ABSTRACT

The availability of large datasets and computational resources has driven significant progress in Artificial Intelligence (AI) and, especially, Machine Learning (ML). These advances have rendered AI systems instrumental for many decision making and policy operations involving individuals: they include assistance in legal decisions, lending, and hiring, as well determinations of resources and benefits, all of which have profound social and economic impacts. While data-driven systems have been successful in an increasing number of tasks, the use of rich datasets, combined with the adoption of black-box algorithms, has sparked concerns about how these systems operate. How much information these systems leak about the individuals whose data is used as input and how they handle biases and fairness issues are two of these critical concerns. While some people argue that privacy and fairness are in alignment, the majority instead believe these are two contrasting metrics.

This thesis firstly studies the interaction between privacy and fairness in machine learning and decision problems. It focuses on the scenario when fairness and privacy are at odds and investigates different factors that can explain for such behaviors. It then proposes effective and efficient mitigation solutions to improve fairness under privacy constraints. In the second part, it analyzes the connection between fairness and other machine learning concepts such as model compression and adversarial robustness. Finally, it introduces a novel privacy concept and an initial implementation to protect such proposed users privacy at inference time.

THE INTERPLAY BETWEEN PRIVACY AND FAIRNESS IN LEARNING AND DECISION MAKING PROBLEMS

By

Cuong Tran
B.E., Hanoi University of Science and Technology, 2012

DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Computer & Information Science & Engineering

Syracuse University
June 2023

Acknowledgments

"Give me six hours to chop down a tree and I will spend the first four sharpening the axe."

attributed to Abraham Lincoln

I would like to take this opportunity to express my sincere gratitude to several individuals who have been instrumental in the successful completion of my PhD thesis. Firstly, I would like to thank my advisor, Prof. Ferdinando Fioretto, for his unwavering support, valuable guidance, and insightful feedback throughout my research journey. His expertise and encouragement have been invaluable in shaping my academic career.

I am also deeply grateful to my parents, Loc Le and Toan Tran, and my brother, Khanh Tran, for their unconditional love, constant encouragement, and unwavering support. Their sacrifices and unwavering support have been instrumental in helping me achieve my academic goals. I could not have done this without their unwavering support and encouragement.

I would also like to express my heartfelt appreciation to the members of our lab including James Kotary, My Dinh and Vincenzo Francessco for their camaraderie, friendship, and intellectual support. Their invaluable feedback, constructive criticism, and insightful discussions have been invaluable in shaping my ideas and helping me to refine my research methodology.

Once again, I am grateful to everyone who has contributed to my academic success, and I will cherish their support and encouragement throughout my life

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

With the growing of data collected in many channels, more and more data driven applications has been investigated and deployed successfully in many aspects of human lives. Noticeably machine learning models are using to replace human efforts in many domains such as health care, banking, finance or transportation. Given their success however there are **two** ethical consequences of these models which are getting more and more attention. First, the intelligent models are trained on a dataset and if the data is bias the model can be discriminative towards certain groups of people. For example, it was recently showed that most commercial facial recognition softwares are much more accurate on White faces than Black faces. Furthermore, some credit scoring models are showed to accept loan applications from men than women. The unfairness consequence of learning models is the first important problem that we should resolve. Second, since the models are trained on personal data which can contain sensitive information, these models can reveal certain confidential information of users.

As a solution for privacy, *Differential Privacy* (DP) [37] has become the paradigm of choice for protecting data privacy and its deployments are also growing at a fast rate. These include several data products related with the 2020 release of the US. Census Bureau [5], and by Google [10], Facebook [57], and Apple [120]. DP is appealing as it bounds the risks of disclosing sensitive information for individuals participating in a computation. However, the process adopted by a

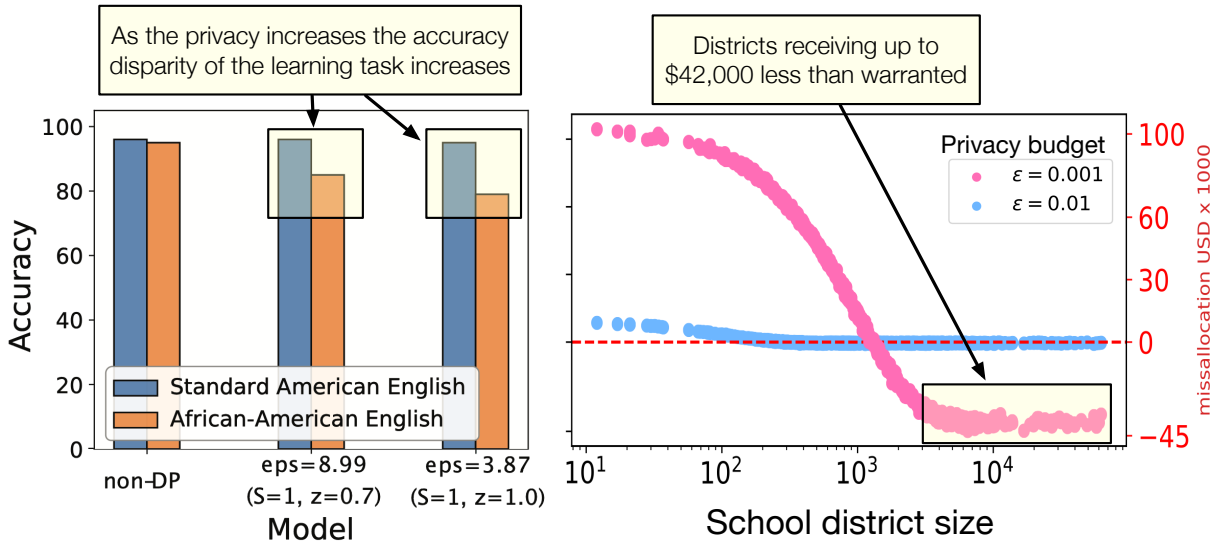


Fig. 1.1: Left: Disparities arising in DP sentiment analysis tasks (image from [12]). Right: Disparity arising in fund allocations to school districts (image from [125]).

DP algorithm to ensure the privacy guarantees involves calibrated perturbations, which inevitably introduce errors to the outputs of the task at hand. More importantly, it has been shown that these errors may affect different groups of individuals differently. An example of these effects are reported in Figure 1.1 (left), which illustrates that a DP learning model affects the accuracy of the minority group (African-American) more than it does the majority group in a sentiment analysis of Tweets [12]. Similar observations were reported in decision tasks (Figure 1.1, right) in which privacy-preserving census data is used to allocate funds to school districts [45]. The illustration shows that, under a privacy-preserving allocation scheme, some school districts may systematically receive considerably less money than what would be warranted otherwise.

These effects can have significant societal and economic impacts on the involved individuals: classification errors may penalize some groups over others in important determinations, including criminal assessment, landing, and hiring, or can result in disparities regarding the allocation of critical funds, benefits, and therapeutics. These fairness issues in DP settings are receiving increasing attention, but a complete understanding of why they arise is still limited. This thesis is firstly devoted to analyze various factors that can attribute for the contrasting property between fairness and privacy. We analyze such conflicting behavior for decision and machine learning problems. We

develop effective mitigation solution to reduce such negative impact of privacy towards fairness on each case. Furthermore, in the second part of the thesis we study the dichotomy between fairness and other machine learning concepts such as model compression and adversarial robustness. In particular, we show that widely used model compression *network pruning* can introduce unfair outcomes w.r.t group accuracy. Likewise, fair models are more vulnerable against the attacks than the regular unfair counterparts. Finally, the thesis proposes a novel privacy preserving framework to protect privacy at the inference time. This is of critical importance since most of privacy preserving frameworks in literature only concentrates on protecting privacy for training data.

1.1 Dissertation organization and chapter summaries

The remainder of the thesis is organized as follows. In Chapter 2 we study the release of differentially private data sets and analyzes their impact on some critical resource allocation tasks under a fairness perspective. In Chapter 3, we provide a thorough theoretical justification on the causes of the disparate impacts arising in the problem of differentially private empirical risk minimization (ERM). In Chapter 4, we analyze whether the use of Private Aggregation of Teacher Assemble (PATE) can result in unfairness, and demonstrates that it can lead to accuracy disparities among groups of individuals. In Chapter 5, we shows that pruning may create or exacerbate disparate impacts. Furthermore, in this chapter we shed light on the factors to cause such disparities, suggesting differences in gradient norms and distance to decision boundary across groups to be responsible for this critical issue. Next, in Chapter 6 we show the existence of a dichotomy between fairness and robustness, and analyzes when striving for fairness decreases the model robustness to adversarial samples. Subsequently, in Chapter 7 we asks whether it is necessary to require all input features for a model to return accurate predictions at test time and shows that, under a personalized setting, each individual may need to release only a small subset of these features without impacting the final decisions. Finally, we conclude the thesis in Chapter 8 by highlighting important future research directions.

