mathbb{P}_i) \geq \frac{\eta_1}{2P_{10}} = \frac{\eta_1}{2P_{10}}\left(1 - \frac{P_{10}}{P_{11}}\right) = 1 - \frac{P_{10}}{P_{11}}.$$

This concludes the proof of Theorem 11. □

C.2 Upper bounds proofs

We now present the complete proofs of our upper bounds. The main challenge lies in understanding the concentration properties of the empirical estimates of the fairness measures, as introduced in the main body of the paper. To this end, we first bound the effect that the data corruption may have on these estimates. We then leverage classic concentration techniques to relate the "ideal" clean data estimates to the corresponding population fairness measures.

C.2.1 Concentration tools and notation

We will use the following versions of the classic Chernoff bounds for large deviations of Binomial random variables, as they can be found, for example, in [KL93]. Let $X \sim Bin(n, p)$. Then

$$\mathbb{P}(X \leq (1-\alpha)pn) \leq e^{-\alpha^2 np/2}$$

and

$$\mathbb{P}(X \geq (1+\alpha)pn) \leq e^{-\alpha^2 np/3},$$

for any $\alpha \in (0, 1)$. We will also use the Hoeffding's inequality [Hoe63]. Let $X_1, X_2, \ldots, X_n$ be independent random variables, such that each X_i is bounded in $[a_i, b_i]$ and let $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$. Then

$$\mathbb{P}\left(\left|\bar{X} - \mathbb{E}(\bar{X})\right| > t\right) \leq 2\exp\left(-\frac{2n^2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right).$$

Throughout the section we denote the clean data distribution by $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$. As in the main body of the paper, we denote $P_a = \mathbb{P}(A = a)$ and $P_{1a} = \mathbb{P}(Y = 1, A = a)$ for both $a \in \{0, 1\}$. We assume without loss of generality that $0 < P_0 \leq \frac{1}{2} \leq P_1$ (when studying demographic parity) and $0 < P_{10} \leq P_{11}$ (when studying equality of opportunity).

We will be interested in the concentration properties of certain empirical estimates based on the corrupted data S^p. Therefore, we denote the distribution that corresponds to all the randomness of the sampling of S^p, that is the randomness of the clean data, the marked points and the adversary, by $\mathbb{P}^{\mathfrak{A}}$. Here we consider both $\mathbb{P}$ and $\mathfrak{A}$ arbitrary, but fixed.

C.2.2 Concentration results

We study the concentration of the demographic parity and the equality of opportunity fairness estimates in Sections C.2.2 and C.2.2 respectively.

Concentration for demographic parity

We use the notation $C_a = \sum_{i=1}^{n} \mathbb{1}\{a_i^p = a, i \notin \mathfrak{P}\}$ for the number of points in S^p that *were not* marked (that is, are *clean*) and contain a point from protected group a and $B_a = \sum_{i=1}^{n} \mathbb{1}\{a_i^p = a, i \in \mathfrak{P}\}$ for the number of points in S^p that *were* marked (that is, are potentially *bad*[1]) and contain a point from protected group a. Note that $B_0 + B_1 = |\mathfrak{P}|$ is the total number of poisoned points, which is $\mathrm{Bin}(n, \alpha)$, and $B_0 + B_1 = n - C_0 - C_1$. Similarly, denote by

[1]We use B_a with B for *bad* here, instead of P for *poisoned*, to avoid confusion with the protected group frequencies P_i.

$C_a^1(h) = \sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a, i \notin \mathfrak{P}\}$ and $B_a^1(h) = \sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a, i \in \mathfrak{P}\}$. Denote

$$\gamma_a^p(h) = \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a\}}$$

and

$$\gamma_a(h) = \mathbb{P}(h(X) = 1 | A = a),$$

so that $\widehat{\Gamma}^{DP}(h) = |\gamma_0^p(h) - \gamma_1^p(h)|$ and $\Gamma^{DP}(h) = |\gamma_0(h) - \gamma_1(h)|$. Note that $\gamma_a^p(h)$ is an estimate of a conditional probability *based on the corrupted data*. We now introduce the corresponding estimate that only uses the clean (but unknown) subset of the training set S^p

$$\gamma_a^c(h) = \frac{C_a^1(h)}{C_a(h)} = \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a, i \notin \mathfrak{P}\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a, i \notin \mathfrak{P}\}}.$$

First we bound how far the corrupted estimates $\gamma_a^p(h)$ of $\gamma_a(h)$ are from the clean estimates $\gamma_a^c(h)$, uniformly over the hypothesis space $\mathcal{H}$:

Proposition 2. *If* $n \geq \max\left\{\frac{8\log(4/\delta)}{(1-\alpha)P_0}, \frac{12\log(3/\delta)}{\alpha}\right\}$, *we have*

$$\mathbb{P}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}\left(|\gamma_0^p(h) - \gamma_0^c(h)| + |\gamma_1^p(h) - \gamma_1^c(h)|\right) \geq \frac{2\alpha}{P_0/3 + \alpha}\right) < \delta. \tag{C.1}$$

Proof. First we show that certain bounds on the random variables B_a and C_a hold with high probability. Then we show that the supremum in equation (C.1) is bounded when these bounds hold.

Step 1 Specifically, since $B_0 + B_1 \sim \mathrm{Bin}(n, \alpha)$, by the Chernoff bounds and the assumption on n

$$\mathbb{P}^{\mathfrak{A}}\left(B_0 + B_1 \geq \frac{3\alpha}{2}n\right) \leq e^{-\alpha n/12} \leq \frac{\delta}{3}.$$

Similarly, $C_0 \sim \mathrm{Bin}(n, (1-\alpha)P_0)$ and $C_1 \sim \mathrm{Bin}(n, (1-\alpha)P_1)$ and since $P_0 \leq P_1$ we get

$$\mathbb{P}^{\mathfrak{A}}\left(C_0 \leq \frac{1-\alpha}{2}P_0 n\right) \leq e^{-(1-\alpha)P_0 n/8} \leq \frac{\delta}{4}$$

and

$$\mathbb{P}^{\mathfrak{A}}\left(C_1 \leq \frac{1-\alpha}{2}P_1 n\right) \leq e^{-(1-\alpha)P_1 n/8} \leq \frac{\delta}{4}$$

Therefore, by a union bound

$$\mathbb{P}^{\mathfrak{A}}\left(\left(B_0 + B_1 \geq \frac{3\alpha}{2}n\right) \vee \left(C_0 \leq \frac{1-\alpha}{2}P_0 n\right) \vee \left(C_1 \leq \frac{1-\alpha}{2}P_1 n\right)\right) \leq \frac{\delta}{3} + \frac{\delta}{4} + \frac{\delta}{4} < \delta.$$

Step 2 Now assume that all of $B_0 + B_1 < \frac{3\alpha}{2}n$, $C_0 > \frac{1-\alpha}{2}P_0 n$, $C_1 > \frac{1-\alpha}{2}P_1 n$ hold. This happens with probability at least $1-\delta$ according to Step 1. Let h be an arbitrary classifier. Since we consider h fixed, we will drop the dependence on h from the notation for the rest of this proof and write $\gamma_a^p = \gamma_a^p(h), C_a^1 = C_a^1(h)$, etc.

We now prove that for both $a \in \{0,1\}$

$$\Delta_a := |\gamma_a^p - \gamma_a^c| \leq \frac{B_a}{C_a + B_a}. \tag{C.2}$$

For each $a \in \{0,1\}$, this can be shown as follows. First, if $\sum_{i=1}^n \mathbb{1}\{a_i^p = a\} = B_a + C_a = 0$, then both $\gamma_a^p(h)$ and $\gamma_a^c(h)$ are equal to 0, because of the convention that $\frac{0}{0} = 0$. In addition, $B_a = C_a = 0$. Therefore, inequality (C.2) trivially holds.

Similarly, if $B_a = 0$, but $C_a > 0$, then $\gamma_a^p(h) = \gamma_a^c(h)$ and so $\Delta_a = 0$ and (C.2) holds.

Assume now that $B_a > 0$. Note that if $C_a = \sum_{i=1}^n \mathbb{1}\{a_i^p = a, i \notin \mathfrak{P}\} = 0$, then

$$\Delta_a = |\gamma_a^p(h) - \gamma_a^c(h)| = \left|\frac{B_a^1}{B_a} - 0\right| = \frac{B_a^1}{B_a} = \frac{B_a^1}{B_a + C_a} \leq \frac{B_a}{C_a + B_a}.$$

Finally, assume that both $C_a > 0$ and $B_a > 0$. Note that under any realization of the randomness of the data sampling and the adversary, for any $a \in \{0,1\}$

$$\begin{aligned}
\gamma_a^p(h) &= \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a\}} \\
&= \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a, i \notin \mathfrak{P}\} + \sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a, i \in \mathfrak{P}\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a, i \notin \mathfrak{P}\} + \sum_{i=1}^n \mathbb{1}\{a_i^p = a, i \in \mathfrak{P}\}} \\
&= \frac{C_a^1 + B_a^1}{C_a + B_a}.
\end{aligned}$$

Therefore,

$$\Delta_a = |\gamma_a^p - \gamma_a^c| = \left|\frac{C_a^1 + B_a^1}{C_a + B_a} - \frac{C_a^1}{C_a}\right| = \frac{B_a}{C_a + B_a}\left|\frac{C_a^1}{C_a} - \frac{B_a^1}{B_a}\right| \leq \frac{B_a}{C_a + B_a}$$

and so (C.2) holds in all cases. Therefore, we can bound the sum $\Delta_0 + \Delta_1$ as follows:

$$\begin{aligned}
\Delta_0 + \Delta_1 &\leq \frac{B_0}{C_0 + B_0} + \frac{B_1}{C_1 + B_1} \\
&< \frac{B_0}{\frac{1-\alpha}{2}P_0 n + B_0} + \frac{B_1}{\frac{1-\alpha}{2}P_1 n + B_1} \\
&\leq \frac{B_0}{\frac{1-\alpha}{2}P_0 n + B_0} + \frac{B_1}{\frac{1-\alpha}{2}P_0 n + B_1} \\
&= \frac{B_0}{\frac{1-\alpha}{2}P_0 n + B_0} + 1 - \frac{\frac{1-\alpha}{2}P_0 n}{\frac{1-\alpha}{2}P_0 n - B_0 + (B_0 + B_1)} \\
&< \frac{B_0}{\frac{1-\alpha}{2}P_0 n + B_0} + 1 - \frac{\frac{1-\alpha}{2}P_0 n}{\frac{1-\alpha}{2}P_0 n - B_0 + \frac{3\alpha}{2}n} \\
&= 2 - (1-\alpha)P_0 n\left(\frac{1}{(1-\alpha)P_0 n + 2B_0} + \frac{1}{(1-\alpha)P_0 n + 3\alpha n - 2B_0}\right)
\end{aligned}$$

Studying the function $f(x) = \frac{1}{(1-\alpha)P_0 n + 2x} + \frac{1}{(1-\alpha)P_0 n + 3\alpha n - 2x}$, we see that

$$f'(x) = 2\left(\frac{1}{((1-\alpha)P_0 n + 3\alpha n - 2x)^2} - \frac{1}{((1-\alpha)P_0 n + 2x)^2}\right).$$

Note that $B_0 \leq B_0 + B_1 < \frac{3\alpha}{2}$, so we may assume $0 \leq x < \frac{3\alpha}{2}$. Therefore, both $(1-\alpha)P_0 n + 3\alpha n - 2x > 0$ and $(1-\alpha)P_0 n + 2x > 0$. Therefore, $f'(x) = 0$ if and only if $(1-\alpha)P_0 n + 3\alpha n - 2x = (1-\alpha)P_0 n + 2x$, that is, $x = \frac{3\alpha}{4}n$. Moreover $f'(x) < 0$ if $x \in [0, \frac{3\alpha}{4}n)$ and $f'(x) > 0$ if $x \in (\frac{3\alpha}{4}n, \frac{3\alpha}{2}n)$. Therefore, $f(x)$ is minimized at $x = \frac{3\alpha}{4}n$ and so

$$\begin{aligned}
\Delta_0 + \Delta_1 &\leq 2 - (1-\alpha)P_0 n\left(\frac{1}{(1-\alpha)P_0 n + 2B_0} + \frac{1}{(1-\alpha)P_0 n + 3\alpha n - 2B_0}\right) \\
&\leq 2 - (1-\alpha)P_0 n\left(\frac{1}{(1-\alpha)P_0 n + \frac{3\alpha}{2}n} + \frac{1}{(1-\alpha)P_0 n + 3\alpha n - \frac{3\alpha}{2}n}\right) \\
&= \frac{6\alpha}{2(1-\alpha)P_0 + 3\alpha} \\
&\leq \frac{6\alpha}{P_0 + 3\alpha} = \frac{2\alpha}{P_0/3 + \alpha}
\end{aligned}$$

and hence (C.2) holds in this case as well. Since the derivations hold for any classifier $h \in \mathcal{H}$, the result follows. □

For the rest of the section, we keep the notation $\Delta_a(h) = |\gamma_a^p(h) - \gamma_a^c(h)|$ for $a \in \{0,1\}$ and $\Delta^{DP} = \frac{2\alpha}{P_0/3+\alpha}$.

Next we use the previous result and the technique of [WGOS17] for proving concentration results about conditional probability estimates to bound the probability of a large deviation of $\widehat{\Gamma}^{DP}(h)$ from $\Gamma^{DP}(h)$, for a fixed hypothesis $h \in \mathcal{H}$.

Lemma 3. *Let $h \in \mathcal{H}$ be a fixed hypothesis and $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be a fixed distribution. Denote $P_a = \mathbb{P}(A = a)$ for $a \in \{0,1\}$. Let $\mathfrak{A}$ be any malicious adversary and denote by $\mathbb{P}^{\mathfrak{A}}$ the probability distribution of the poisoned data S^p, under the random sampling of the clean data, the marked points and the randomness of the adversary. Then for any $n \geq \max\left\{\frac{8\log(8/\delta)}{(1-\alpha)P_0}, \frac{12\log(6/\delta)}{\alpha}\right\}$ and $\delta \in (0,1)$*

$$\mathbb{P}^{\mathfrak{A}}\left(\left|\widehat{\Gamma}^{DP}(h) - \Gamma^{DP}(h)\right| \leq \Delta^{DP} + 2\sqrt{\frac{\log(16/\delta)}{n(1-\alpha)P_0}}\right) \geq 1 - \delta. \tag{C.3}$$

Proof. Again we write $\gamma_a^p = \gamma_a^p(h), C_a^1 = C_a^1(h)$, etc. since h is fixed. First we study the concentration of the clean estimate $\frac{C_a^1}{C_a}$ around γ_a. To this end, denote by $S_a^c = \{i : a_i^p = a, i \notin \mathfrak{P}\}$ the set of indexes of the poisoned data for which the protected group is a and the corresponding point was not marked for the adversary. Notice that S_a^c is a random variable and that $|S_a^c| = C_a$. Since $n \geq \frac{8\log(8/\delta)}{(1-\alpha)P_a}$ for both $a \in \{0,1\}$, we have

$$\begin{aligned}
\mathbb{P}^{\mathfrak{A}}\left(|\gamma_a^c - \gamma_a| > t\right) &= \sum_{S_a^c} \mathbb{P}^{\mathfrak{A}}\left(|\gamma_a^c - \gamma_a| > t | S_a^c\right) \mathbb{P}(S_a^c) \\
&\leq \mathbb{P}^{\mathfrak{A}}\left(C_a \leq \frac{(1-\alpha)}{2} P_a n\right)
\end{aligned}$$

$$
\begin{aligned}
&+ \sum_{S_a^c : C_a > \frac{(1-\alpha)}{2} P_a n} \mathbb{P}^{\mathfrak{A}} \left(\left| \frac{C_a^1}{C_a} - \gamma_a \right| > t \middle| S_a^c \right) \mathbb{P}^{\mathfrak{A}}(S_a^c) \\
&\leq \exp\left(-\frac{(1-\alpha) P_a n}{8} \right) + \sum_{S_a^p : C_a > \frac{(1-\alpha)}{2} P_a n} 2 \exp\left(-2t^2 C_a \right) \mathbb{P}^{\mathfrak{A}}(S_a^c) \\
&\leq \frac{\delta}{8} + 2 \exp\left(-t^2 (1-\alpha) P_a n \right),
\end{aligned}
$$

where the second inequality follows from Hoeffding's inequality. Note that this step crucially uses that the marked indexes are independent of the data. The triangle law gives

$$
\begin{aligned}
||\gamma_0^p - \gamma_1^p| - |\gamma_0 - \gamma_1|| \leq |\gamma_0^p - \gamma_1^p - \gamma_0 + \gamma_1| &\leq |\gamma_0^p - \gamma_0| + |\gamma_1^p - \gamma_1| \\
&\leq |\gamma_0^p - \gamma_0^c| + |\gamma_0^c - \gamma_0| + |\gamma_1^p - \gamma_1^c| + |\gamma_1^c - \gamma_1| \\
&= |\gamma_0^c - \gamma_0| + |\gamma_1^c - \gamma_1| + \Delta_0 + \Delta_1.
\end{aligned}
$$

Combining the previous two results (recall that we assume $P_0 \leq P_1$)

$$
\begin{aligned}
&\mathbb{P}^{\mathfrak{A}}(||\gamma_0^p - \gamma_1^p| - |\gamma_0 - \gamma_1|| > 2t + \Delta_0 + \Delta_1) \\
&\quad \leq \mathbb{P}^{\mathfrak{A}} \left(|\gamma_0^c - \gamma_0| + |\gamma_1^c - \gamma_1| + \Delta_0 + \Delta_1 > 2t + \Delta_0 + \Delta_1 \right) \\
&\quad \leq \mathbb{P}^{\mathfrak{A}} \left((|\gamma_0^c - \gamma_0| > t) \vee (|\gamma_1^c - \gamma_1| > t) \right) \\
&\quad \leq \mathbb{P}^{\mathfrak{A}} \left(|\gamma_0^c - \gamma_0| > t \right) + \mathbb{P}^{\mathfrak{A}} \left(|\gamma_1^c - \gamma_1| > t \right) \\
&\quad \leq \frac{\delta}{4} + 4 \exp(-t^2 n (1-\alpha) P_0).
\end{aligned}
$$

Setting $t = t_0 = \sqrt{\frac{\log(16/\delta)}{n(1-\alpha)P_0}}$ gives

$$
\mathbb{P}^{\mathfrak{A}} \left(||\gamma_0^p - \gamma_1^p| - |\gamma_0 - \gamma_1|| > \Delta_0 + \Delta_1 + 2\sqrt{\frac{\log(16/\delta)}{n(1-\alpha)P_0}} \right) \leq \frac{\delta}{4} + 4\frac{\delta}{16} = \frac{\delta}{2}. \tag{C.4}
$$

In addition Proposition 2 gives

$$
\mathbb{P}^{\mathfrak{A}} \left(\Delta_0 + \Delta_1 > \Delta^{DP} \right) \leq \frac{\delta}{2}. \tag{C.5}
$$

Using (C.4) and (C.5) we obtain that:

$$
\begin{aligned}
&\mathbb{P}^{\mathfrak{A}} \left(||\gamma_0^p - \gamma_1^p| - |\gamma_0 - \gamma_1|| \leq \Delta^{DP} + 2\sqrt{\frac{\log(16/\delta)}{N(1-\alpha)P_0}} \right) \\
&\geq \mathbb{P}^{\mathfrak{A}} \left(\left(||\gamma_0^p - \gamma_1^p| - |\gamma_0 - \gamma_1|| \leq \Delta_0 + \Delta_1 + 2\sqrt{\frac{\log(16/\delta)}{N(1-\alpha)P_0}} \right) \wedge \left(\Delta_0 + \Delta_1 \leq \Delta^{DP} \right) \right) \\
&\geq 1 - \frac{\delta}{2} - \frac{\delta}{2} = 1 - \delta.
\end{aligned}
$$

□

Finally, we show how to extend the previous result to hold uniformly over the whole hypothesis space, provided that $\mathcal{H}$ has a finite VC-dimension $d := VC(\mathcal{H})$

Lemma 4. *Under the setup of Lemma 3, assume additionally that $\mathcal{H}$ has a finite VC-dimension d. Then for any $n \geq \max\left\{\frac{8\log(8/\delta)}{(1-\alpha)P_0}, \frac{12\log(6/\delta)}{\alpha}, \frac{d}{2}\right\}$ and $\delta \in (0,1)$*

$$\mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}}|\widehat{\Gamma}^{DP}(h) - \Gamma^{DP}(h)| \leq \Delta^{DP} + 16\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(48/\delta)}{(1-\alpha)P_0 n}}\right) \geq 1-\delta. \quad \text{(C.6)}$$

Proof. From Proposition 2, we have that whenever $n \geq \max\left\{\frac{8\log(8/\delta)}{(1-\alpha)P_0}, \frac{12\log(6/\delta)}{\alpha}\right\}$ and $\delta \in (0,1)$

$$\mathbb{P}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}\left(\Delta_0(h) + \Delta_1(h)\right) \geq \Delta^{DP}\right) < \frac{\delta}{2}. \quad \text{(C.7)}$$

Additionally, in the proof of Lemma 3 we showed that for a fixed classifier $h \in \mathcal{H}$ for any $\delta \in (0,1), t \in (0,1)$ and both $a \in \{0,1\}$, we have

$$\begin{aligned}\mathbb{P}^{\mathfrak{A}}\left(|\gamma^c_a(h) - \gamma_a(h)| > t\right) &\leq \exp\left(-\frac{(1-\alpha)P_a n}{8}\right) + 2\exp\left(-t^2(1-\alpha)P_a n\right) \\ &\leq 3\exp\left(-\frac{t^2(1-\alpha)P_a n}{8}\right). \end{aligned} \quad \text{(C.8)}$$

The proof consists of two steps. In Steps 1 and 2 we show how to extend inequality (C.8) to hold uniformly over $\mathcal{H}$. Then, we combine the two uniform bounds with a similar argument as in the proof of Lemma 3.

The first step uses the classic symmetrization technique [Vap13] for proving bounds uniformly over hypothesis spaces of finite VC dimension. However, since the objective is different from the 0-1 loss, care is needed to ensure that the proof goes through, so we present it here in full detail.

Step 1 To make the dependence of the left-hand side of (C.8) on both h and the data S^p explicit, we set $\gamma^c_a(h, S^p) := \frac{C^1_a(h)}{C_a}$.

Introduce a ghost sample $S^1 = \{(x^1_i, a^1_i, y^1_i)\}_{i=1}^n$ also sampled in an i.i.d. manner from $\mathbb{P}^{\mathfrak{A}}$, that is, S^1 is another, independent poisoned dataset [2]. Let $\gamma^c_a(h, S^1)$ be the empirical estimate of $\gamma_a(h)$ based on S^1.

First we show a symmetrization inequality for the γ_a measures

$$\mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}}|\gamma_a(h) - \gamma^c_a(h, S^p)| \geq t\right) \leq 2\mathbb{P}^{\mathfrak{A}}_{S^p, S^1}\left(\sup_{h\in\mathcal{H}}\left|\gamma^c_a(h, S^1) - \gamma^c_a(h, S^p)\right| \geq t/2\right), \quad \text{(C.9)}$$

for any constant $1 > t \geq 2\sqrt{\frac{8\log(6)}{(1-\alpha)P_0 n}}$.

Indeed, let h^* be the hypothesis achieving the supremum on the left-hand side [3]. Note that

$$\mathbb{1}\left(|\gamma_a(h^*) - \gamma^c_a(h^*, S^p)| \geq t\right)\mathbb{1}\left(\left|\gamma_a(h^*) - \gamma^c_a(h^*, S^1)\right| \leq t/2\right)$$

[2] Formally, we associate S^1 also with a set $\mathfrak{P}_1$ of marked indexes.

[3] If the supremum is not attained, this argument can be repeated for each element of a sequence of classifiers approaching the supremum

$$\leq \mathbb{1}\left(\left|\gamma_a^c(h^*, S^1) - \gamma_a^c(h^*, S^p)\right| \geq t/2\right).$$

Taking expectation with respect to S^1

$$\begin{aligned}\mathbb{1}\left(|\gamma_a(h^*) - \gamma_a^c(h^*, S^p)| \geq t\right)&\mathbb{P}_{S^1}^{\mathfrak{A}}\left(\left|\gamma_a(h^*) - \gamma_a^c(h^*, S^1)\right| \leq t/2\right)\\ &\leq \mathbb{P}_{S^1}^{\mathfrak{A}}\left(\left|\gamma_a^c(h^*, S^1) - \gamma_a^c(h^*, S^p)\right| \geq t/2\right).\end{aligned}$$

Now using Lemma 3

$$\begin{aligned}\mathbb{P}_{S^1}^{\mathfrak{A}}\left(\left|\gamma_a(h^*) - \gamma_a^c(h^*, S^1)\right| \leq t/2\right) &\geq \mathbb{P}_{S^1}^{\mathfrak{A}}\left(\left|\gamma_a(h^*) - \gamma_a^c(h^*, S^1)\right| \leq \sqrt{\frac{8\log(6)}{(1-\alpha)P_a n}}\right)\\ &\geq 1 - \frac{1}{2}\\ &= \frac{1}{2}.\end{aligned}$$

so

$$\frac{1}{2}\mathbb{1}(|\gamma_a(h^*) - \gamma_a^c(h^*, S^p)| \geq t) \leq \mathbb{P}_{S^1}^{\mathfrak{A}}(\left|\gamma_a^c(h^*, S^1) - \gamma_a^c(h^*, S^p)\right| \geq t/2).$$

Taking expectation with respect to S^p

$$\begin{aligned}\mathbb{P}_{S^p}^{\mathfrak{A}}(|\gamma_a(h^*) - \gamma_a^c(h^*, S^p)| \geq t) &\leq 2\mathbb{P}_{S^p,S^1}^{\mathfrak{A}}(\left|\gamma_a^c(h^*, S^1) - \gamma_a^c(h^*, S^p)\right| \geq t/2)\\ &\leq 2\mathbb{P}_{S^p,S^1}^{\mathfrak{A}}(\sup_{h\in\mathcal{H}}\left|\gamma_a^c(h, S^1) - \gamma_a^c(h, S^p)\right| \geq t/2).\end{aligned}$$

Step 2 Next we use the growth function of $\mathcal{H}$ and the symmetrization inequality (C.9) to bound the large deviations of $\gamma_a^c(h)$ uniformly over $\mathcal{H}$.