1.2 Publications

Parts of the thesis has appeared in the following publications:

Conferences:

- **Cuong Tran**, Ferdinando Fioretto, Pascal Van Hentenryck, and Zhiyan Yao. Decision making with differential privacy under the fairness lens. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2021 (**in Chapter 2**).
- **Cuong Tran**, My H. Dinh, and Ferdinando Fioretto. Differentially private deep learning under the fairness lens. In Advances in Neural Information Processing Systems (NeurIPS), 2021 (**in Chapter 3**).
- **Cuong Tran** and Ferdinando Fioretto. On the fairness impacts of private ensembles models. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2023 (**in Chapter 4**).
- **Cuong Tran**, Ferdinando Fioretto, Jung-Eun Kim, and Rakshit Naidu. Pruning has a disparate impact on model accuracy. In Advances in Neural Information Processing Systems (NeurIPS), 2022 (**in Chapter 5**).
- **Cuong Tran**, Keyu Zhu, Ferdinando Fioretto, and Pascal Van Hentenryck. Fairness increases adversarial vulnerability. Under submission to CVPR, 2023 (**in Chapter 6**).
- **Cuong Tran** and Ferdinando Fioretto. Personalized privacy auditing and optimization at test time. Under submission to ICML, 2023 (**in Chapter 7**).

Furthermore, the following publications are not included in this thesis:

Conferences:

- Ferdinando Fioretto, **Cuong Tran**, Keyu Zhu, and Pascal Van Hentenryck. Differential privacy and fairness in decisions and learning tasks: A survey. In In IJCAI Survey Track, 2022.

- **Cuong Tran**, Ferdinando Fioretto, and Pascal Van Hentenryck. Sf-pate: Scalable, fair, and private aggregation of teacher ensembles. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2023.

- **Cuong Tran**, Ferdinando Fioretto, and Pascal Van Hentenryck. Differentially private and fair deep learning: A lagrangian dual approach. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2021.

- Ferdinando Fioretto, Pascal Van Hentenryck, Terrence W.K. Mak, **Cuong Tran**, Federico Baldo, and Michele Lombardi. A Lagrangian Dual Framework for Deep Neural Networks with Constraints. Proceedings of the European Conference on Machine Learning (ECML), 2020.

- Anudit Nagar, **Cuong Tran**, Ferdinando and Fioretto. A Privacy-Preserving and Accountable Multi-agent Learning Framework. Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 2021.

CHAPTER 2
DISPARATE IMPACTS OF PRIVACY INTO DECISION TASKS

Agencies, such as the U.S. Census Bureau, release data sets and statistics about groups of individuals that are used as input to a number of critical decision processes. To conform with privacy and confidentiality requirements, these agencies are often required to release privacy-preserving versions of the data. This chapter studies the release of differentially private data sets and analyzes their impact on some critical resource allocation tasks under a fairness perspective. The chapter shows that, when the decisions take as input differentially private data, the noise added to achieve privacy disproportionately impacts some groups over others. The chapter analyzes the reasons for these disproportionate impacts and proposes guidelines to mitigate these effects. The proposed approaches are evaluated on critical decision problems that use differentially private census data.

2.1 Introduction

Agencies, such as the U.S. Census Bureau, release data sets and statistics about groups of individuals that are then used as inputs to a number critical decision processes. For example, the census data is used to decide whether a jurisdiction must provide language assistance during elections,

Title I fund allocation in education [71] and to establish national level COVID-19 vaccination distribution plans for states and jurisdictions [116]. The resulting decisions can have significant societal, economic, and medical impacts for participating individuals.

In many cases, the released data contain sensitive information and their privacy is strictly regulated. For example, in the U.S., the census data is regulated under Title 13 [2], which requires that no individual be identified from any data release by the Census Bureau. In Europe, data release are regulated according to the *General Data Protection Regulation* [48], which addresses the control and transfer of personal data.

Statistical agencies thus release *privacy-preserving* data and statistics that conform to privacy and confidentiality requirements. In the U.S., a small number of decisions, such as congressional apportionment, are taken using unprotected true values, but the vast majority of decisions rely on privacy-preserving data. Of particular interest are resource allocation decisions relying on the U.S. Census Bureau data, since the bureau will release several privacy-preserving data products using the framework of *Differential Privacy* [5] for their 2020 release. Recently, [71] empirically showed that differential privacy may have a disparate impact on several resource allocation problems. The noise introduced by the privacy mechanism may result in decisions that impact various groups differently.

This chapter builds on these observations and provides a step towards a deeper understanding of the fairness issues arising when differentially private data is used as input to several resource allocation problems. *One of its main results is to prove that several allotment problems and decision rules with significant societal impact (e.g., the allocation of educational funds, the decision to provide minority language assistance on election ballots, or the distribution of COVID-19 vaccines) exhibit inherent unfairness when applied to a differentially private release of the census data.* To counteract this negative results, the chapter examines the conditions under which decision making is fair when using differential privacy, and techniques to bound unfairness. The chapter also provides a number of mitigation approaches to alleviate biases introduced by differential privacy on such decision making problems. More specifically, the chapter makes the following contributions:

1. It formally defines notions of fairness and bounded fairness for decision making subject to privacy requirements.

2. It characterizes decision making problems that are fair or admits bounded fairness. In addition, it investigates the composition of decision rules and how they impact bounded fairness.

3. It proves that several decision problems with high societal impact induce inherent biases when using a differentially private input.

4. It examines the roots of the induced unfairness by analyzing the structure of the decision making problems.

5. It proposes several guidelines to mitigate the negative fairness effects of the decision problems studied.

To the best of the authors' knowledge, this is the first study that attempt at characterizing the relation between differential privacy and fairness in decision problems. All proofs are reported in the Section 2.9.

2.2 Preliminaries: differential privacy

Differential Privacy [37] (DP) is a rigorous privacy notion that characterizes the amount of information of an individual's data being disclosed in a computation.

Definition 2.1. *A randomized algorithm $\mathcal{M} : \mathcal{X} \to \mathcal{R}$ with domain $\mathcal{X}$ and range $\mathcal{R}$ satisfies* ϵ-differential privacy *if for any output $O \subseteq \mathcal{R}$ and data sets $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$ differing by at most one entry (written $\boldsymbol{x} \sim \boldsymbol{x}'$)*

$$\Pr[\mathcal{M}(\boldsymbol{x}) \in O] \leq \exp(\epsilon) \Pr[\mathcal{M}(\boldsymbol{x}') \in O]. \tag{2.1}$$

Parameter $\epsilon > 0$ is the *privacy loss*, with values close to 0 denoting strong privacy. Intuitively, DP states that any event occur with similar probability regardless of the participation of any individual data to the data set. DP satisfies several properties including *immunity to post-processing*,

which states that the privacy loss of DP outputs is not affected by arbitrary data-independent post-processing [38].

A function f from a data set $\boldsymbol{x} \in \mathcal{X}$ to a result set $R \subseteq \mathbb{R}^n$ can be made differentially private by injecting random noise onto its output. The amount of noise relies on the notion of *global sensitivity* $\Delta_f = \max_{\boldsymbol{x} \sim \boldsymbol{x}'} \|f(\boldsymbol{x}) - f(\boldsymbol{x}')\|_1$. The *Laplace mechanism* [37] that outputs $f(\boldsymbol{x}) + \boldsymbol{\eta}$, where $\boldsymbol{\eta} \in \mathbb{R}^n$ is drawn from the i.i.d. Laplace distribution with 0 mean and scale Δ_f/ϵ over n dimensions, achieves ϵ-DP.

2.3 Problem setting and goals

The chapter considers a dataset $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^k$ of n entities, whose elements $x_i = (x_{i1}, \ldots, x_{1k})$ describe k measurable quantities of entity $i \in [n]$, such as the number of individuals living in a geographical region i and their English proficiency. The chapter considers two classes of problems:

- An *allotment problem* $P : \mathcal{X} \times [n] \to \mathbb{R}$ is a function that distributes a finite set of resources to some problem entity. P may represent, for instance, the amount of money allotted to a school district.

- A *decision rule* $P : \mathcal{X} \times [n] \to \{0, 1\}$ determines whether some entity qualifies for some benefits. For instance, P may represent if election ballots should be described in a minority language for an electoral district.

The chapter assumes that P has bounded range, and uses the shorthand $P_i(\boldsymbol{x})$ to denote $P(\boldsymbol{x}, i)$ for entity i. The focus of the chapter is to study the effects of a DP data-release mechanism $\mathcal{M}$ to the outcomes of problem P. Mechanism $\mathcal{M}$ is applied to the dataset $\boldsymbol{x}$ to produce a privacy-preserving counterpart $\tilde{\boldsymbol{x}}$ and the resulting private outcome $P_i(\tilde{\boldsymbol{x}})$ is used to make some allocation decisions. Figure 2.1 provides an illustrative diagram.