Specifically, gives n points $x_1, \ldots, x_n \in \mathcal{X}$, denote

$$\mathcal{H}_{x_1,\ldots,x_n}\{(h(x_1), \ldots, h(x_n)) : h \in \mathcal{H}\}.$$

Then define the growth function of $\mathcal{H}$ as

$$S_{\mathcal{H}}(n) = \sup_{x_1,\ldots,x_n} |\mathcal{H}_{x_1,\ldots,x_n}|.$$

We will use that well-known Sauer's lemma (see, for example, [BBL03]), which states that whenever $n \geq d$, $S_{\mathcal{H}}(n) \leq \left(\frac{en}{d}\right)^d$

Notice that given the two datasets S^p, S^1 and the corresponding sets of marked indexes, the values of $\gamma_a^c(h, S^p)$ and $\gamma_a^c(h, S^1)$ depend only on the values of h on S^p and S^1 respectively. Therefore for any $1 > t \geq 2\sqrt{\frac{8\log(6)}{(1-\alpha)P_0 n}}$,

$$\begin{aligned}&\mathbb{P}_{S^p}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}|\gamma_a(h) - \gamma_a^c(h, S^p)| \geq t\right)\\ &\leq 2\mathbb{P}_{S^p,S^1}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}\left|\gamma_a^c(h, S^1) - \gamma_a^c(h, S^p)\right| \geq t/2\right)\\ &\leq 2S_{\mathcal{H}}(2n)\mathbb{P}_{S^p,S^1}^{\mathfrak{A}}\left(\left|\gamma_a^c(h, S^1) - \gamma_a^c(h, S^p)\right| \geq t/2\right)\end{aligned}$$

$$
\begin{aligned}
&\leq 2S_{\mathcal{H}}(2n)\mathbb{P}^{\mathfrak{A}}_{S^p,S^1}\left(\left|\gamma^c_a(h,S^1)-\gamma^c_a(h)\right|\geq t/4 \vee |\gamma^c_a(h,S^p)-\gamma^c_a(h)|\geq t/4\right)\\
&\leq 4S_{\mathcal{H}}(2n)\mathbb{P}^{\mathfrak{A}}_{S^p}\left(|\gamma^c_a(h,S^p)-\gamma^c_a(h)|\geq t/4\right)\\
&\leq 12S_{\mathcal{H}}(2n)\exp\left(-\frac{t^2(1-\alpha)P_an}{128}\right).
\end{aligned}
$$

Using $P_0 \leq P_1$ and Sauer's lemma, whenever $2n \geq d$ we have

$$
\mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}}|\gamma_a(h)-\gamma^c_a(h,S^p)|\geq t\right)\leq 12\left(\frac{2en}{d}\right)^d\exp\left(-\frac{t^2(1-\alpha)P_0n}{128}\right).
$$

Using inversion, we get that

$$
\mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}}|\gamma_a(h)-\gamma^c_a(h,S^p)|\geq 8\sqrt{\frac{2d\log(\frac{2en}{d})+2\log(48/\delta)}{(1-\alpha)P_0n}}\right)\leq\frac{\delta}{4}, \tag{C.10}
$$

whenever

$$
1>8\sqrt{2\frac{d\log(\frac{2en}{d})+2\log(12/\delta)}{(1-\alpha)P_0n}}\geq 2\sqrt{\frac{8\log(6)}{(1-\alpha)P_0n}}.
$$

It's easy to see that the right inequality holds whenever $\delta<1$ and $2n\geq d$. In addition, inequality (C.10) trivially holds if the left inequality is not fulfilled. Therefore, (C.10) holds whenever $2n\geq d$.

Step 3 Finally, we use (C.7) and (C.10) to proof the lemma. Recall from the proof of Lemma 3 that

$$
\begin{aligned}
|\widehat{\Gamma}^{DP}(h)-\Gamma^{DP}(h)| &= ||\gamma^c_0(h,S^p)-\gamma^c_1(h,S^p)|-|\gamma_0(h)-\gamma_1(h)||\\
&\leq |\gamma^c_0(h,S^p)-\gamma_0(h)|+|\gamma^c_1(h,S^p)-\gamma_1(h)|+\Delta_0(h)+\Delta_1(h).
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
\sup_{h\in\mathcal{H}}|\widehat{\Gamma}^{DP}(h)-\Gamma^{DP}(h)| \leq &\sup_{h\in\mathcal{H}}|\gamma^c_0(h,S^p)-\gamma_0(h)|\\
&+\sup_{h\in\mathcal{H}}|\gamma^c_1(h,S^p)-\gamma_1(h)|\\
&+\sup_{h\in\mathcal{H}}(\Delta_0(h)+\Delta_1(h)).
\end{aligned}
$$

Now, using the union bound and inequalities (C.7) and (C.10), whenever

$$
n\geq\max\left\{\frac{8\log(8/\delta)}{(1-\alpha)P_0},\frac{12\log(6/\delta)}{\alpha},\frac{d}{2}\right\}
$$

we get

$$
\begin{aligned}
&\mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}}|\widehat{\Gamma}^{DP}(h)-\Gamma^{DP}(h)|\geq\Delta^{DP}+16\sqrt{\frac{2d\log(\frac{2en}{d})+2\log(48/\delta)}{(1-\alpha)P_0n}}\right)\\
&\quad\leq\mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}}|\gamma_0(h)-\gamma^c_0(h,S^p)|\geq 8\sqrt{\frac{2d\log(\frac{2en}{d})+2\log(48/\delta)}{(1-\alpha)P_0n}}\right)
\end{aligned}
$$

$$
\begin{aligned}
&+ \mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}}|\gamma_1(h) - \gamma_1^c(h, S^p)| \geq 8\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(48/\delta)}{(1-\alpha)P_0 n}}\right)\\
&+ \mathbb{P}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}(\Delta_0(h) + \Delta_1(h)) \geq \Delta^{DP}\right)\\
&\leq \frac{\delta}{4} + \frac{\delta}{4} + \frac{\delta}{2}\\
&= \delta
\end{aligned}
$$

□

Concentration for equality of opportunity

We introduce similar notation as in Section C.2.2, but tailored to the equality of opportunity conditional probabilities.

We use the notation $C_{1a} = \sum_{i=1}^n \mathbb{1}\{i : a_i^p = a, y_i^p = 1, i \notin \mathfrak{P}\}|$ for the number of points in S^p that *were not* marked (are *clean*) and contain a point from protected group a and label $y = 1$ and $B_{1a} = \sum_{i=1}^n \mathbb{1}\{i : a_i^p = a, y_i^p = 1, i \in \mathfrak{P}\}|$ for the number of points in S^p that *were* marked (are potentially *bad*) and contain a point from protected group a and label $y = 1$. Note that $B_{10} + B_{11}$ is the total number of poisoned points for which $y = 1$ and so is at most $\mathrm{Bin}(n, \alpha)$. Similarly, denote by $C_{1a}^1(h) = \sum_{i=1}^n \mathbb{1}\{i : h(x_i^p) = 1, a_i^p = a, y_1^p = 1, i \notin \mathfrak{P}\}|$ and $B_{1a}^1(h) = \sum_{i=1}^n \mathbb{1}\{i : h(x_i^p) = 1, a_i^p = a, y_i^p = 1, i \in \mathfrak{P}\}|$.

Denote

$$
\gamma_{1a}^p(h) = \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a, y_1^p = 1\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a, y_i^p = 1\}}
$$

and

$$
\gamma_{1a}(h) = \mathbb{P}(h(X) = 1 | A = a, Y = 1),
$$

so that $\widehat{\Gamma}^{EOp}(h) = |\gamma_{10}^p(h) - \gamma_{11}^p(h)|$ and $\Gamma^{EOp}(h) = |\gamma_{10}(h) - \gamma_{11}(h)|$. Note that $\gamma_{1a}^p(h)$ is an estimate of a conditional probability *based on the corrupted data*. We now introduce the corresponding estimate that only uses the clean (but unknown) subset of the training set S^p:

$$
\gamma_a^c(h) = \frac{C_a^1(h)}{C_a(h)} = \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 1, a_i^p = a, y_i^p = 1, i \notin \mathfrak{P}\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a, y_i^p = 1, i \notin \mathfrak{P}\}}.
$$

Similarly to before, we first bound how far the corrupted estimates $\gamma_{1a}^p(h)$ of $\gamma_{1a}(h)$ are from the clean estimates $\gamma_{1a}^c(h)$, uniformly over the hypothesis space $\mathcal{H}$:

Proposition 3. *If* $n \geq \max\left\{\frac{8\log(4/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(3/\delta)}{\alpha}\right\}$, *we have*

$$
\mathbb{P}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}(|\gamma_{10}^p(h) - \gamma_{10}^c(h)| + |\gamma_{11}^p(h) - \gamma_{11}^c(h)|) \geq \frac{2\alpha}{P_{10}/3 + \alpha}\right) < \delta. \tag{C.11}
$$

Proof. Similarly to the proof of Proposition 2, we first show that certain bounds on B_{1a} and C_{1a} hold with high probability. Then we show that the supremum in (C.11) is bounded whenever these bounds hold.

Step 1 Note that $B_{10} + B_{11} \leq B_0 + B_1 \sim \mathrm{Bin}(n, \alpha)$, and so

$$\mathbb{P}^{\mathfrak{A}}\left(B_{10} + B_{11} \geq \frac{3\alpha}{2}n\right) \leq \mathbb{P}^{\mathfrak{A}}\left(B_0 + B_1 \geq \frac{3\alpha}{2}n\right) \leq e^{-\alpha n/12} \leq \frac{\delta}{3}.$$

Similarly, $C_{10} \sim \mathrm{Bin}(n, (1-\alpha)P_{1a})$ and $C_{11} \sim \mathrm{Bin}(n, (1-\alpha)P_{11})$ and so

$$\mathbb{P}^{\mathfrak{A}}\left(C_{10} \leq \frac{1-\alpha}{2}P_{10}n\right) \leq e^{-(1-\alpha)P_{10}n/8} \leq \frac{\delta}{4}$$

and

$$\mathbb{P}^{\mathfrak{A}}\left(C_{11} \leq \frac{1-\alpha}{2}P_{11}n\right) \leq e^{-(1-\alpha)P_{11}n/8} \leq \frac{\delta}{4}.$$

Now since $n \geq \max\left\{\frac{8\log(4/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(3/\delta)}{\alpha}\right\}$ and $P_{10} \leq P_{11}$

$$\mathbb{P}^{\mathfrak{A}}\left(\left(B_{10} + B_{11} \geq \frac{3\alpha}{2}n\right) \vee \left(C_{10} \leq \frac{1-\alpha}{2}P_{10}n\right) \vee \left(C_{11} \leq \frac{1-\alpha}{2}P_{11}n\right)\right) \leq \frac{\delta}{3} + \frac{\delta}{4} + \frac{\delta}{4} \tag{C.12}$$

$$< \delta, \tag{C.13}$$

Step 2 Now assume that all of $B_{10} + B_{11} < \frac{3\alpha}{2}n, C_{10} > \frac{1-\alpha}{2}P_{10}n, C_{11} > \frac{1-\alpha}{2}P_{11}n$ hold.

Consider an arbitrary, fixed $h \in \mathcal{H}$. Since h is fixed, we drop the dependence on h from the notation for the rest of the proof and write $\gamma^p_{1a} = \gamma^p_{1a}(h), C^1_{1a} = C^1_{1a}(h)$ etc.

We now prove that for both $a \in \{0, 1\}$

$$\Delta_{1a} \coloneqq |\gamma^p_a - \gamma^c_a| \leq \frac{B_{1a}}{C_{1a} + B_{1a}}. \tag{C.14}$$

For each $a \in \{0, 1\}$, this can be shown as follows. First, if $\sum_{i=1}^n \mathbb{1}\{a^p_i = a, y^p_i = 1\} = B_{1a} + C_{1a} = 0$, then both $\gamma^p_{1a}(h)$ and $\gamma^c_{1a}(h)$ are equal to 0, because of the convention that $\frac{0}{0} = 0$. In addition, $B_{1a} = C_{1a} = 0$. Therefore, inequality (C.14) trivially holds.

Similarly, if $B_{1a} = 0$ and $C_{1a} > 0$, then $\gamma^p_{1a}(h) = \gamma^c_{1a}(h)$ and so $\Delta_{1a} = 0$ and (C.14) holds.

Assume now that $B_{1a} > 0$. Note that if $C_{1a} = \sum_{i=1}^n \mathbb{1}\{a^p_i = a, y^p_i = 1, i \notin \mathfrak{P}\} = 0$, then

$$\Delta_{1a} = |\gamma^p_{1a}(h) - \gamma^c_{1a}(h)| = \left|\frac{B^1_{1a}}{B_{1a}} - 0\right| = \frac{B^1_{1a}}{B_{1a}} = \frac{B^1_{1a}}{B_{1a} + C_{1a}} \leq \frac{B_{1a}}{C_{1a} + B_{1a}}.$$

Finally, assume that both $C_{1a} > 0$ and $B_{1a} > 0$. Note that under any realization of the randomness of the data sampling and the adversary, for any $a \in \{0, 1\}$

$$\begin{aligned}\gamma^p_{1a} &= \frac{\sum_{i=1}^n \mathbb{1}\{h(x^p_i) = 1, a^p_i = a, y^p_i = 1, i \notin \mathfrak{P}\} + \sum_{i=1}^n \mathbb{1}\{h(x^p_i) = 1, a^p_i = a, y^p_i = 1, i \in \mathfrak{P}\}}{\sum_{i=1}^n \mathbb{1}\{a^p_i = a, y^p_i = 1, i \notin \mathfrak{P}\} + \sum_{i=1}^n \mathbb{1}\{a^p_i = a, y^p_i = 1, i \in \mathfrak{P}\}} \\ &= \frac{C^1_{1a} + B^1_{1a}}{C_{1a} + B_{1a}}.\end{aligned}$$

Next we bound how far this quantity is from the clean estimator $\frac{C^1_{1a}}{C_{1a}}$

$$\Delta_{1a} \coloneqq |\gamma^p_{1a} - \gamma^c_{1a}| = \left|\frac{C^1_{1a} + B^1_{1a}}{C_{1a} + B_{1a}} - \gamma^c_{1a}\right| = \frac{B_{1a}}{C_{1a} + B_{1a}}\left|\gamma^c_{1a} - \frac{B^1_{1a}}{B_{1a}}\right| \leq \frac{B_{1a}}{C_{1a} + B_{1a}}.$$

Now since $B_{10} + B_{11} < \frac{3\alpha}{2}n, C_{10} > \frac{1-\alpha}{2}P_{10}n, C_{11} > \frac{1-\alpha}{2}P_{11}n$ hold, we get

$$\begin{aligned}
\Delta_{10} + \Delta_{11} &\leq \frac{B_{10}}{C_{10} + B_{10}} + \frac{B_{11}}{C_{11} + B_{11}} \\
&< \frac{B_{10}}{\frac{1-\alpha}{2}P_{10}n + B_{10}} + \frac{B_{11}}{\frac{1-\alpha}{2}P_{11}n + B_{11}} \\
&\leq \frac{B_{10}}{\frac{1-\alpha}{2}P_{10}n + B_{10}} + \frac{B_{11}}{\frac{1-\alpha}{2}P_{10}n + B_{11}} \\
&= \frac{B_{10}}{\frac{1-\alpha}{2}P_{10}n + B_{10}} + 1 - \frac{\frac{1-\alpha}{2}P_{10}n}{\frac{1-\alpha}{2}P_{10}n - B_{10} + (B_{10} + B_{11})} \\
&< \frac{B_{10}}{\frac{1-\alpha}{2}P_{10}n + B_{10}} + 1 - \frac{\frac{1-\alpha}{2}P_{10}n}{\frac{1-\alpha}{2}P_{10}n - B_{10} + \frac{3\alpha}{2}n} \\
&= 2 - (1-\alpha)P_{10}n\left(\frac{1}{(1-\alpha)P_{10}n + 2B_{10}} + \frac{1}{(1-\alpha)P_{10}n + 3\alpha n - 2B_{10}}\right)
\end{aligned}$$

The same argument as in Proposition 2 shows that this is maximized at $B_{10} = \frac{3\alpha}{4}n$ and so

$$\begin{aligned}
\Delta_{10} + \Delta_{11} &\leq \frac{B_{10}}{C_{10} + B_{10}} + \frac{B_{11}}{C_{11} + B_{11}} \qquad \text{(C.15)} \\
&< 2 - (1-\alpha)P_{10}n\left(\frac{1}{(1-\alpha)P_{10}n + \frac{3\alpha}{2}n} + \frac{1}{(1-\alpha)P_{10}n + 3\alpha n - \frac{3\alpha}{2}n}\right) \\
&\leq \frac{2\alpha}{P_{10}/3 + \alpha}.
\end{aligned}$$

Since this holds for any arbitrary hypothesis $h \in \mathcal{H}$, the result follows. □

Denote the irreducible error term for equality of opportunity by $\Delta^{EOp} = \frac{2\alpha}{P_{10}/3+\alpha}$. We then have the following bound for a fixed $h \in \mathcal{H}$:

Lemma 5. *Let $h \in \mathcal{H}$ be a fixed hypothesis and $D \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be a fixed distribution. Denote $P_{1a} = \mathbb{P}(A = a, Y = 1)$ for $a \in \{0, 1\}$. Let $\mathfrak{A}$ be any malicious adversary and denote by $\mathbb{P}^{\mathfrak{A}}$ the probability distribution of the poisoned data S^p, under the random sampling of the clean data, the marked points and the randomness of the adversary. Then for any $n \geq \max\left\{\frac{8\log(8/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(6/\delta)}{\alpha}\right\}$ and $\delta \in (0, 1)$*

$$\mathbb{P}^{\mathfrak{A}}\left(\left|\widehat{\Gamma}^{EOp}(h) - \Gamma^{EOp}(h)\right| \leq \Delta^{EOp} + 2\sqrt{\frac{\log(16/\delta)}{n(1-\alpha)P_{10}}}\right) \geq 1 - \delta \qquad \text{(C.16)}$$

Proof. The proof is exactly the same as the one of Lemma 3 , but with conditioning on $S^c_{1a} = \{i : a^p_i = a, y^p_i = 1, i \notin \mathfrak{P}\}$ (the set of indexes of the poisoned data for which the protected group is a, the label is 1 and the corresponding point was not marked for the adversary) instead. □

The same argument as in Lemma 4 gives a uniform bound over the whole hypothesis space, provided that $\mathcal{H}$ has a finite VC-dimension $d := VC(\mathcal{H})$:

Lemma 6. *Under the setup of Lemma 5, assume additionally that $\mathcal{H}$ has a finite VC-dimension d. Then for any $n \geq \max\left\{\frac{8\log(8/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(6/\delta)}{\alpha}, \frac{d}{2}\right\}$ and $\delta \in (0,1)$*

$$\mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}}|\widehat{\Gamma}^{EOp}(h) - \Gamma^{EOp}(h)| \leq \Delta^{EOp} + 16\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(48/\delta)}{(1-\alpha)P_{10}n}}\right) \geq 1-\delta. \tag{C.17}$$

Finally, we prove multiplicative bounds and claims in the case when $\mathbb{P}(h(X)=1|A=0,Y=1) = \mathbb{P}(h(X)=1|A=1,Y=1) = 1$ (which holds for example when $h(X)=Y$ almost surely). These will come in useful for proving the component-wise upper bound with fast rates.

We will be interested in the estimate

$$\bar{\gamma}^p_{1a}(h) = \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p)=0, a_i^p = a, y_1^p = 1\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a, y_i^p = 1\}}$$

of $\bar{\gamma}_{1a}(h) = \mathbb{P}(h(X)=0|A=a,Y=1)$. Again, we also introduce the corresponding clean data estimate $C^0_{1a}(h) \coloneqq \sum_{i=1}^n \mathbb{1}\{i : h(x_i^p)=0, a_i^p=a, y_1^p=1, i\notin\mathfrak{P}\}$ and

$$\bar{\gamma}^c_{1a}(h) = \frac{C^0_{1a}(h)}{C_{1a}} = \frac{\sum_{i=1}^n \mathbb{1}\{i : h(x_i^p)=0, a_i^p=a, y_1^p=1, i\notin\mathfrak{P}\}}{\sum_{i=1}^n \mathbb{1}\{i : a_i^p=a, y_1^p=1, i\notin\mathfrak{P}\}}.$$

Denote also

$$\bar{\Delta}_{1a}(h) \coloneqq |\bar{\gamma}^p_{1a}(h) - \gamma^c_{1a}(h)|,$$

We only show non-uniform bounds for a fixed $h\in\mathcal{H}$ here, so we omit the dependence of these quantities on h. We have:

Lemma 7. *Let $\mathbb{P}\in\mathcal{P}(\mathcal{X}\times\mathcal{A}\times\mathcal{Y})$ be a fixed distribution and let $h\in\mathcal{H}$ be a fixed classifier. Denote $P_{1a} = \mathbb{P}(A=a,Y=1)$ for $a\in\{0,1\}$. Let $\mathfrak{A}$ be any malicious adversary and denote by $\mathbb{P}^{\mathfrak{A}}$ the probability distribution of the poisoned data S^p, under the random sampling of the clean data, the marked points and the randomness of the adversary. Then:*
(a) For any $n>0$ and any $\eta,\delta\in(0,1)$

$$\mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}^p_{1a} \geq (1+\eta)\bar{\gamma}_{1a} + \bar{\Delta}_{1a}\right) \leq \exp\left(-\frac{(1-\alpha)P_{1a}n}{8}\right) + \exp\left(-\frac{1}{6}\eta^2(1-\alpha)P_{1a}\bar{\gamma}_{1a}n\right). \tag{C.18}$$

and

$$\mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}^p_{1a} \leq (1-\eta)\bar{\gamma}_{1a} - \bar{\Delta}_{1a}\right) \leq \exp\left(-\frac{(1-\alpha)P_{1a}n}{8}\right) + \exp\left(-\frac{1}{4}\eta^2(1-\alpha)P_{1a}\bar{\gamma}_{1a}n\right). \tag{C.19}$$

(b) Assume further that $\mathbb{P}(h(X)=0|A=0,Y=1) = \mathbb{P}(h(X)=0|A=1,Y=1) = 0$. Then for any $\delta\in(0,1)$ and $n\geq\max\left\{\frac{8\log(4/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(3/\delta)}{\alpha}\right\}$

$$\mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}^p_{10} + \bar{\gamma}^p_{11} \geq \Delta^{EOp}\right) \leq \delta \tag{C.20}$$

Proof. Let $S_{1a}^c = \{i : a_i^p = a, y_i^p = 1, i \notin \mathfrak{P}\}$. For any $a \in \{0,1\}$ we have

$$\begin{aligned}
\mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}_{1a}^p \geq (1+\eta)\bar{\gamma}_{1a} + \bar{\Delta}_{1a}\right) &= \sum_{S_{1a}^c} \mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}_{1a}^p \geq (1+\eta)\bar{\gamma}_{1a} + \bar{\Delta}_{1a}\middle| S_a^c\right)\mathbb{P}(S_a^c) \\
&\leq \mathbb{P}^{\mathfrak{A}}\left(C_{1a} \leq \frac{(1-\alpha)}{2}P_{1a}n\right) \\
&+ \sum_{S_{1a}^c : C_{1a} \geq \frac{(1-\alpha)}{2}P_{1a}n} \mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}_{1a}^p \geq (1+\eta)\bar{\gamma}_{1a} + \bar{\Delta}_{1a}\middle| S_{1a}^c\right)\mathbb{P}^{\mathfrak{A}}(S_{1a}^c) \\
&\leq \mathbb{P}^{\mathfrak{A}}\left(C_{1a} \leq \frac{(1-\alpha)}{2}P_{1a}n\right) \\
&+ \sum_{S_{1a}^c : C_{1a} \geq \frac{(1-\alpha)}{2}P_{1a}n} \mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}_{1a}^p - \frac{C_{1a}^1}{C_{1a}} + \frac{C_{1a}^1}{C_{1a}} \geq \right. \\
&\qquad \left. (1+\eta)\bar{\gamma}_{1a} + \bar{\Delta}_{1a}\middle| S_{1a}^c\right)\mathbb{P}^{\mathfrak{A}}(S_{1a}^c) \\
&\leq \mathbb{P}^{\mathfrak{A}}\left(C_{1a} \leq \frac{(1-\alpha)}{2}P_{1a}n\right) \\
&+ \sum_{S_{1a}^c : C_{1a} \geq \frac{(1-\alpha)}{2}P_{1a}n} \mathbb{P}^{\mathfrak{A}}\left(\frac{C_{1a}^1}{C_{1a}} \geq (1+\eta)\bar{\gamma}_{1a}\middle| S_{1a}^c\right)\mathbb{P}^{\mathfrak{A}}(S_{1a}^c) \\
&\leq \exp\left(-\frac{(1-\alpha)P_{1a}n}{8}\right) \\
&+ \sum_{S_a^p : C_{1a} \geq \frac{(1-\alpha)}{2}P_{1a}n} \exp\left(-\frac{\eta^2 C_{1a}\bar{\gamma}_{1a}}{3}\right)\mathbb{P}^{\mathfrak{A}}(S_{1a}^c) \\
&\leq \exp\left(-\frac{(1-\alpha)P_{1a}n}{8}\right) + \exp\left(-\frac{1}{6}\eta^2(1-\alpha)P_{1a}\bar{\gamma}_{1a}n\right).
\end{aligned}$$

A similar argument, with the other direction of the Chernoff bounds, gives the other bound.