Because random noise is added to the original dataset $\boldsymbol{x}$, the output $P_i(\tilde{\boldsymbol{x}})$ incurs some error. *The focus of this chapter is to characterize and quantify the disparate impact of this error among*

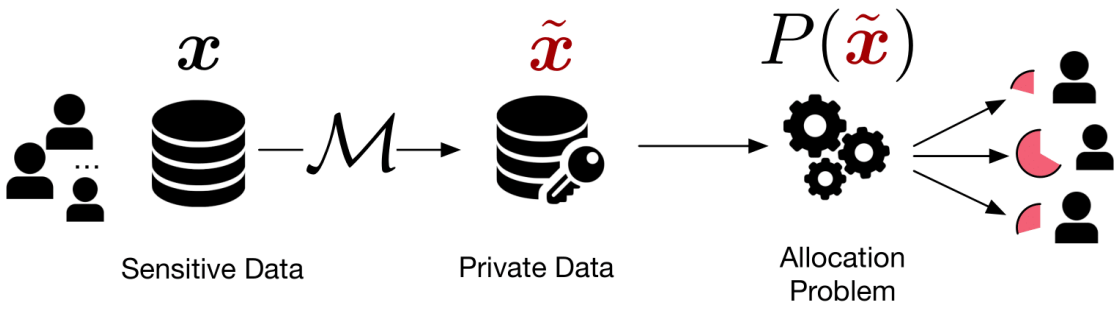


Fig. 2.1: Diagram of the private allocation problem.

the problem entities. In particular, the chapter focuses on measuring the bias of problem P_i

$$B_P^i(\mathcal{M}, \boldsymbol{x}) = \mathbb{E}_{\tilde{\boldsymbol{x}} \sim \mathcal{M}(\boldsymbol{x})} \left[P_i(\tilde{\boldsymbol{x}}) \right] - P_i(\boldsymbol{x}), \tag{2.2}$$

which characterizes the distance between the expected privacy-preserving allocation and the one based on the ground truth. The chapter considers the absolute bias $|B_P^i|$, in place of the bias B_P^i, when P is a decision rule. The distinction will become clear in the next sections.

The results in the chapter assume that $\mathcal{M}$, used to release counts, is the Laplace mechanism with an appropriate finite sensitivity Δ. *However, the results are general and apply to any data-release DP mechanism that add unbiased noise*.

2.4 Motivating problems

This section introduces two Census-motivated problem classes that grant benefits or privileges to groups of people.

Allotment problems The *Title I of the Elementary and Secondary Education Act of 1965* [117] distributes about $6.5 billion through basic grants. The federal allotment is divided among qualifying school districts in proportion to the count x_i of children aged 5 to 17 who live in necessitous

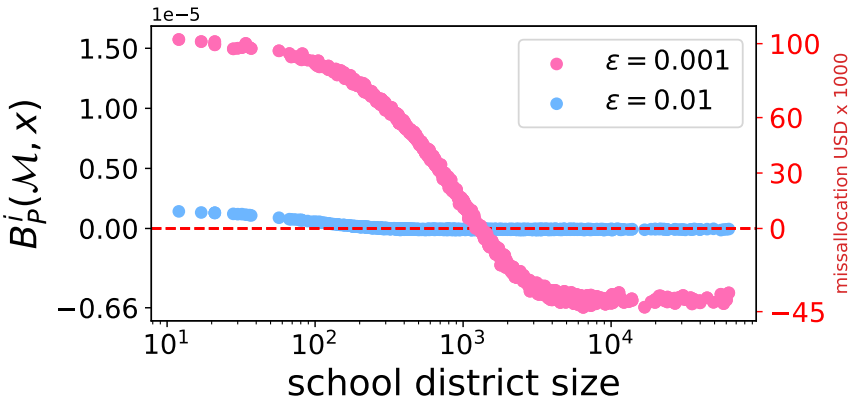


Fig. 2.2: Disproportionate Title 1 Funds Allocation in NY.

families in district i. The allocation is formalized by

$$P_i^F(\boldsymbol{x}) \stackrel{\text{def}}{=} \left(\frac{x_i \cdot a_i}{\sum_{i \in [n]} x_i \cdot a_i} \right),$$

where $\boldsymbol{x} = (x_i)_{i \in [n]}$ is the vector of all districts counts and a_i is a weight factor reflecting students expenditures.

Figure 2.2 illustrates the expected disparity errors arising when using private data as input to problem P^F, for various privacy losses ϵ. These errors are expressed in terms of bias (left y-axis) and USD misallocation (right y-axis) across the different New York school districts, ordered by their size. The allotments for small districts are typically overestimated while those for large districts are underestimated. Translated in economic factors, some school districts may receive up to 42,000 dollars less than warranted.

Decision Rules *Minority language voting right benefits* are granted to qualifying voting jurisdictions. The problem is formalized as

$$P_i^M(\boldsymbol{x}) \stackrel{\text{def}}{=} \left(\frac{x_i^{sp}}{x_i^s} > 0.05 \vee x_i^{sp} > 10^4 \right) \wedge \frac{x_i^{spe}}{x_i^{sp}} > 0.0131.$$

For a jurisdiction i, x_i^s, x_i^{sp}, and x_i^{spe} denote, respectively, the number of people in i speaking the minority language of interest, those that have also a limited English proficiency, and those that,

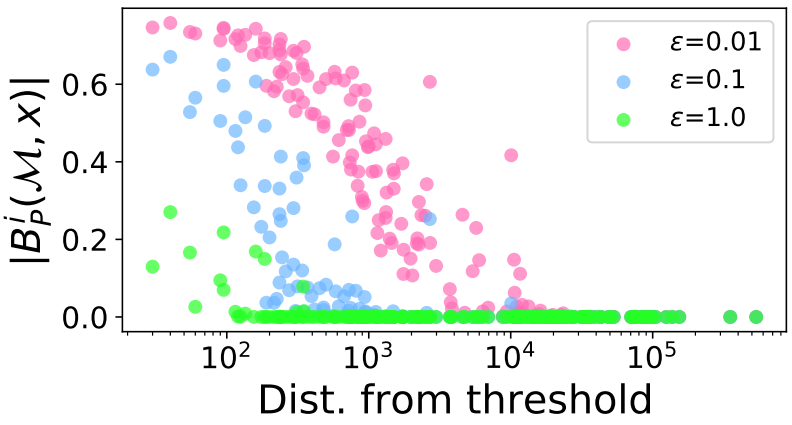


Fig. 2.3: Disproportionate Minority Language Voting Benefits.

in addition, have less than a 5^{th} grade education. Jurisdiction i must provide language assistance (including voter registration and ballots) *iff* $P_i^M(\boldsymbol{x})$ is *True*.

Figure 2.3 illustrates the decision error (y-axis), corresponding to the absolute bias $|B^i_{PM}(\mathcal{M}, \boldsymbol{x})|$, for sorted x_i^s, considering only true positives[1] for the *Hispanic* language. The figure shows that there are significant disparities in decision errors and that these errors strongly correlate to their distance to the thresholds. A similar issue was also observed in [71].

2.5 Fair allotments and decision rules

This section analyzes the fairness impact in allotment problems and decision rules. The adopted fairness concept captures the desire of equalizing the allocation errors among entities, which is of paramount importance given the critical societal and economic impact of the motivating applications.

Definition 2.2. *A data-release mechanism $\mathcal{M}$ is said fair w.r.t. a problem P if, for all datasets $\boldsymbol{x} \in \mathcal{X}$,*

$$B^i_P(\mathcal{M}, \boldsymbol{x}) = B^j_P(\mathcal{M}, \boldsymbol{x}) \quad \forall i, j \in [n].$$

That is, P does not induce disproportionate errors when taking as input a DP dataset generated by

[1] This is because misclassification, in this case, implies potentially disenfranchising a group of individuals.

$\mathcal{M}$. The chapter also introduces a notion to quantify and bound the mechanism unfairness.

Definition 2.3. *A mechanism $\mathcal{M}$ is said α-fair w.r.t. problem P if, for all datasets $\boldsymbol{x} \in \mathcal{X}$ and all $i \in [n]$,*

$$\xi_B^i(P, \mathcal{M}, \boldsymbol{x}) = \max_{j \in [n]} \left| B_P^i(\mathcal{M}, \boldsymbol{x}) - B_P^j(\mathcal{M}, \boldsymbol{x}) \right| \leq \alpha,$$

where ξ_B^i is referred to as the disparity error *of entity i.*

Parameter α is called the *fairness bound* and captures the fairness violation, with values close to 0 denoting strong fairness. A fair mechanism is also 0-fair.