(b) Similarly to the argument in the proof of Proposition 3

$$\bar{\Delta}_{1a} = \left|\bar{\gamma}_{1a}^p - \frac{C_{1a}^0}{C_{1a}}\right| \leq \frac{B_{1a}}{C_{1a} + B_{1a}}. \tag{C.21}$$

Using the inequalities (C.12) and (C.15),

$$\mathbb{P}^{\mathfrak{A}}\left(\bar{\Delta}_{10} + \bar{\Delta}_{11} \geq \frac{2\alpha}{P_{10}/3 + \alpha}\right) \leq \mathbb{P}^{\mathfrak{A}}\left(\frac{B_{10}}{C_{10}+B_{10}} + \frac{B_{11}}{C_{11}+B_{11}} \geq \frac{2\alpha}{P_{10}/3+\alpha}\right) < \delta. \tag{C.22}$$

Since also

$$\mathbb{P}^{\mathfrak{A}}\left(\frac{C_{1a}^0}{C_{1a}} > 0\right) = \sum_{S_{1a}^c} \mathbb{P}^{\mathfrak{A}}\left(\frac{C_{1a}^0}{C_{1a}} > 0\middle| S_{1a}^c\right)\mathbb{P}^{\mathfrak{A}}\left(S_{1a}^c\right) = \sum_{S_{1a}^c} \mathbb{P}\left(\mathrm{Bin}(|S_{1a}^c|, 0) > 0\right)\mathbb{P}^{\mathfrak{A}}(S_{1a}^c) = 0,$$

we have that $0 \leq \bar{\gamma}_{1a}^p = \bar{\Delta}_{1a}$ almost surely, for both $a \in \{0,1\}$. Therefore, $0 < \bar{\gamma}_{10}^p + \bar{\gamma}_{11}^p = \bar{\Delta}_{10} + \bar{\Delta}_{11}$ and the result follows. □

C.2.3 Upper bound theorems - proofs

We are now ready to present the proofs of the upper bound results from the main body of the paper.

Upper bounds on the λ-weighted objective

First we prove the bounds for the λ-weighted objective.

Bound for demographic parity Let $\lambda \geq 0$ be fixed. Recall our notation for the λ-weighted objective:

$$L_\lambda^{DP}(h) = \mathcal{R}(h) + \lambda \Gamma^{DP}(h).$$

Suppose that a learner $\mathcal{L}_\lambda^{DP} : \cup_{n=1}^{\infty}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})^n \to \mathcal{H}$ is such that

$$\mathcal{L}^{DP}(S^p) \in \operatorname*{argmin}_{h \in \mathcal{H}}(\widehat{R}^p(h) + \lambda \widehat{\Gamma}^{DP}(h)) \quad \text{for all } S^p.$$

That is, $\mathcal{L}_\lambda^{DP}$ always returns a minimizer of the λ-weighted empirical objective. Then we have the following:

Theorem 12. *Let $\mathcal{H}$ be any hypothesis space with $d = VC(\mathcal{H}) < \infty$. Let $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be a fixed distribution and $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Denote by $\mathbb{P}^{\mathfrak{A}}$ the probability distribution of the poisoned data S^p, under the random sampling of the clean data, the marked points and the randomness of the adversary. Then for any $\delta \in (0,1)$ and $n \geq \max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_0}, \frac{12\log(12/\delta)}{\alpha}, \frac{d}{2}\right\}$, we have*

$$\mathbb{P}^{\mathfrak{A}}\left(L_\lambda^{DP}(\mathcal{L}_\lambda^{DP}(S^p)) \leq \min_{h \in \mathcal{H}} L_\lambda^{DP}(h) + \Delta_\lambda^{DP}\right) > 1 - \delta,$$

where[4]

$$\Delta_\lambda^{DP} = 3\alpha + 2\lambda\Delta^{DP} + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}} + \lambda\sqrt{\frac{d}{P_0 n}}\right)$$

and

$$\Delta^{DP} = \frac{2\alpha}{P_0/3 + \alpha} = \mathcal{O}\left(\frac{\alpha}{P_0}\right).$$

Proof. By the standard concentrations results for the $0/1$ loss (see, for example, Chapter 28.1 in [SSBD14])

$$\mathbb{P}\left(\sup_{h \in \mathcal{H}} |\widehat{\mathcal{R}}^c(h) - \mathcal{R}(h)| > 2\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}\right) \leq \frac{\delta}{4},$$

where $\widehat{\mathcal{R}}^c(h) = \frac{1}{n}\sum_{i=1}^n \mathbb{1}(h(x_i^c) \neq y_i^c)$ is the loss of h on the clean data. Since the total number of poisoned points $|\mathfrak{P}| \sim \mathrm{Bin}(n, \alpha)$ and since $n > \frac{12\log(4/\delta)}{\alpha}$

$$\mathbb{P}^{\mathfrak{A}}\left(\sup_{h \in \mathcal{H}} |\widehat{\mathcal{R}}^c(h) - \widehat{\mathcal{R}}^p(h)| > \frac{3\alpha}{2}\right) \leq \mathbb{P}^{\mathfrak{A}}\left(|\mathfrak{P}| \geq \frac{3\alpha}{2}n\right) \leq e^{-\alpha n/12} \leq \frac{\delta}{4}.$$

[4] the $\widetilde{\mathcal{O}}$-notation hides constant and logarithmic factors

Since $\sup_{h\in\mathcal{H}}|\widehat{\mathcal{R}}^p(h)-\mathcal{R}(h)| \leq \sup_{h\in\mathcal{H}}|\widehat{\mathcal{R}}^p(h)-\widehat{\mathcal{R}}^c(h)|+\sup_{h\in\mathcal{H}}|\widehat{\mathcal{R}}^c(h)-\mathcal{R}(h)|$, we obtain

$$\mathbb{P}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}|\widehat{\mathcal{R}}^p(h)-\mathcal{R}(h)| > \frac{3\alpha}{2}+2\sqrt{\frac{8d\log(\frac{en}{d})+2\log(16/\delta)}{n}}\right)\leq\frac{\delta}{2}. \tag{C.23}$$

In addition, Lemma 3 implies that

$$\mathbb{P}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}}\left|\widehat{\Gamma}^{DP}(h)-\Gamma^{DP}(h)\right| > \Delta^{DP}+16\sqrt{\frac{2d\log(\frac{2en}{d})+2\log(96/\delta)}{(1-\alpha)P_0 n}}\right)\leq\frac{\delta}{2}. \tag{C.24}$$

Now let $h_\lambda = \operatorname{argmin}_{h\in\mathcal{H}}(\widehat{R}^p(h)+\lambda\widehat{\Gamma}^{DP}(h))$ and let

$$\begin{aligned}\Delta_\lambda^{DP} &= 3\alpha+4\sqrt{\frac{8d\log(\frac{en}{d})+2\log(16/\delta)}{n}}+2\lambda\Delta^{DP}+32\lambda\sqrt{\frac{2d\log(\frac{2en}{d})+2\log(96/\delta)}{(1-\alpha)P_0 n}}\\ &= \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}}+\lambda\sqrt{\frac{d}{P_0 n}}\right).\end{aligned}$$

Then, using (C.23) and (C.26), we have that with probability at least $1-\delta$

$$\begin{aligned}L_\lambda^{DP}(\mathcal{L}_\lambda^{DP}(S^p)) &= \mathcal{R}(\mathcal{L}_\lambda^{DP}(S^p))+\lambda\Gamma^{DP}(\mathcal{L}_\lambda^{DP}(S^p))\\ &\leq \widehat{\mathcal{R}}^p(\mathcal{L}_\lambda^{DP}(S^p))+\lambda\widehat{\Gamma}^{DP}(\mathcal{L}_\lambda^{DP}(S^p))+\frac{1}{2}\Delta_\lambda^{DP}\\ &= \min_{h\in\mathcal{H}}\left(\widehat{\mathcal{R}}^p(h)+\lambda\widehat{\Gamma}^{DP}(h)\right)+\frac{1}{2}\Delta_\lambda^{DP}\\ &\leq \min_{h\in\mathcal{H}} L_\lambda^{DP}(h)+\Delta_\lambda^{DP}.\end{aligned}$$

□

Bound for equality of opportunity We now show a similar result for the weighted-objective with the equality of opportunity deviation measure

$$L_\lambda^{EOp}(h) = \mathcal{R}(h)+\lambda\Gamma^{EOp}(h).$$

Let $\mathcal{L}_\lambda^{EOp}: \cup_{n=1}^{\infty}(\mathcal{X}\times\mathcal{A}\times\mathcal{Y})^n\to\mathcal{H}$ be such that

$$\mathcal{L}_\lambda^{EOp}(S^p)\in\operatorname*{argmin}_{h\in\mathcal{H}}(\widehat{R}^p(h)+\lambda\widehat{\Gamma}^{EOp}(h)),\quad \text{for all } S^p.$$

That is, $\mathcal{L}_\lambda^{EOp}$ always returns a minimizer of the λ-weighted empirical objective. Then:

Theorem 13. *Let $\mathcal{H}$ be any hypothesis space with $d=VC(\mathcal{H})<\infty$. Let $\mathbb{P}\in\mathcal{P}(\mathcal{X}\times\mathcal{A}\times\mathcal{Y})$ be a fixed distribution and $\mathfrak{A}$ be any malicious adversary of power $\alpha<0.5$. Denote by $\mathbb{P}^{\mathfrak{A}}$ the probability distribution of the poisoned data S^p, under the random sampling of the clean data, the marked points and the randomness of the adversary. Then for any $\delta\in(0,1)$ and $n\geq\max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_{10}},\frac{12\log(12/\delta)}{\alpha},\frac{d}{2}\right\}$, we have*

$$\mathbb{P}^{\mathfrak{A}}\left(L_\lambda^{EOp}(\mathcal{L}_\lambda^{EOp}(S^p))\leq\min_{h\in\mathcal{H}}L_\lambda^{EOp}(h)+\Delta_\lambda^{EOp}\right)\leq\delta,$$

where

$$\Delta_\lambda^{EOp} = 3\alpha + 2\lambda\Delta^{EOp} + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}} + \lambda\sqrt{\frac{d}{P_{10}n}}\right)$$

and

$$\Delta^{EOp} = \frac{2\alpha}{P_{10}/3 + \alpha} = \mathcal{O}\left(\frac{\alpha}{P_{10}}\right).$$

Proof. Similarly to the proof of Theorem 12, we combine

$$\mathbb{P}^{\mathfrak{A}}\left(\sup_{h\in\mathcal{H}} |\widehat{\mathcal{R}}^p(h) - \mathcal{R}(h)| > \frac{3\alpha}{2} + 2\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}\right) \leq \frac{\delta}{2}. \tag{C.25}$$

and Lemma 5

$$\mathbb{P}^{\mathfrak{A}}_{S^p}\left(\sup_{h\in\mathcal{H}} |\widehat{\Gamma}^{EOp}(h) - \Gamma^{EOp}(h)| > \Delta^{EOp} + 16\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_{10}n}}\right) \leq \frac{\delta}{2} \tag{C.26}$$

Now let $h_\lambda = \operatorname{argmin}_{h\in\mathcal{H}}(\widehat{R}^p(h) + \lambda\widehat{\Gamma}^{EOp}(h))$ and let

$$\Delta_\lambda^{EOp} = 3\alpha + 4\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}} + 2\lambda\Delta^{EOp} + 32\lambda\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_{10}n}}.$$

Then we have that with probability at least $1-\delta$

$$\begin{aligned}
L_\lambda^{EOp}(\mathcal{L}_\lambda^{EOp}(S^p)) &= \mathcal{R}(\mathcal{L}_\lambda^{EOp}(S^p)) + \lambda\Gamma^{EOp}(\mathcal{L}_\lambda^{EOp}(S^p)) \\
&\leq \widehat{\mathcal{R}}^p(\mathcal{L}_\lambda^{EOp}(S^p)) + \lambda\widehat{\Gamma}^{EOp}(\mathcal{L}_\lambda^{DP}(S^p)) + \frac{1}{2}\Delta_\lambda^{EOp} \\
&= \min_{h\in\mathcal{H}}\left(\widehat{\mathcal{R}}^p(h) + \lambda\widehat{\Gamma}^{EOp}(h)\right) + \frac{1}{2}\Delta_\lambda^{EOp} \\
&\leq \min_{h\in\mathcal{H}} L_\lambda^{EOp}(h) + \Delta_\lambda^{EOp}.
\end{aligned}$$

□

Component-wise upper bounds

We now prove the component-wise upper bound results.

Bound for demographic parity Recall our notation $\widehat{h}^r \in \operatorname{argmin}_{h\in\mathcal{H}} \widehat{\mathcal{R}}^p(h)$ and $\widehat{h}^{DP} \in \operatorname{argmin}_{h\in\mathcal{H}} \widehat{\Gamma}^{DP}(h)$. Further, we define the sets

$$\mathcal{H}_1 = \left\{h \in \mathcal{H} : \widehat{\mathcal{R}}^p(h) - \widehat{\mathcal{R}}^p(\widehat{h}^r) \leq 3\alpha + 4\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}\right\}$$

$$\mathcal{H}_2 = \left\{h \in \mathcal{H} : \widehat{\Gamma}^{DP}(h) - \widehat{\Gamma}^{DP}(\widehat{h}^{DP}) \leq 2\Delta^{DP} + 32\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_0 n}}\right\}.$$

That is, $\mathcal{H}_1$ and $\mathcal{H}_2$ are the sets of classifiers that are not far from optimal on the train data, in terms of their risk and their fairness respectively. Define the *component-wise learner*:

$$\mathcal{L}_{cw}^{DP}(S^p) = \begin{cases} \text{any } h \in \mathcal{H}_1 \cap \mathcal{H}_2, & \text{if } \mathcal{H}_1 \cap \mathcal{H}_2 \neq \emptyset \\ \text{any } h \in \mathcal{H}, & \text{otherwise,} \end{cases}$$

that returns a classifier that is good in both metrics, if such exists, or an arbitrary classifier otherwise. Then we have the following:

Theorem 14. *Let $\mathcal{H}$ be any hypothesis space with $d = VC(\mathcal{H}) < \infty$. Let $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be a fixed distribution and let $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Suppose that there exists a hypothesis $h^* \in \mathcal{H}$, such that $\mathfrak{V}(h^*) \preceq \mathfrak{V}(h)$ for all $h \in \mathcal{H}$. Then for any $\delta \in (0,1)$ and $n \geq \max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_0}, \frac{12\log(12/\delta)}{\alpha}, \frac{d}{2}\right\}$, with probability at least $1-\delta$:*

$$\mathbf{L}^{DP}(\mathcal{L}_{cw}^{DP}(S^p)) \preceq \left(6\alpha + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}}\right), 4\Delta^{DP} + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{P_0 n}}\right)\right).$$

Proof. From the proof of Theorem 12, we have that with probability at least $1-\delta$, both of the following hold:

$$\sup_{h\in\mathcal{H}} |\widehat{\mathcal{R}}^p(h) - \mathcal{R}(h)| \leq \frac{3\alpha}{2} + 2\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}},$$

$$\sup_{h\in\mathcal{H}} \left|\widehat{\Gamma}^{DP}(h) - \Gamma^{DP}(h)\right| \leq \Delta^{DP} + 16\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_0 n}}.$$

We show that under this event, $\mathcal{H}_1 \cap \mathcal{H}_2 \neq \emptyset$ and for any $h \in \mathcal{H}_1 \cap \mathcal{H}_2$,

$$\mathbf{L}^{DP}(h) \preceq \left(6\alpha + 8\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}, 4\Delta^{DP} + 64\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_0 n}}\right),$$

from which the result follows. Note that

$$\begin{aligned} \widehat{\mathcal{R}}^p(h^*) &\leq \mathcal{R}(h^*) + \frac{3\alpha}{2} + 2\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}} \\ &\leq \mathcal{R}(\widehat{h}^r) + \frac{3\alpha}{2} + 2\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}} \\ &\leq \widehat{\mathcal{R}}^p(\widehat{h}^r) + 3\alpha + 4\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}} \end{aligned}$$

and similarly

$$\widehat{\Gamma}^{DP}(h^*) \leq \widehat{\Gamma}^{DP}(\widehat{h}^r) + 2\Delta^{DP} + 32\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_0 n}}$$

Therefore, $h^* \in \mathcal{H}_1 \cap \mathcal{H}_2$ and so $\mathcal{H}_1 \cap \mathcal{H}_2 \neq \emptyset$.

Now take any $h \in \mathcal{H}_1 \cap \mathcal{H}_2$. We have that

$$\mathcal{R}(h) \leq \widehat{\mathcal{R}}^p(h) + \frac{3\alpha}{2} + 2\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}$$

$$\leq \widehat{\mathcal{R}}^p(\widehat{h}^r) + 3\frac{3\alpha}{2} + 6\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}$$
$$\leq \widehat{\mathcal{R}}^p(h^*) + 3\frac{3\alpha}{2} + 6\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}$$
$$\leq \mathcal{R}(h^*) + 6\alpha + 8\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}.$$

Similarly,

$$\Gamma^{DP}(h) \leq \Gamma^{DP}(h^*) + 4\Delta^{DP} + 64\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_0 n}}$$

and the result follows. □

Bound for equality of opportunity Similarly, let $\widehat{h}^{EOp} \in \operatorname{argmin}_{h\in\mathcal{H}} \widehat{\Gamma}^{EOp}(h)$. Further, we define the set

$$\mathcal{H}_3 = \left\{ h \in \mathcal{H} : \widehat{\Gamma}^{EOp}(h) - \widehat{\Gamma}^{EOp}(\widehat{h}^{EOp}) \leq 2\Delta^{EOp} + 32\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_{10} n}} \right\}.$$

That is, $\mathcal{H}_3$ is the set of classifiers that are not far from optimal on the train data, in terms of equality of opportunity fairness. Now define the *component-wise learner* for equality of opportunity:

$$\mathcal{L}^{EOp}_{cw}(S^p) = \begin{cases} \text{any } h \in \mathcal{H}_1 \cap \mathcal{H}_3, & \text{if } \mathcal{H}_1 \cap \mathcal{H}_3 \neq \emptyset \\ \text{any } h \in \mathcal{H}, & \text{otherwise,} \end{cases}$$

that returns a classifier that is good in both metrics, if such exists, or an arbitrary classifier otherwise. Then we have the following:

Theorem 15. *Let $\mathcal{H}$ be any hypothesis space with $d = VC(\mathcal{H}) < \infty$. Let $\mathbb{P} \in \mathcal{P}(\mathcal{X}\times\mathcal{A}\times\mathcal{Y})$ be a fixed distribution and let $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Suppose that there exists a hypothesis $h^* \in \mathcal{H}$, such that $\mathfrak{V}(h^*) \preceq \mathfrak{V}(h)$ for all $h \in \mathcal{H}$. Then for any $\delta \in (0,1)$ and $n \geq \max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(12/\delta)}{\alpha}, \frac{d}{2}\right\}$, with probability at least $1-\delta$*

$$\mathbf{L}^{EOp}(\mathcal{L}^{EOp}_{cw}(S^p)) \preceq \left(6\alpha + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{n}}\right), 4\Delta^{EOp} + \widetilde{\mathcal{O}}\left(\sqrt{\frac{d}{P_{10}n}}\right)\right).$$

Proof. From the proof of Theorem 13 we have that with probability at least $1-\delta$:

$$\sup_{h\in\mathcal{H}} |\widehat{\mathcal{R}}^p(h) - \mathcal{R}(h)| \leq \frac{3\alpha}{2} + 2\sqrt{\frac{8d\log(\frac{en}{d}) + 2\log(16/\delta)}{n}}$$

and Lemma 5

$$\sup_{h\in\mathcal{H}} |\widehat{\Gamma}^{EOp}(h) - \Gamma^{EOp}(h)| \leq \Delta^{EOp} + 16\sqrt{\frac{2d\log(\frac{2en}{d}) + 2\log(96/\delta)}{(1-\alpha)P_{10}n}}.$$

The proof proceeds in an identical manner to that of Theorem 14. □

Upper bound with fast rates Recall our notation:

$$\bar{\gamma}^p_{1a}(h) = \frac{\sum_{i=1}^n \mathbb{1}\{h(x_i^p) = 0, a_i^p = a, y_1^p = 1\}}{\sum_{i=1}^n \mathbb{1}\{a_i^p = a, y_i^p = 1\}} \tag{C.27}$$

as the empirical estimate of $\bar{\gamma}_{1a}(h) \coloneqq \mathbb{P}(h(X) = 0 | A = a, Y = 1) = 0$ for $a \in \{0, 1\}$. Given a (corrupted) training set S^p, denote by

$$\mathcal{H}^* := \left\{ h \in \mathcal{H} \middle| \max_a \bar{\gamma}^p_{1a}(h) \leq \Delta^{EOp} \wedge \widehat{\mathcal{R}}^p(h) \leq \frac{3\alpha}{2} \right\} \tag{C.28}$$

the set of all classifiers that have a small loss and small values of $\bar{\gamma}^p_{1a}$ for both $a \in \{0, 1\}$ on S^p. Consider the learner $\mathcal{L}^{fast}$ defined by

$$\mathcal{L}^{fast}(S^p) = \begin{cases} \text{any } h \in \mathcal{H}^*, & \text{if } \mathcal{H}^* \neq \emptyset \\ \text{any } h \in \mathcal{H}, & \text{otherwise.} \end{cases} \tag{C.29}$$

We then have the following:

Theorem 16. *Let $\mathcal{H}$ be finite and $\mathbb{P} \in \mathcal{P}(\mathcal{X} \times \mathcal{A} \times \mathcal{Y})$ be such that for some $h^* \in \mathcal{H}$, $\mathbb{P}(h^*(X) = Y) = 1$. Denote by $P_{1a} = \mathbb{P}(Y = 1, A = a)$ for $a \in \{0, 1\}$. Let $\mathfrak{A}$ be any malicious adversary of power $\alpha < 0.5$. Then for any $\delta, \eta \in (0, 1)$ and any*

$$\begin{aligned} n &\geq \max \left\{ \frac{8 \log(16|\mathcal{H}|/\delta)}{(1-\alpha)P_{10}}, \frac{12 \log(12/\delta)}{\alpha}, \frac{2 \log(8|\mathcal{H}|/\delta)}{3\eta^2 \alpha}, \frac{2 \log(\frac{16|\mathcal{H}|}{\delta})}{3\eta^2(1-\alpha)P_{10}\alpha} \right\} \\ &= \Omega\left(\frac{\log(|\mathcal{H}|/\delta)}{\eta^2 P_{10} \alpha} \right) \end{aligned}$$

with probability at least $1 - \delta$

$$\mathbf{L}^{EOp}(\mathcal{L}^{fast}) \preceq \left(\frac{3\alpha}{1-\eta}, \frac{2\Delta^{EOp}}{1-\eta} \right).$$

Proof. Throughout the proof we will drop the dependence of $\mathcal{H}^*$ (and other subsets of $\mathcal{H}$) of the data S^p. We will be interested in the probability of certain events involving $\mathcal{H}^*$ under all randomness in the generation of S^p: the random sampling of the clean data, the marked point and the adversary (denoted by $\mathbb{P}^{\mathfrak{A}}$ as elsewhere).

Step 1 First note that by Lemma 7(b), whenever $n \geq \max\left\{ \frac{8 \log(16/\delta)}{(1-\alpha)P_{10}}, \frac{12 \log(12/\delta)}{\alpha} \right\}$

$$\mathbb{P}^{\mathfrak{A}}\left(\left(\bar{\gamma}^p_{10}(h^*) > \Delta^{EOp} \right) \vee \left(\bar{\gamma}^p_{11}(h^*) > \Delta^{EOp} \right) \right) \leq \mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}^p_{10}(h^*) + \bar{\gamma}^p_{11}(h^*) > \Delta^{EOp} \right) \leq \frac{\delta}{4}$$

In addition, since $|\mathfrak{P}| \sim \mathrm{Bin}(n, \alpha)$

$$\mathbb{P}^{\mathfrak{A}}\left(\widehat{\mathcal{R}}^p(h^*) > \frac{3\alpha}{2} \right) \leq \mathbb{P}^{\mathfrak{A}}\left(|\mathfrak{P}| \geq \frac{3\alpha}{2} n \right) \leq \exp(-\frac{\alpha n}{12}) \leq \frac{\delta}{12}.$$

It follows that $\mathbb{P}^{\mathfrak{A}}\left(h^* \notin \mathcal{H}^*\right) \leq \frac{\delta}{4} + \frac{\delta}{12} = \frac{\delta}{3}$.

Step 2 Next let $\mathcal{H}_1 \subset \mathcal{H}$ be the set $\left\{h \in \mathcal{H} \middle| \mathcal{R}(h,\mathbb{P}) > \frac{3\alpha}{1-\eta}\right\}$. For any $h \in \mathcal{H}_1$

$$\begin{aligned}\mathbb{P}^{\mathfrak{A}}\left(\widehat{\mathcal{R}}^c(h) \leq 3\alpha\right) &\leq \mathbb{P}^{\mathfrak{A}}\left(\mathrm{Bin}\left(n, \frac{3\alpha}{1-\eta}\right) \leq (1-\eta)\frac{3\alpha}{(1-\eta)}n\right) \\ &\leq \exp\left(-\eta^2\frac{3\alpha}{2(1-\eta)}n\right) \\ &\leq \frac{\delta}{8|\mathcal{H}|},\end{aligned}$$

as long as $n \geq \frac{2\log(\frac{8|\mathcal{H}|}{\delta})}{3\eta^2\alpha} > \frac{2\log(\frac{8|\mathcal{H}|}{\delta})(1-\eta)}{3\eta^2\alpha}$. Taking a union bound over all $h \in \mathcal{H}_1$,

$$\mathbb{P}^{\mathfrak{A}}\left(\min_{h\in\mathcal{H}_1}\widehat{\mathcal{R}}^c(h) \leq 3\alpha\right) \leq \frac{\delta}{8}$$

Since also $\mathbb{P}^{\mathfrak{A}}(|\mathfrak{P}| \geq \frac{3\alpha}{2}) \leq \frac{\delta}{12}$ and $\widehat{\mathcal{R}}^p(h) \geq \widehat{\mathcal{R}}^c(h) - |\mathfrak{P}|$, we obtain

$$\mathbb{P}^{\mathfrak{A}}\left(\min_{h\in\mathcal{H}_1}\widehat{\mathcal{R}}^p(h) \leq \frac{3\alpha}{2}\right) \leq \mathbb{P}^{\mathfrak{A}}\left(\left(\min_{h\in\mathcal{H}_1}\widehat{\mathcal{R}}^c(h) \leq 3\alpha\right) \vee \left(|\mathfrak{P}| \geq \frac{3\alpha}{2}\right)\right) \leq \frac{\delta}{8} + \frac{\delta}{12} = \frac{5\delta}{24}. \tag{C.30}$$

Similarly, let $\mathcal{H}_2 = \left\{h \in \mathcal{H} \middle| \Gamma^{EOp}(h) > \frac{2}{1-\eta}\Delta^{EOp}\right\}$. Fix any $h \in \mathcal{H}_2$. Assume without loss of generality that $\bar{\gamma}_{10} \geq \bar{\gamma}_{11} \geq 0$ (for this particular h only). Then $\bar{\gamma}_{10} \geq \bar{\gamma}_{10} - \bar{\gamma}_{11} = |\bar{\gamma}_{10} - \bar{\gamma}_{11}| = |\gamma_{10} - \gamma_{11}| > \frac{2}{1-\eta}\Delta^{EOp}$ (note that the γ_{1a} are non-negative). At the same time, by Lemma 7(a),

$$\mathbb{P}^{\mathfrak{A}}\left(\bar{\gamma}^p_{10} \leq (1-\eta)\bar{\gamma}_{10} - \bar{\Delta}_{10}\right) \leq \frac{\delta}{8|\mathcal{H}|},$$

whenever

$$n > \max\left\{\frac{8\log(\frac{16|\mathcal{H}|}{\delta})}{(1-\alpha)P_{10}}, \frac{4\log(\frac{16|\mathcal{H}|}{\delta})}{\eta^2(1-\alpha)P_{10}\bar{\gamma}_{10}}\right\}.$$

This is indeed the case since $n > \frac{8\log(\frac{16|\mathcal{H}|}{\delta})}{(1-\alpha)P_{10}}$ by assumption and also

$$n > \frac{2\log(\frac{16|\mathcal{H}|}{\delta})}{3\eta^2(1-\alpha)P_{10}\alpha} \geq \frac{4\log(\frac{16|\mathcal{H}|}{\delta})}{\eta^2(1-\alpha)P_{10}\bar{\gamma}_{10}}.$$

The last inequality is obtained by observing that $\bar{\gamma}_{10} \geq \frac{2}{1-\eta}\Delta^{EOp} \geq 6\alpha$, which follows by using $P_{10} \leq 0.5, \alpha \leq 0.5, \eta > 0$.