Note that computing the fairness bound α analytically may not be feasible for some problem classes, since it may involve computing the expectation of complex functions P. Therefore, in the analytical assessments, the chapter recurs to a sampling approach to compute the *empirical expectation* $\hat{E}[P_i(\tilde{\boldsymbol{x}})] = \frac{1}{m} \sum_{j \in [m]} P_i(\tilde{\boldsymbol{x}}^j)$ in place of the true expectation in Equation (2.2). Therein, m is a sufficiently large sample size and $\tilde{\boldsymbol{x}}^j$ is the j-th outcome of the application of mechanism $\mathcal{M}$ on data set $\boldsymbol{x}$.

2.5.1 Fair Allotments: characterization

The first result characterizes a sufficient condition for the allotment problems to achieve finite fairness violations. The presentation uses $\boldsymbol{H}P_i$ to denote the Hessian of problem P_i, and $\mathrm{Tr}(\cdot)$ to denote the trace of a matrix. In this context, the Hessian entries are functions receiving a dataset as input. The presentation thus uses $(\boldsymbol{H}P_i)_{j,l}(\boldsymbol{x})$ and $\mathrm{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x})$ to denote the application of the second partial derivatives of P_i and of the *Hessian trace function* on dataset $\boldsymbol{x}$.

Theorem 2.1. *Let P be an allotment problem which is at least twice differentiable. A data-release mechanism $\mathcal{M}$ is α-fair w.r.t. P for some $\alpha < \infty$ if there exist some constant values c_{jl}^i $(i \in [n], j, l \in [k])$ such that, for all datasets $\boldsymbol{x} \in \mathcal{X}$,*

$$(\boldsymbol{H}P_i)_{j,l}(\boldsymbol{x}) = c_{j,l}^i \;\; (i \in [n] \; j, l \in [k]).$$

Corollary 2.1. *If P is a linear function, then $\mathcal{M}$ is fair w.r.t. P.*

Corollary 2.2. *$\mathcal{M}$ is fair w.r.t. P if there exists a constant c such that, for all dataset $\boldsymbol{x}$,*

$$\mathrm{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x}) = c \;\; (i \in [n]).$$

2.5.2 Fair decision rules: characterization

The next results bound the fairness violations of a class of indicator functions, called *thresholding functions*, and discusses the loss of fairness caused by the *composition of boolean predicates*, two recurrent features in decision rules. The fairness definition adopted uses the concept of absolute bias, in place of bias in Definition 2.3. Indeed, the absolute bias $|B_P^i|$ corresponds to the classification error for (binary) decision rules of P_i, i.e., $\Pr[P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})]$. The results also assume $\mathcal{M}$ to be a non-trivial mechanism, i.e., $|B_P^i(\mathcal{M}, \boldsymbol{x})| < 0.5 \, \forall i \in [n]$. Note that this is a non-restrictive condition, since the focus of data-release mechanisms is to preserve the quality of the original inputs, and the mechanisms considered in this chapter (and in the DP-literature, in general) all satisfy this assumption.

Theorem 2.2. *Consider a decision rule $P_i(\boldsymbol{x}) = \mathbf{1}\{x_i \geq \ell\}$ for some real value ℓ. Then, mechanism $\mathcal{M}$ is 0.5-fair w.r.t. P_i.*

This is a worst-case result and the mechanism may enjoy a better bound for specific datasets and decision rules. It is however significant since thresholding functions are ubiquitous in decision making over census data.

The next results focus on the composition of Boolean predicates under logical operators. The results are given under the assumption that mechanism $\mathcal{M}$ adds independent noise to the inputs of the predicates P_1 and P_2 to be composed, which is often the case. This assumption for P_1 and P_2 is denoted by $P^1 P^2$. Future work will aim at generalizing this results to broader assumptions.

Theorem 2.3. *Consider predicates P_1 and P_2 such that $P^1 P^2$ and assume that mechanism $\mathcal{M}$ that is α_k-fair for predicate P^k ($k \in \{1, 2\}$). Then $\mathcal{M}$ is α-fair for predicates $P^1 \vee P^2$ and $P^1 \wedge P^2$*

with

$$\alpha = \left(\alpha_1 + \underline{B}^1 + \alpha_2 + \underline{B}^2 - (\alpha_1 + \underline{B}^1)(\alpha_2 + \underline{B}^2) - \underline{B}^1 \underline{B}^2\right),$$

where $\overline{B}^k$ *and* $\underline{B}^k$ *are the maximum and minimum absolute biases for* $\mathcal{M}$ *w.r.t.* P^k *(for* $k = \{1, 2\}$*).*

The result above bounds the fairness violation derived by the composition of Boolean predicates under logical operators.

Theorem 2.4. *Consider predicates* P_1 *and* P_2 *such that* $P^1 \not\equiv P^2$ *and assume that mechanism* $\mathcal{M}$ *that is* α_k*-fair for predicate* P^k $(k \in \{1, 2\})$*. Then* $\mathcal{M}$ *is* α*-fair for* $P^1 \oplus P^2$ *with*

$$\alpha = \left(\alpha_1(1 - 2\underline{B}^2) + \alpha_2(1 - 2\underline{B}^1) - 2\alpha_1\alpha_2\right),$$

where $\underline{B}^k$ *is the minimum absolute bias for* $\mathcal{M}$ *w.r.t.* P^k *(*$k = \{1, 2\}$*).*

The following is a direct consequence of Theorem 2.9.

Corollary 2.3. *Assume that mechanism* $\mathcal{M}$ *is fair w.r.t. problems* P^1 *and* P^2*. Then* $\mathcal{M}$ *is also fair w.r.t.* $P^1 \oplus P^2$*.*

While XOR operator $\oplus$ is not adopted in the case studies considered in this chapter, it captures a surprising, positive compositional fairness result.

2.6 The nature of bias

The previous section characterized conditions bounding fairness violations. In contrast, this section analyzes the reasons for disparity errors arising in the motivating problems.

2.6.1 The problem structure

The first result is an important corollary of Theorem 2.6. It studies which restrictions on the structure of problem P are needed to satisfy fairness. Once again, P is assumed to be at least twice differentiable.

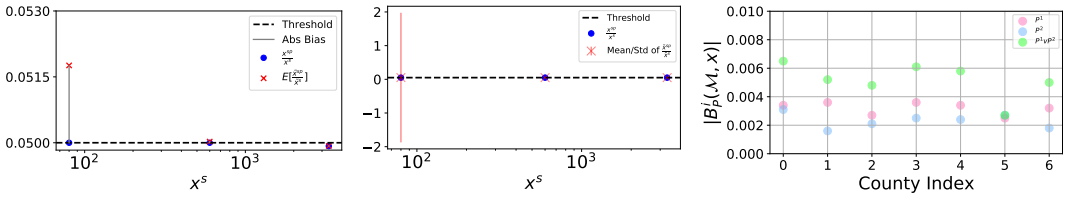


Fig. 2.4: Unfairness effect in *ratios* (left), *thresholding* (middle) and predicates disjunction (right)

Corollary 2.4. *Consider an allocation problem P. Mechanism $\mathcal{M}$ is not fair w.r.t. P if there exist two entries $i, j \in [n]$ such that $\mathrm{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x}) \neq \mathrm{Tr}(\boldsymbol{H}P_j)(\boldsymbol{x})$ for some dataset $\boldsymbol{x}$.*

The above implies that fairness cannot be achieved if P is *a non-convex function*, as is the case for *all* the allocation problems considered in this chapter. *A fundamental consequence of this result is the recognition that adding Laplacian noise to the inputs of the motivating example will necessarily introduce fairness issues.* For instance, consider P^F and notice that the trace of its Hessian

$$\mathrm{Tr}(\boldsymbol{H}P_i^F) = 2a_i \left[\frac{x_i \sum_{j\in[n]} a_j^2 - a_i \left(\sum_{j\in[n]} x_j a_j \right)}{\left(\sum_{j\in[n]} x_j a_j \right)^3} \right],$$

is not constant with respect to its inputs. Thus, any two entries i, j whose $x_i \neq x_j$ imply $\mathrm{Tr}(\boldsymbol{H}P_i^F) \neq \mathrm{Tr}(\boldsymbol{H}P_j^F)$. As illustrated in Figure 2.2, Problem P^F can introduce significant disparity errors. For $\epsilon = 0.001, 0.01$, and 0.1 the estimated fairness bounds are 0.003, 3×10^{-5}, and 1.2×10^{-6} respectively, which amount to an average misallocation of \$43,281, \$4,328, and \$865.6 respectively. The estimated fairness bounds were obtained by performing a linear search over all n school districts and selecting the maximal $\mathrm{Tr}(\boldsymbol{H}P_i^F)$.

Ratio Functions The next result considers *ratio functions* of the form $P_i(\langle x, y\rangle) = {}^{x}/_{y}$ with $x, y \in \mathbb{R}$ and $x \leq y$, which occur in the Minority language voting right benefits problem P_i^M. In the following $\mathcal{M}$ is the Laplace mechanism.