Therefore, with probability at least $1 - \frac{\delta}{8|\mathcal{H}|}$, $\max_a \bar{\gamma}^p_{1a} = \bar{\gamma}^p_{10} > (1-\eta)\bar{\gamma}_{10} - \Delta_{10} \geq 2\Delta^{EOp} - E_{10} \geq 2\Delta^{EOp} - E_{10} - E_{11}$, with $E_{1a} = \frac{B_{1a}}{C_{1a}+B_{1a}}$, where we used inequality (C.21).

Crucially, $2\Delta^{EOp} - E_{10} - E_{11}$ does not depend on h. Therefore, taking a union bound over all $h \in \mathcal{H}_2$,

$$\mathbb{P}^{\mathfrak{A}}\left(\min_{h\in\mathcal{H}_2}\max_a \bar{\gamma}^p_{1a}(h) \leq 2\Delta^{EOp} - E_{10} - E_{11}\right) \leq \frac{\delta}{8}.$$

Note also that since $n \geq \max\left\{\frac{8\log(16/\delta)}{(1-\alpha)P_{10}}, \frac{12\log(12/\delta)}{\alpha}\right\}$, using inequality (C.22),

$$\mathbb{P}^{\mathfrak{A}}\left(E_{10} + E_{11} > \Delta^{EOp}\right) \leq \frac{\delta}{4}.$$

Therefore

$$\begin{aligned}\mathbb{P}^{\mathfrak{A}}\left(\min_{h\in\mathcal{H}_2}\max_a \bar{\gamma}^p_{1a}(h) \leq \Delta^{EOp}\right) &\leq \mathbb{P}^{\mathfrak{A}}\left(\min_{h\in\mathcal{H}_2}\max_a \bar{\gamma}^p_{1a} \leq 2\Delta^{EOp} - E_{10} - E_{11}\right) \\ &\quad + \mathbb{P}^{\mathfrak{A}}\left(E_{10} + E_{11} > \Delta^{EOp}\right) \\ &\leq \frac{3\delta}{8}. \end{aligned} \tag{C.31}$$

Finally, using (C.30) and (C.31),

$$\begin{aligned}\mathbb{P}^{\mathfrak{A}}\left(\mathcal{H}^* \cap (\mathcal{H}_1 \cup \mathcal{H}_2) \neq \emptyset\right) &= \mathbb{P}^{\mathfrak{A}}\left(\left(\min_{h\in\mathcal{H}_1}\widehat{\mathcal{R}}^p(h) \leq \frac{3\alpha}{2}\right) \vee \left(\min_{h\in\mathcal{H}_2}\max_a \bar{\gamma}^p_{1a}(h) \leq \Delta^{EOp}\right)\right) \\ &\leq \frac{5\delta}{24} + \frac{3\delta}{8} \\ &< \frac{2\delta}{3}.\end{aligned}$$

Step 3 Combining steps 1 and 2, we have that with probability at least $1-\delta$, $h^* \in \mathcal{H}^*$ (and so $\mathcal{H}^*$ is non-empty) and for any $h \in \mathcal{H}$, $\mathcal{R}(h,\mathbb{P}) \leq \frac{3\alpha}{1-\eta}$ and $\Gamma^{EOp}(h) \leq \frac{2}{1-\eta}\Delta^{EOp}$ which completes the proof. □

APPENDIX D

Proofs from Chapter 6

Here we present the complete proof of Theorem 17. To this end, we first show in Section D.1 how the technique of [Jan04] for studying the large deviations of sums of dependent random variables can be used to derive large deviation bounds for the three fairness notions, given a fixed classifier. The proof is similar to the corresponding i.i.d. result of [WGOS17], however an application of the results from [Jan04] is needed because of the dependence between the samples. Then in Section D.2 we show how these bounds can be made uniform over the hypothesis space by adapting the classic symmetrization argument (e.g. [Vap13]) to a dependent data scenario.

D.1 Non-uniform bounds

First we use the tools from the previous section and a technique of [WGOS17, ABD+18] to show a non-uniform Hoeffding-type bound for equality of opportunity and equalized odds:

Lemma 8. *Fix $\delta \in (0,1)$ and a binary predictor $f : \mathcal{Q} \times \mathcal{I} \to \{0,1\}$. Suppose that $N > \frac{8\log(8/\delta)}{\tau^2}$, where $\tau = \min_{ar} \mathbb{P}(A(d) = a, r(q,d) = r)$, then:*

$$\mathbb{P}\left(|\Gamma^{EOp}(f,S) - \Gamma^{EOp}(f)| > 2\sqrt{\frac{\log(8/\delta)}{N\tau}}\right) \leq \delta. \tag{D.1}$$

and

$$\mathbb{P}\left(|\Gamma^{EOd}(f,S) - \Gamma^{EOd}(f)| > 2\sqrt{\frac{\log(16/\delta)}{N\tau}}\right) \leq \delta. \tag{D.2}$$

Proof. Denote by $I_{ar} = \{(i,j) : A(d_j^i) = a, r(q_i, d_j^i) = r\}$ the set of indexes of the training data for which the document belongs to the group a and the relevance of the query-document pair is r. Notice that I_{ar} is a random variable and that $|I_{ar}| = |S_{a,r}|$. We first bound the probability of a large deviation of

$$\gamma_{ar}^S(f) := \frac{1}{|I_{ar}|} \sum_{(i,j)\in I_{ar}} f(q_i, d_j^i)$$

from $\gamma_{ar}(f) := \mathbb{P}(f(q,d) = 1 | A(d) = a, r(q,d) = r)$, for each pair $r \in \{0,1\}, a \in \{0,1\}$. Since f is fixed here, we omit the dependence of $\gamma_{ar}(f), \gamma^S_{ar}(f), \Gamma^{\mathrm{EOp}}(f), \Gamma^{\mathrm{EOd}}(f)$, etc. on f for the rest of this proof.

For any fixed I_{ar}:

$$\mathbb{E}\left(\gamma^S_{ar} | I_{ar}\right) = \mathbb{E}\left(\frac{1}{|I_{ar}|} \sum_{(i,j)\in I_{ar}} f(q_i, d^i_j)\right) = \mathbb{P}(f(q,d) = 1 | A(d) = a, r(q,d) = r) = \gamma_{ar}(f), \tag{D.3}$$

since the marginal distribution of every $(q_i, d^i_j, r(q_i, d^i_j))$ is $\mathcal{D}$. It is also easy to see that if $\mathcal{A} = \{(i,j) : i \in [N], j \in [m]\}$ is the index set of the random variables $Y_{(i,j)} = f(q_i, d^i_j)$, then $\chi(\mathcal{A}) = m$. Therefore, for any fixed set $I_{ar} \subset \mathcal{A}$, we have $\chi(I_{ar}) \leq \chi(\mathcal{A}) = m$. Now conditional on I_{ar}:

$$\mathbb{E}(|\gamma^S_{ar} - \gamma_{ar}| > t | I_{ar}) = \mathbb{E}\left(\left|\frac{1}{|I_{ar}|} \sum_{(i,j)\in I_{ar}} f(q_i, d^i_j) - \gamma_{ar}\right| > t\right) \leq 2\exp\left(-2\frac{t^2|I_{ar}|}{m}\right). \tag{D.4}$$

Similarly, $|I_{ar}| = \sum_{i\in[N]} \sum_{j\in[m]} \mathbb{1}(r(q_i, d^i_j) = r, A(d^i_j) = a)$ is the sum of Nm Bernoulli random variables indexed by $\mathcal{A} = \{(i,j)\}_{i\in[N], j\in[m]}$, such that $\chi(\mathcal{A}) = m$. Denote by $P_{ar} = \mathbb{P}_{(q,d,r)\sim\mathcal{D}}(A(d) = a, r(q,d) = r)$ and recall the notation $\tau = \min_{ar} P_{ar}$. Then $\mathbb{E}(|I_{ar}|) = P_{ar}Nm$. Therefore,

$$\mathbb{P}\left(|I_{ar}| \leq P_{ar}Nm - t\right) \leq \exp\left(-2\frac{t^2}{Nm^2}\right).$$

Setting $t = P_{ar}Nm/2$, we obtain:

$$\mathbb{P}\left(|I_{ar}| \leq \frac{P_{ar}}{2}Nm\right) \leq \exp\left(-\frac{P^2_{ar}N}{2}\right). \tag{D.5}$$

Now assume that $N \geq \frac{2\log(8/\delta)}{P^2}$. Then for any $r \in \{0,1\}, a \in \{0,1\}$:

$$\begin{aligned}
\mathbb{P}(|\gamma^S_{ar} - \gamma_{ar}| > t) &= \sum_{I_{ar}} \mathbb{P}(|\gamma^S_{ar} - \gamma_{ar}| > t | I_{ar}) \mathbb{P}(I_{ar}) \\
&\leq \mathbb{P}(|I_{ar}| \leq \frac{P_{ar}}{2}Nm) + \sum_{I_{ar}:|I_{ar}|\geq\frac{P_{ar}Nm}{2}} \mathbb{P}(|\gamma^S_{ar} - \gamma_{ar}| > t | I_{ar}) \mathbb{P}(I_{ar}) \\
&\leq \exp\left(-\frac{P^2_{ar}N}{2}\right) + \sum_{I_{ar}:|I_{ar}|\geq\frac{P_{ar}Nm}{2}} 2\exp\left(-2\frac{t^2|I_{ar}|}{m}\right) \mathbb{P}(S_{ar}) \\
&\leq \frac{\delta}{8} + 2\exp\left(-t^2 N P_{ar}\right).
\end{aligned}$$

The rest of the proof proceeds as in [WGOS17]. For a fixed $r \in \{0,1\}$ the triangle law gives:

$$||\gamma^S_{0r} - \gamma^S_{1r}| - |\gamma_{0r} - \gamma_{1r}|| \leq |\gamma^S_{0r} - \gamma^S_{1r} - \gamma_{0r} + \gamma_{1r}| \leq |\gamma^S_{0r} - \gamma_{0r}| + |\gamma^S_{1r} - \gamma_{1r}|.$$

Therefore,

$$\mathbb{P}(||\gamma^S_{0r} - \gamma^S_{1r}| - |\gamma_{0r} - \gamma_{1r}|| > 2t) \leq \mathbb{P}(|\gamma^S_{0r} - \gamma_{0r}| + |\gamma^S_{1r} - \gamma_{1r}| > 2t)$$

$$
\begin{aligned}
&\leq \mathbb{P}((|\gamma^S_{0r} - \gamma_{0r}| > t) \vee (|\gamma^S_{1r} - \gamma_{1r}| > t)) \\
&\leq \mathbb{P}(|\gamma^S_{0r} - \gamma_{0r}| > t) + \mathbb{P}(|\gamma^S_{1r} - \gamma_{1r}| > t) \\
&\leq \frac{\delta}{4} + 4\exp(-t^2 N\tau).
\end{aligned}
$$

Setting $t = t_0 = \sqrt{\frac{\log(16/\delta)}{N\tau}}$ gives:

$$
\mathbb{P}\left(||\gamma^S_{0r} - \gamma^S_{1r}| - |\gamma_{0r} - \gamma_{1r}|| > 2\sqrt{\frac{\log(16/\delta)}{N\tau}}\right) \leq \frac{\delta}{4} + 4\frac{\delta}{16} = \frac{\delta}{2}.
$$

Setting $r = 1$ gives the first result.