Corollary 2.5. *Mechanism $\mathcal{M}$ is not fair w.r.t. $P_i(\langle x, y\rangle) = {}^{x}/_{y}$ and inputs x, y.*

Figure 2.4 (left) provides an illustration linked to problem P^M. It shows the original values ${}^{x^{sp}}/_{x^s}$ (blue circles) and the expected values of the privacy-preserving counterparts (red crosses) of three

counties; from left to right: *Loving county, TX*, where $x^{sp}/x^s = 4/80 = 0.05$, *Terrell county, TX*, where $x^{sp}/x^s = 30/600 = 0.05$, and *Union county, NM*, where $x^{sp}/x^s = 160/3305 = 0.0484$. The length of the gray vertical line represents the absolute bias and the dotted line marks a threshold value (0.05) associated with the formula P_i^M. While the three counties have (almost) identical ratios values, they induce significant differences in absolute bias. This is due to the difference in scale of the numerator (and denominator), with smaller numerators inducing higher bias.

Thresholding Functions As discussed in Theorem 2.7, discontinuities caused by indicator functions, including thresholding, may induce unfairness. This is showcased in Figure 2.4 (center) which describes the same setting depicted in Figure 2.4 (left) but with the red line indicating the variance of the noisy ratios. Notice the significant differences in error variances, with Loving county exhibiting the largest variance. This aspect is also shown in Figure 2.3 where the counties with ratios lying near the threshold value have higher decisions errors than those whose ratios lies far from it.

2.6.2 Predicates composition

The next result highlights the negative impact coming from the composition of Boolean predicates. The following important result is corollary of Theorem 2.8 and provides a lower bound on the fairness bound.

Corollary 2.6. *Let mechanism $\mathcal{M}$ be α_k-fair w.r.t. to problem P^k ($k \in \{1, 2\}$). Then $\mathcal{M}$ is α-fair w.r.t. problems $P = P^1 \vee P^2$ and $P = P^1 \wedge P^2$, with $\alpha > \max(\alpha_1, \alpha_2)$.*

Figure 2.4 (right) illustrates Corollary 2.12. It once again uses the minority language problem P^M. In the figure, each dot represents the absolute bias $|B^i_{P^M}(\mathcal{M}, \boldsymbol{x})|$ associated with a selected county. Red and blue circles illustrate the absolute bias introduced by mechanism $\mathcal{M}$ for problem $P^1(x^{sp}) = \mathbb{1}\{x^{sp} \geq 10^4\}$ and $P^2(x^{sp}, x^{spe}) = \mathbb{1}\{\frac{x^{spe}}{x^{sp}} > 0.0131\}$ respectively. The selected counties have all similar and small absolute bias on the two predicates P^1 and P^2. However, when they

are combined using logical connector $\vee$, the resulting absolute bias increases substantially, as illustrated by the associated green circles.

The Section 2.9 (Section 2.1) also analyzes an interesting difference in errors based on the Truth values of the composing predicates P^1 and P^2, and shows that the highest error is achieved when they both are True for $\wedge$ and when they both are False for $\vee$ connectors. This may have strong implications in classification tasks.

2.6.3 Post-processing

The final analysis of bias relates to the effect of post-processing the output of the differentially private data release. In particular, the section focuses on ensuring non-negativity of the released data. The discussion focuses on problem P^F but the results are, once again, general. The Section 2.9 also include a discussion for other post-processing functions.

The section first presents a *negative result*: the application of a post-processing operator $\text{PP}^{\geq \ell}(z) \stackrel{\text{def}}{=} \max(\ell, z)$ to ensure that the result is at least ℓ induces a positive bias which, in turn, can exacerbate the disparity error of the allotment problem.

Theorem 2.5. *Let $\tilde{x} = x + (\lambda)$, with scale $\lambda > 0$, and $\hat{x} = PP^{\geq \ell}(\tilde{x})$, with $\ell < x$, be its post-processed value. Then,*

$$\mathbb{E}[\hat{x}] = x + \frac{\lambda}{2} \exp(\frac{\ell - x}{\lambda}).$$

Lemma 2.10 indicates the presence of positive bias of post-processed Laplace random variable when ensuring non-negativity, and that such bias is $B^i(\mathcal{M}, \boldsymbol{x}) = \mathbb{E}[\hat{x}_i] - x_i = \exp(\frac{\ell - x_i}{\lambda}) \leq \lambda/2$ for $\ell \leq x_i$. As shown in Figure 2.2 the effect of this bias has a negative impact on the final disparity of the allotment problem, where smaller entities have the largest bias (in the Figure $\ell = 0$).

Discussion The results highlighted in this section are both surprising and significant. They show that *the motivating allotment problems and decision rules induce inherent unfairness when given as input differentially private data*. This is remarkable since the resulting decisions have

significant societal, economic, and political impact on the involved individuals: federal funds, vaccines, and therapeutics may be unfairly allocated, minority language voters may be disenfranchised, and congressional apportionment may not be fairly reflected. The next section identifies a set of guidelines to mitigate these negative effects.

2.7 Mitigating solutions

2.7.1 The output perturbation approach

This section proposes three guidelines that may be adopted to mitigate the unfairness effects presented in the chapter, with focus on the motivating allotments problems and decision rules.

A simple approach to mitigate the fairness issues discussed is to recur to *output perturbation* to randomize the outputs of problem P_i, rather than its inputs, using an unbiased mechanism. Injecting noise directly after the computation of the outputs $P_i(\boldsymbol{x})$, ensures that the result will be unbiased. However, this approach has two shortcomings. First, it is not applicable to the context studied in this chapter, where a data agency desires to release a privacy-preserving data set $\tilde{\boldsymbol{x}}$ that may be used for various decision problems. Second, computing the sensitivity of the problem P_i may be hard, it may require to use a conservative estimate, or may even be impossible, if the problem has unbounded range. A conservative sensitivity implies the introduction of significant loss in accuracy, which may render the decisions unusable in practice.

2.7.2 Linearization by redundant releases

A different approach considers modifying on the decision problem P_i itself. Many decision rules and allotment problems are designed in an ad-hoc manner to satisfy some property on the original data, e.g., about the percentage of population required to have a certain level of education. Motivated by Corollaries 4.2 and 2.8, this section proposes guidelines to modify the original problem P_i with the goal of reducing the unfairness effects introduced by differential privacy.

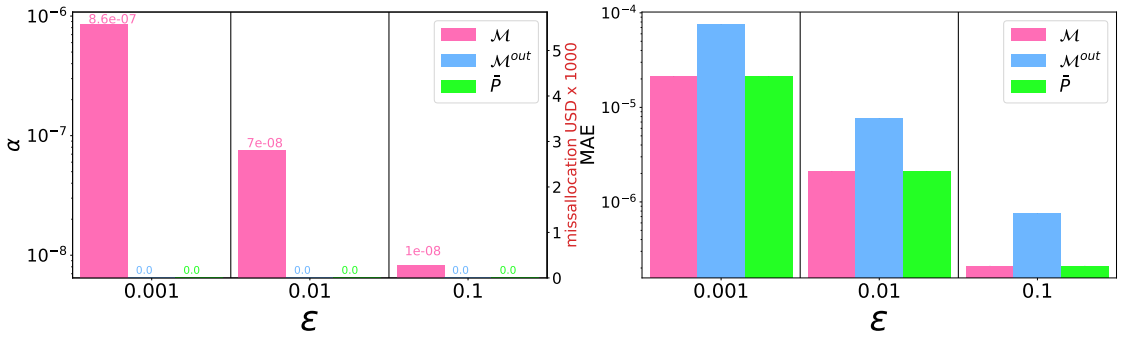


Fig. 2.5: Linearization by redundant release: Fairness and error comparison.

The idea is to use a linearized version $\bar{P}_i$ of problem P_i. While many linearizion techniques exists [102], and are often problem specific, the section focuses on a linear proxy $\bar{P}_i^F$ to problem P_i^F that can be obtained by enforcing a redundant data release. While the discussion focuses on problem P_i^F, the guideline is general and applies to any allotment problem with similar structure.

Let $Z = \sum_i a_i x_i$. Problem $P_i^F(\boldsymbol{x}) = {}^{a_i x_i}/_{Z}$ is linear w.r.t. the inputs x_i but non-linear w.r.t. Z. However, releasing Z, in addition to releasing the privacy-preserving values $\tilde{\boldsymbol{x}}$, would render Z a constant rather than a problem input to P^F. To do so, Z can either be released publicly, at cost of a (typically small) privacy leakage or by perturbing it with fixed noise. The resulting linear proxy allocation problem $\bar{P}_i^F$ is thus linear in the inputs $\boldsymbol{x}$.

Figure 2.5 illustrates this approach in practice. The left plot shows the fairness bound α and the right plot shows the empirical mean absolute error $\frac{1}{m}\sum_{k=1}^{m} |P_i(\boldsymbol{x}^k) - P_i(\tilde{\boldsymbol{x}}^m)|$, obtained using $m = 10^4$ repetitions, when the DP data $\tilde{\boldsymbol{x}}$ is applied to (1) the original problem P, (2) its linear proxy $\bar{P}$, and (3) when output perturbation (denoted $\mathcal{M}^{\text{out}}$) is adopted. The number on top of each bar reports the fairness bounds, and emphasize that the proposed remedy solutions achieve perfect-fairness. Notice that the proposed linear proxy solution can reduce the fairness violation dramatically while retaining similar errors.

While the output perturbation method reduces the disparity error, it also incurs significant errors that make the approach rarely usable in practice. The Section 2.9 (section 3.1) also discuss a solution based on a piecewise linear proxy function for the more complex decision rule P^M.

It is important to note that the experiments above use a data release mechanism $\mathcal{M}$ that applies

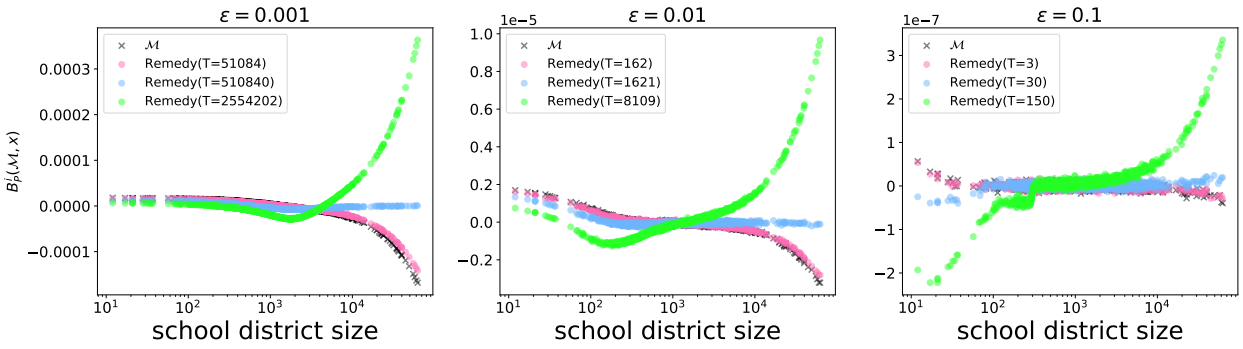


Fig. 2.6: Modified post-processing: Unfairness reduction.

no post-processing. A discussion about the mitigating solutions for the bias effects caused by post-processing is presented next.

2.7.3 Modified post-processing

This section introduces a simple, yet effective, solution to mitigate the negative fairness impact of the non-negative post-processing. The proposed solution operates in 3 steps: It first (1) performs a non-negative post-processing of the privacy-preserving input $\tilde{x}$ to obtain value $\bar{x} = \text{PP}^{\geq \ell}(\tilde{x})$. Next, (2) it computes $\bar{x}_T = \bar{x} - \frac{T}{\bar{x}+1-\ell}$. Its goal is to correct the error introduced by the post-processing operator, which is especially large for quantities near the boundary ℓ. Here T is a *temperature* parameter that controls the strengths of the correction. This step reduces the value $\bar{x}$ by quantity $\frac{T}{\bar{x}+1-\ell}$. The effect of this operation is to reduce the expected value $\mathbb{E}[\bar{x}]$ by larger (smaller) amounts as x get closer (farther) to the boundary value ℓ. Finally, (3) it ensures that the final estimate is indeed lower bounded by ℓ, by computing $\hat{x} = \max(\bar{x}_T, \ell)$.

The benefits of this approach are illustrated in Figure 2.6, which show the absolute bias $|B^i_{PF}|$ for the Title 1 fund allocation problem that is induced by the original mechanism $\mathcal{M}$ with standard post-processing $\text{PP}^{\geq 0}$ and by the proposed modified post-processing for different temperature values T. The figure illustrates the role of the temperature T in the disparity errors. Small values T may have small impacts in reducing the disparity errors, while large T values can introduce errors,

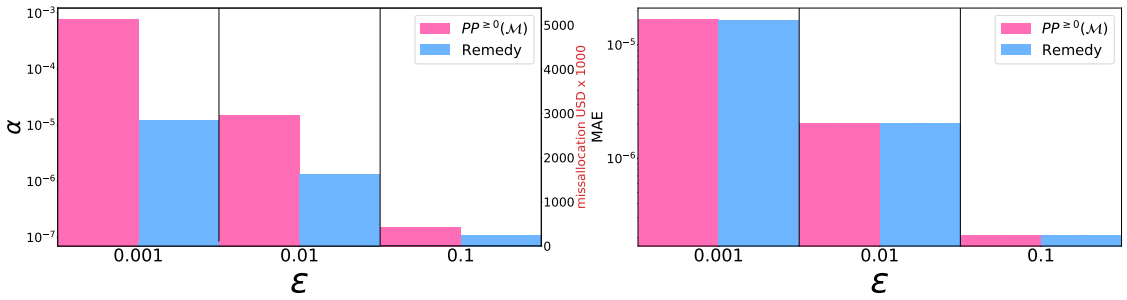


Fig. 2.7: Modified post-processing on problem P^F.

thus may exacerbate unfairness. The optimal choice for T can be found by solving the following:

$$T^* = \operatorname*{argmin}_T \left(\max_{\boldsymbol{x} \geq \ell} |\mathbb{E}[\hat{\boldsymbol{x}}_T] - \boldsymbol{x}| - \min_{\boldsymbol{x} \geq \ell} |\mathbb{E}[\hat{\boldsymbol{x}}_T] - \boldsymbol{x}| \right), \tag{2.3}$$

where $\hat{\boldsymbol{x}}_T$ is a random variable obtained by the proposed 3 step solution, with temperature T. The expected value of $\hat{\boldsymbol{x}}$ can be approximated via sampling. Note that naively finding the optimal T may require access to the true data. Solving the problem above in a privacy-preserving way is beyond the scope of the chapter and the subject of future work.

The reductions in the fairness bound α for problem P^F are reported in Figure 2.7 (left), while Figure 2.7 (right) shows that this method has no perceptible impact on the mean absolute error. Once again, these errors are computed via sampling and use 10^4 samples.

2.7.4 Fairness payment

Finally, this section focuses on allotment problems, like P^F, that distribute a budget B among n entities, and where the allotment for entity i represents the fraction of budget B it expects. Differential privacy typically implements a postprocessing step to renormalize the fractions so that they sum to 1. This normalization, together with nonnegativity constraints, introduces a bias and hence more unfairness. One way to alleviate this problem is to increase the total budget B, and avoiding the normalization. This section quantifies the cost of doing so: it defines the *cost of privacy*, which is the increase in budget B^+ required to achieve this goal.

Definition 2.4 (Cost of Privacy)**.** *Given problem P, that distributes budget B among n entities,*

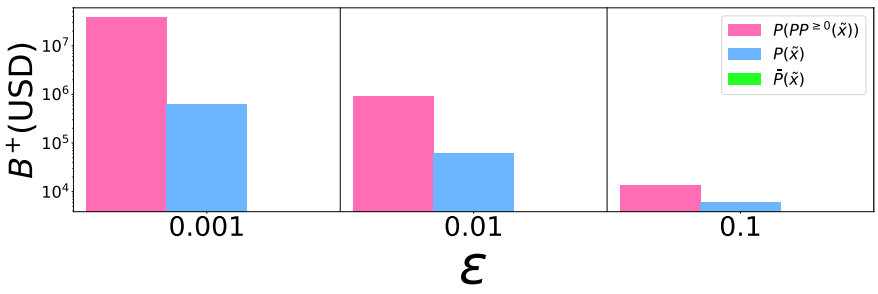


Fig. 2.8: Cost of privacy on problem P^F.

data release mechanism $\mathcal{M}$, and dataset $\boldsymbol{x}$, the cost of privacy is:

$$B^+ = \sum_{i \in I^-} |B_P^i(\mathcal{M}, \boldsymbol{x})| \times B$$

with $I^- = \{i \,:\, B_P^i(\mathcal{M}, \boldsymbol{x}) < 0\}$.

Figure 2.8 illustrates the cost of privacy, in USD, required to render each county in the state of New York not negatively penalized by the effects of differential privacy. The figure shows, in decreasing order, the different costs associated with a mechanism $P^F(\text{PP}^{\geq 0}(\boldsymbol{x}))$ that applies a post-processing step, one $P^F(\boldsymbol{x})$ that does not apply post-processing, and one that uses a linear proxy problem $\bar{P}^F(\boldsymbol{x})$.

2.8 Conclusions

This chapter analyzed the disparity arising in decisions granting benefits or privileges to groups of people when these decisions are made adopting differentially private statistics about these groups.