For the second result, note that taking the union bound over $r \in \{0, 1\}$ shows that with probability at least $1-\delta$ both $||\gamma^S_{00} - \gamma^S_{10}| - |\gamma_{00} - \gamma_{10}|| \leq 2t_0$ and $||\gamma^S_{01} - \gamma^S_{11}| - |\gamma_{01} - \gamma_{11}|| \leq 2t_0$ hold.

Under this event we have:

$$
\begin{aligned}
|\Gamma^{\mathrm{EOd}}(f, S) - \Gamma^{\mathrm{EOd}}(f)| &= \left|\frac{1}{2}\left(|\gamma^S_{00} - \gamma^S_{10}| + |\gamma^S_{01} - \gamma^S_{11}|\right) - \frac{1}{2}\left(|\gamma_{00} - \gamma_{10}| + |\gamma_{01} - \gamma_{11}|\right)\right| \\
&= \left|\frac{1}{2}\left(|\gamma^S_{00} - \gamma^S_{10}| - |\gamma_{00} - \gamma_{10}|\right) + \frac{1}{2}\left(|\gamma^S_{01} - \gamma^S_{11}| - |\gamma_{01} - \gamma_{11}|\right)\right| \\
&\leq \frac{1}{2}\left||\gamma^S_{00} - \gamma^S_{10}| - |\gamma_{00} - \gamma_{10}|\right| + \frac{1}{2}\left||\gamma^S_{01} - \gamma^S_{11}| - |\gamma_{01} - \gamma_{11}|\right| \\
&\leq 2t_0
\end{aligned}
$$

and hence the result follows. □

An identical argument, by conditioning on the values of the set $I_a = \{(i, j) : A(d^i_j) = a\}$ gives a similar result for demographic parity:

Lemma 9. *Fix $\delta \in (0, 1)$ and a binary predictor $f : \mathcal{Q} \times \mathcal{I} \to \{0, 1\}$. Suppose that $N > \frac{8\log(8/\delta)}{\upsilon^2}$, where $\upsilon = \min_a \mathbb{P}(A(d) = a)$, then:*

$$
\mathbb{P}\left(|\Gamma^{DP}(f, S) - \Gamma^{DP}(f)| > 2\sqrt{\frac{\log(8/\delta)}{N\upsilon}}\right) \leq \delta. \tag{D.6}
$$

D.2 Uniform bounds on proof of Theorem 17

In this section we show how to formally extend the non-uniform bounds from the previous section to hold uniformly over the hypothesis space $\mathcal{H}$.

Let $S' = \{(q'_i, d'^i_j, r(q'_i, d'^i_j))\}_{i\in[N], j\in[m]}$ be a ghost sample independent of S and also sampled via the same procedure as S, as described in the main body of the paper. In the proof of Lemma 8 we showed that for any classifier f and any $t \in (0, 1)$:

$$
\begin{aligned}
\mathbb{P}\left(|\Gamma^{\mathrm{EOp}}(f) - \Gamma^{\mathrm{EOp}}(f, S)| > 2t\right) &\leq 2\exp\left(-\frac{\tau^2 N}{2}\right) + 4\exp\left(-\frac{t^2 N\tau}{2}\right) \\
&\leq 6\exp\left(-\frac{t^2 N\tau^2}{2}\right)
\end{aligned} \tag{D.7}
$$

$$\mathbb{P}\left(|\Gamma^{\mathrm{EOd}}(f) - \Gamma^{\mathrm{EOd}}(f,S)| > 2t\right) \leq 4\exp\left(-\frac{\tau^2 N}{2}\right) + 8\exp\left(-\frac{t^2 N\tau}{2}\right)$$
$$\leq 12\exp\left(-\frac{t^2 N\tau^2}{2}\right) \tag{D.8}$$

Similarly, from the proof of Lemma 9

$$\mathbb{P}\left(|\Gamma^{\mathrm{DP}}(f) - \Gamma^{\mathrm{DP}}(f,S)| > 2t\right) \leq 2\exp\left(-\frac{\upsilon^2 N}{2}\right) + 4\exp\left(-\frac{t^2 N\upsilon}{2}\right) \leq 6\exp\left(-\frac{t^2 N\upsilon^2}{2}\right) \tag{D.9}$$

We will use these in particular to prove the following symmetrization lemma:

Lemma 10. *For any* $1 > t \geq 4\sqrt{\frac{2\log(12)}{N\tau^2}}$,

$$\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{EOp}(f) - \Gamma^{EOp}(f,S)) \geq t\right) \leq 2\mathbb{P}_{S,S'}\left(\sup_{f\in\mathcal{F}}(\Gamma^{EOp}(f,S') - \Gamma^{EOp}(f,S)) \geq t/2\right). \tag{D.10}$$

For any $1 > t \geq 4\sqrt{\frac{2\log(24)}{N\tau^2}}$*:*

$$\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{EOd}(f) - \Gamma^{EOd}(f,S)) \geq t\right) \leq 2\mathbb{P}_{S,S'}\left(\sup_{f\in\mathcal{F}}(\Gamma^{EOd}(f,S') - \Gamma^{EOd}(f,S)) \geq t/2\right). \tag{D.11}$$

For any $1 > t \geq 4\sqrt{\frac{2\log(12)}{N\upsilon^2}}$*:*

$$\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{DP}(f) - \Gamma^{DP}(f,S)) \geq t\right) \leq 2\mathbb{P}_{S,S'}\left(\sup_{f\in\mathcal{F}}(\Gamma^{DP}(f,S') - \Gamma^{DP}(f,S)) \geq t/2\right). \tag{D.12}$$

Proof. We show the result for the equality of opportunity fairness measure, the rest follow in an identical manner.

Let f^* be the function achieving the supremum on the left-hand side [1]. Note that:

$$\begin{aligned}
&\mathbb{1}(\Gamma^{\mathrm{EOp}}(f^*) - \Gamma^{\mathrm{EOp}}(f^*,S) \geq t)\mathbb{1}(\Gamma^{\mathrm{EOp}}(f^*) - \Gamma^{\mathrm{EOp}}(f^*,S') < t/2)\\
&\quad = \mathbb{1}(\Gamma^{\mathrm{EOp}}(f^*) - \Gamma^{\mathrm{EOp}}(f^*,S) \geq t \wedge \Gamma^{\mathrm{EOp}}(f^*,S') - \Gamma^{\mathrm{EOp}}(f^*) > -t/2)\\
&\quad \leq \mathbb{1}(\Gamma^{\mathrm{EOp}}(f^*,S') - \Gamma^{\mathrm{EOp}}(f^*,S) > t/2).
\end{aligned}$$

Taking expectation with respect to S':

$$\begin{aligned}
&\mathbb{1}(\Gamma^{\mathrm{EOp}}(f^*) - \Gamma^{\mathrm{EOp}}(f^*,S) \geq t)\mathbb{P}_{S'}(\Gamma^{\mathrm{EOp}}(f^*) - \Gamma^{\mathrm{EOp}}(f^*,S') < t/2)\\
&\quad \leq \mathbb{P}_{S'}(\Gamma^{\mathrm{EOp}}(f^*,S') - \Gamma^{\mathrm{EOp}}(f^*,S) > t/2).
\end{aligned}$$

Now using (D.7):

$$\mathbb{P}_{S'}(\Gamma^{\mathrm{EOp}}(f^*) - \Gamma^{\mathrm{EOp}}(f^*,S') \geq t/2) \leq 6\exp\left(-\frac{t^2 N\tau^2}{32}\right) \leq \frac{1}{2},$$

[1] If the supremum is not attained, this argument can be repeated for each element of a sequence of classifiers approaching the supremum

so:

$$\frac{1}{2}\mathbb{1}(\Gamma^{\text{EOp}}(f^*) - \Gamma^{\text{EOp}}(f^*, S) \geq t) \leq \mathbb{P}_{S'}(\Gamma^{\text{EOp}}(f^*, S') - \Gamma^{\text{EOp}}(f^*, S) > t/2).$$

Taking expectation with respect to S:

$$\begin{aligned}\mathbb{P}_S(\Gamma^{\text{EOp}}(f^*) - \Gamma^{\text{EOp}}(f^*, S) \geq t) &\leq 2\mathbb{P}_{S,S'}(\Gamma^{\text{EOp}}(f^*, S') - \Gamma^{\text{EOp}}(f^*, S) > t/2) \\ &\leq 2\mathbb{P}_{S,S'}(\sup_{f\in\mathcal{F}}(\Gamma^{\text{EOp}}(f, S') - \Gamma^{\text{EOp}}(f, S)) \geq t/2).\end{aligned}$$

□

Given a set of n input datapoints $z_1, \ldots, z_n$ with $z_i = (q_i, d_i, r(q_i, d_i))$, consider:

$$\mathcal{F}_{z_1,\ldots,z_n} = \{(f(q_1, d_1), \ldots, f(q_n, d_n)) : f \in \mathcal{F}\} \tag{D.13}$$

Then the growth function of $\mathcal{F}$ is defined as:

$$S_{\mathcal{F}}(n) = \sup_{(z_1,\ldots,z_n)} |\mathcal{F}_{z_1,\ldots,z_n}| \tag{D.14}$$

We can now present a proof of Theorem 17:

Theorem 17. *Suppose that $v = VC(\mathcal{F}) \geq 1$ and that $2Nm > v$. Then for any $\delta \in (0, 1)$:*

$$\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{EOp}(f) - \Gamma^{EOp}(f, S)) \geq 8\sqrt{2\frac{d\log(\frac{2eNm}{d}) + \log(\frac{24}{\delta})}{N\tau^2}}\right) \leq \delta \tag{D.15}$$

$$\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{DP}(f) - \Gamma^{DP}(f, S)) \geq 8\sqrt{2\frac{d\log(\frac{2eNm}{d}) + \log(\frac{24}{\delta})}{Nv^2}}\right) \leq \delta \tag{D.16}$$

$$\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{EOd}(f) - \Gamma^{EOd}(f, S)) \geq 8\sqrt{2\frac{d\log(\frac{2eNm}{d}) + \log(\frac{48}{\delta})}{N\tau^2}}\right) \leq \delta \tag{D.17}$$

Proof. Again we present the proof for equality of opportunity, with the other inequalities following in an identical manner.

Note that given sets S and S', the values of $\Gamma^{\text{EOp}}(f, S)$ and $\Gamma^{\text{EOp}}(f, S')$ are completely determined by the values of f on S and S' respectively. Therefore, for any $t \in \left(4\sqrt{\frac{2\log(12)}{N\tau^2}}, 1\right)$ using Lemma 10 and the union bound:

$$\begin{aligned}\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{\text{EOp}}(f) - \Gamma^{\text{EOp}}(f, S)) \geq t\right) &\leq 2\mathbb{P}_{S,S'}\left(\sup_{f\in\mathcal{F}}(\Gamma^{\text{EOp}}(f, S') - \Gamma^{\text{EOp}}(f, S)) \geq t/2\right) \\ &\leq 2S_{\mathcal{F}}(2Nm)\mathbb{P}_{S,S'}\left(\Gamma^{\text{EOp}}(f, S') - \Gamma^{\text{EOp}}(f, S) \geq t/2\right) \\ &\leq 2S_{\mathcal{F}}(2Nm)\mathbb{P}_{S,S'}\left((|\Gamma^{\text{EOp}}(f, S') - \Gamma^{\text{EOp}}(f)| \geq t/4)\right. \\ &\quad \left.\vee\ (|\Gamma^{\text{EOp}}(f) - \Gamma^{\text{EOp}}(f, S)| \geq t/4)\right) \\ &\leq 4S_{\mathcal{F}}(2Nm)\mathbb{P}_S\left(|\Gamma^{\text{EOp}}(f) - \Gamma^{\text{EOp}}(f, S)| \geq t/4\right) \\ &\leq 24S_{\mathcal{F}}(2Nm)\exp\left(-\frac{t^2N\tau^2}{128}\right)\end{aligned}$$

In particular, if $d = VC(\mathcal{F})$, by Sauer's lemma $S_{\mathcal{F}}(2Nm) \leq \left(\frac{2eNm}{d}\right)^d$ whenever $2Nm > d$, so:

$$\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{\mathrm{EOp}}(f) - \Gamma^{\mathrm{EOp}}(f,S)) \geq t\right) \leq 24\left(\frac{2eNm}{d}\right)^d \exp\left(-\frac{t^2N\tau^2}{128}\right)$$

It follows that:

$$\mathbb{P}_S\left(\sup_{f\in\mathcal{F}}(\Gamma^{\mathrm{EOp}}(f) - \Gamma^{\mathrm{EOp}}(f,S)) \geq 8\sqrt{2\frac{d\log(\frac{2eNm}{d}) + \log(\frac{24}{\delta})}{N\tau^2}}\right) \leq \delta \tag{D.18}$$

whenever:

$$1 > 8\sqrt{2\frac{d\log(\frac{2eNm}{d}) + \log(\frac{24}{\delta})}{N\tau^2}} \geq 4\sqrt{\frac{2\log(12)}{N\tau^2}}$$

It is easy to see that the right inequality holds whenever $d \geq 1$, $2Nm \geq d$ and $\delta < 1$. In addition, inequality (D.18) trivially holds if the left inequality is not fulfilled. Hence the result follows. □

,