It first characterized the conditions for which allotment problems achieve finite fairness violations and bound the fairness violations induced by important components of decision rules, including reasoning about the composition of Boolean predicates under logical operators. Then, the chapter analyzed the reasons for disparity errors arising in the motivating problems and recognized the problem structure, the predicate composition, and the mechanism post-processing, as paramount to the bias and unfairness contribution. Finally, it suggested guidelines to act on the

decision problems or on the mechanism (i.e., via modified post-processing steps) to mitigate the unfairness issues. The analysis provided in this chapter may provide useful guidelines for policy-makers and data agencies for testing the fairness and bias impacts of privacy-preserving decision making.

2.9 Appendix

This appendix is organized as follows. Section 2.9.1 provides all the missing proofs of the theoretical results that were presented in the chapter. Section 2.9.2 extends Section 2.6 in the main chapter to discuss additional post-processing steps that may introduce bias. Section 2.9.3 extends Section 2.7 of the main chapter and discusses an interesting piece-wise linear proxy solution and applies it to the *Minority language voting right benefits* decision rule P^M.

Section 2.9.4 discusses the experimental setting adopted by the chapter in the reported results. Finally, section 2.9.5 discusses existing research in the scope of differential privacy and fairness. While our contribution is unique, in that it characterizes the fairness is-sues arising in decision making problems that take as input differential private data, section 2.9.5 covers the work that has looked at the tradeoff between differential privacy and some notion of group fairness in machine learning.

2.9.1 Missing proofs

Theorem 2.6. *Let P be an allotment problem which is at least twice differentiable. A data-release mechanism $\mathcal{M}$ is α-fair w.r.t. P for some $\alpha < \infty$ if there exist some constant values c^i_{jl} $(i \in [n], j, l \in [k])$ such that, for all datasets $\boldsymbol{x} \in \mathcal{X}$,*

$$(\boldsymbol{H}P_i)_{j,l}(\boldsymbol{x}) = c^i_{j,l} \;\; (i \in [n]\; j, l \in [k])$$

Proof. For a given entity i we would like the disparity error to be bounded by a finite value α:

$$\begin{aligned} & \xi_B^i(P, \mathcal{M}, \boldsymbol{x}) \leq \alpha \\ \Leftrightarrow\ & |B_P^i(\mathcal{M}, \boldsymbol{x}) - B_P^j(\mathcal{M}, \boldsymbol{x})| \leq \alpha \ \ \forall i, j \in [n] \\ \Leftrightarrow\ & |P_i(\boldsymbol{x}) - \mathbb{E}[P_i(\tilde{\boldsymbol{x}})] - P_j(\boldsymbol{x}) - \mathbb{E}[P_j(\tilde{\boldsymbol{x}})]| \ \ \forall i, j \in [n]. \end{aligned} \tag{2.4}$$

Since, by assumption, P_i is at least twice differentiable, with constant second derivatives, a Taylor expansion implies that:

$$P_i(\tilde{\boldsymbol{x}}) = P_i(\boldsymbol{x} + \eta) = P_i(\boldsymbol{x}) + \eta^T \nabla P_i(\boldsymbol{x}) + \frac{1}{2}\eta^T \boldsymbol{H} P_i(\boldsymbol{x}) \tag{2.5}$$

where $\eta = (\Delta/\epsilon)$ and $\boldsymbol{H}$ is the Hessian of function P_i.

From the above, taking the expectation on both size, $\mathbb{E}[P_i(\boldsymbol{x})]$ can be rewritten as

$$\begin{aligned} & \mathbb{E}[P_i(\tilde{\boldsymbol{x}})] \\ & \quad = \mathbb{E}[P_i(\boldsymbol{x})] + \mathbb{E}[\eta^T \nabla P_i(\boldsymbol{x})] + \mathbb{E}[\frac{1}{2}\eta^T \boldsymbol{H} P_i(\boldsymbol{x})\eta] && (2.6) \\ & \quad = P_i(\boldsymbol{x}) + \mathbb{E}[\eta^T \nabla P_i(\boldsymbol{x})] + \mathbb{E}[\frac{1}{2}\eta^T \boldsymbol{H} P_i(\boldsymbol{x})\eta] && (2.7) \\ & \quad = P_i(\boldsymbol{x}) + \mathbb{E}[\eta^T]\mathbb{E}[\nabla P_i(\boldsymbol{x})] + \mathbb{E}[\frac{1}{2}\eta^T \boldsymbol{H} P_i(\boldsymbol{x})\eta] && (2.8) \\ & \quad = P_i(\boldsymbol{x}) + 0 + \mathbb{E}[\frac{1}{2}\eta^T \boldsymbol{H} P_i(\boldsymbol{x})\eta] && (2.9) \\ & \quad = P_i(\boldsymbol{x}) + \mathbb{E}[\frac{1}{2}\sum_{jk\in[n]} \eta_j (\boldsymbol{H} P_i)_{jk}(\boldsymbol{x})\eta_k] && (2.10) \\ & \quad = P_i(\boldsymbol{x}) + \frac{1}{2}\mathbb{E}[\sum_{j\in[n]} \eta_j^2 (\boldsymbol{H} P_i)_{jj}(\boldsymbol{x})] && (2.11) \\ & \quad = P_i(\boldsymbol{x}) + \frac{1}{2}\mathbb{E}[\sum_{j\in[n]} \eta_j^2]\mathbb{E}[\sum_{j\in[n]} (\boldsymbol{H} P_i)_{jj}(\boldsymbol{x})] && (2.12) \\ & \quad = P_i(\boldsymbol{x}) + \frac{\Delta^2}{\epsilon^2}\mathbb{E}[\sum_{j\in[n]} (\boldsymbol{H} P_i)_{jj}(\boldsymbol{x})] && (2.13) \\ & \quad = P_i(\boldsymbol{x}) + \frac{\Delta^2}{\epsilon^2} \cdot \mathrm{Tr}(\boldsymbol{H} P_i)(\boldsymbol{x}) && (2.14) \end{aligned}$$

where (2.6) follows by linearity of expectation, (2.7) by $P_i(\boldsymbol{x})$ not being random, (2.8) by independence of η^T and $\nabla P_i(\boldsymbol{x})$, (2.9) since $\mathbb{E}[\eta^T] = \mathbf{0}^T$, by definition of the Laplace mechanism, (2.11) since η^T is a vector of independent noise, and thus $\mathbb{E}[\eta_j \eta_k] = 0$ for all $j \neq k$, (2.12) by linearity of expectation, (2.13) since $\mathbb{E}[\eta_j^2] = {}^{\Delta^2}\!/_{\epsilon^2}$, and where Tr in (2.14) denotes the trace of the Hessian matrix.

From (2.14) and (2.5), Equation (2.4) can be rewritten as:

$$\begin{aligned}
&\big|P_i(\boldsymbol{x}) - P_i(\boldsymbol{x}) + \frac{\Delta^2}{\epsilon^2}\operatorname{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x}) \\
&\qquad - P_j(\boldsymbol{x}) - P_j(\boldsymbol{x}) + \frac{\Delta^2}{\epsilon^2}\operatorname{Tr}(\boldsymbol{H}P_j)(\boldsymbol{x})\big| \\
&= \big|\frac{\Delta^2}{\epsilon^2}\operatorname{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x}) - \frac{\Delta^2}{\epsilon^2}\operatorname{Tr}(\boldsymbol{H}P_j)(\boldsymbol{x})\big| \\
&= \frac{\Delta^2}{\epsilon^2}\big|\operatorname{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x}) - \operatorname{Tr}(\boldsymbol{H}P_j)(\boldsymbol{x})\big|. \qquad (2.15)
\end{aligned}$$

Since, by assumption, there exists constants c_k such that $\forall x \in \mathcal{X}$, $\operatorname{Tr}(\boldsymbol{H}P_k)(\boldsymbol{x}) = \sum_{j,l} c_{j,l}^k = c_k$ for $k \in [n]$, it follows, that, from Equation (2.15),

$$\frac{\Delta^2}{\epsilon^2}|c_i - c_j| < \frac{\Delta^2}{\epsilon^2}\big(\max_{i\in[n]} c_i - \min_{i\in[n]} c_i\big) < \infty, \qquad (2.16)$$

which concludes the proof. □

Corollary 2.7. *If P be a linear function, then $\mathcal{M}$ is fair w.r.t. P.*

Proof. The result follows by noticing that the second derivative of linear function is 0 for any input. Thus, for any $i \in [n]$, and $\boldsymbol{x} \in \mathcal{X}$,

$$\operatorname{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x}) = 0.$$

Therefore, from (2.15), for every $i \in [n]$,

$$\xi_B^i(P, \mathcal{M}, \boldsymbol{x}) = \max_{j\in[n]} |\operatorname{Tr}(\boldsymbol{H}P_i(\boldsymbol{x}) - \operatorname{Tr}(\boldsymbol{H}P_j(\boldsymbol{x}))| = 0,$$

which proves the claim. □

A more general result is the following.

Corollary 2.8. *$\mathcal{M}$ is fair w.r.t. P if there exists a constant c such that, for all dataset $\boldsymbol{x}$,*

$$\mathrm{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x}) = c \;\; (i \in [n]).$$

The proof is similar in spirit to proof of Corollary 4.2, noting that, in the above, the constant c is equal among all Traces of the Hessian of problems P_i $(i \in [n])$.

Corollary 1 and Corollary 2 are completely fine

Theorem 2.7. *Consider a decision rule $P_i(\boldsymbol{x}) = \mathbb{1}\{x_i \geq \ \ell\}$ for some real value ℓ. Then, mechanism $\mathcal{M}$ is 0.5-fair w.r.t. P_i.*

Proof. From Definition 3 of the main text that uses the absolute bias $|B_P^i|$, and since the absolute bias is always non-negative, it follows that, for every $i \in [n]$:

$$\begin{aligned}
\xi_B^i(P, \mathcal{M}, \boldsymbol{x}) &= \max_{j \in [n]} \big| |B_P^i| - |B_P^j| \big| && (2.17)\\
&\leq \max_{j \in [n]} |B_P^j| - \min_{j \in [n]} |B_P^j| && (2.18)\\
&\leq \max_{j \in [n]} |B_P^j|. && (2.19)
\end{aligned}$$

Thus, by definition, mechanism $\mathcal{M}$ is $\max_{j \in [n]} |B_j^P|$-fair w.r.t. problem P. The following shows that the maximum absolute bias $\max_{j \in [n]} |B_j^P| \leq 0.5$. Let i be the entry with the largest absolute bias. W.l.o.g. consider the case in which $P_i(\boldsymbol{x}) = \textit{True}$ (the other case is symmetric). It follows that,

$$\begin{aligned}
|B_P^i(\mathcal{M}, \boldsymbol{x})| &= |P_i(\boldsymbol{x}) - \mathbb{E}_{\tilde{\boldsymbol{x}}_i \sim \mathcal{M}(\boldsymbol{x})}[P_i(\tilde{\boldsymbol{x}})]| \\
&= |1 - \Pr(\tilde{x}_i \geq \ell)| \\
&= |1 - \Pr(\eta \geq \ell - x_i)|,
\end{aligned}$$

where $\eta \sim \text{Lap}(0, \frac{\Delta}{\epsilon})$. Notice that,

$$\Pr(\eta \geq \ell - x_i) \geq \Pr(\eta \geq 0) = 0.5$$

since $\ell - x_i \leq 0$, by case assumption (i.e., $P_i(\boldsymbol{x}) = \textit{True}$ implies that $x_i \geq \ell$) and by that the mechanism considered adds 0-mean symmetric noise. Thus, $|B_P^i(\mathcal{M}, \boldsymbol{x})| \leq 0.5$, and since, i is the entity contributing maximum absolute bias among all other entities in $[n]$, it follows that

$$\max_{j \in [n]} |B_P^j(\mathcal{M}, \boldsymbol{x})| \leq 0.5$$

and thus, for every $i \in [n]$, $\xi_B^i(P, \mathcal{M}, \boldsymbol{x}) \leq 0.5$, and, therefore, $\mathcal{M}$ is 0.5-fair. □

Next, the chapter proves Theorems 3 and 4. Recall that, these results hold under the assumption that mechanism $\mathcal{M}$ adds independent noise to the inputs of the predicates P_1 and P_2 to be composed, which is often the case.

The chapter first discusses the following property and lemmas.

Property 2.1. *The following three bivariate functions:*

- $f(a, b) = ab$
- $f(a, b) = a + b - ab$
- $f(a, b) = a + b - 2ab$

with support $[0, 0.5]$ *and range* $\mathcal{R}$ *all are monotonically increasing functions on its domain* $a, b \in (0, 0.5)$

The property above can be shown by noticing that for all functions and support considered, their derivatives w.r.t. their inputs are positive.

Lemma 2.1. *Consider predicates* P_i^1 *and* P_i^2 *and let* $P = P_i^1 \wedge P_i^2$*, then, for any dataset* $\boldsymbol{x} \in \mathcal{X}$*,*

label=() $P_i^1(\boldsymbol{x}) = 0 \wedge P_i^2(\boldsymbol{x}) = 0 \Rightarrow$

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = |B^i_{P_i^1}||B^i_{P_i^2}|$$

lbbel=() $P_i^1(\boldsymbol{x}) = 0 \wedge P_i^2(\boldsymbol{x}) = 1 \Rightarrow$

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = |B^i_{P_i^1}|(1 - |B^i_{P_i^2}|)$$

lcbel=() $P_i^1(\boldsymbol{x}) = 1 \wedge P_i^2(\boldsymbol{x}) = 0 \Rightarrow$

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = (1 - |B^i_{P_i^1}|)|B^i_{P_i^2}|$$

ldbel=() $P_i^1(\boldsymbol{x}) = 1 \wedge P_i^2(\boldsymbol{x}) = 1 \Rightarrow$

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = |B^i_{P^1}| + |B^i_{P^2}| - |B^i_{P^1}||B^i_{P^2}|$$

where $\tilde{\boldsymbol{x}} = \mathcal{M}(\boldsymbol{x})$ is the privacy-preserving dataset.

Proof. The proof proceeds by cases. Case (i): $P_i^1(\boldsymbol{x}) = 0 \wedge P_i^2(\boldsymbol{x}) = 0$, therefore $P_i(\boldsymbol{x}) = P_i^1(\boldsymbol{x}) \wedge P_i^2(\boldsymbol{x}) = 0 \wedge 0 = 0$. Hence:

$$\begin{aligned}
&\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \wedge P_i^2(\tilde{\boldsymbol{x}})) \neq 0\right] \\
&= \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \wedge P_i^2(\tilde{\boldsymbol{x}})) = 1\right] \\
&= \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) = 1) \wedge (P_i^2(\tilde{\boldsymbol{x}}) = 1))\right] \\
&= \Pr(P_i^1(\tilde{\boldsymbol{x}}) = 1)) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) = 1)) \\
&= \Pr(P_i^1(\tilde{\boldsymbol{x}}) \neq P_i^1(\boldsymbol{x}))) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) \neq P_i^2(\boldsymbol{x})) \\
&= |B^i_{P^1}||B^i_{P^2}|
\end{aligned} \tag{2.20}$$

Where the fourth equality is due to $P_i^1 P_i^2$.

Case (ii): $P_i^1(\boldsymbol{x}) = 0 \wedge P_i^2(\boldsymbol{x}) = 1$, therefore $P_i(\boldsymbol{x}) = P_i^1(\boldsymbol{x}) \wedge P_i^2(\boldsymbol{x}) = 0 \wedge 1 = 0$. Hence:

$$\begin{aligned}
&\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \wedge P_i^2(\tilde{\boldsymbol{x}})) \neq 0\right] \\
&= \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \wedge P_i^2(\tilde{\boldsymbol{x}})) = 1\right] \\
&= \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) = 1) \wedge (P_i^2(\tilde{\boldsymbol{x}}) = 1))\right] \\
&= \Pr(P_i^1(\tilde{\boldsymbol{x}}) = 1)) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) = 1)) \\
&= \Pr(P_i^1(\tilde{\boldsymbol{x}}) \neq P_i^1(\boldsymbol{x}))) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) = P_i^2(\boldsymbol{x})) \\
&= \Pr(P_i^1(\tilde{\boldsymbol{x}}) \neq P_i^1(\boldsymbol{x}))) \cdot \left(1 - \Pr(P_i^2(\tilde{\boldsymbol{x}}) \neq P_i^2(\boldsymbol{x}))\right) \\
&= |B_{P^1}^i|(1 - |B_{P^2}^i|)
\end{aligned} \tag{2.21}$$

Where the fourth equality is due to $P_i^1 P_i^2$.

Case (iii): $P_i^1(\boldsymbol{x}) = 1 \wedge P_i^2(\boldsymbol{x}) = 0$, therefore $P_i(\boldsymbol{x})) = P_i^1(\boldsymbol{x}) \wedge P_i^2(\boldsymbol{x}) = 1 \wedge 0 = 0$. Hence:

$$\begin{aligned}
&\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \wedge P_i^2(\tilde{\boldsymbol{x}})) \neq 0\right] \\
&= \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \wedge P_i^2(\tilde{\boldsymbol{x}})) = 1\right] \\
&= \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) = 1) \wedge (P_i^2(\tilde{\boldsymbol{x}}) = 1))\right] \\
&= \Pr(P_i^1(\tilde{\boldsymbol{x}}) = 1)) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) = 1)) \\
&= \Pr(P_i^1(\tilde{\boldsymbol{x}}) = P_i^1(\boldsymbol{x}))) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) \neq P_i^2(\boldsymbol{x})) \\
&= \left(1 - \Pr(P_i^1(\tilde{\boldsymbol{x}}) \neq P_i^1(\boldsymbol{x})))\right) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) \neq P_i^2(\boldsymbol{x})) \\
&= (1 - |B_{P^1}^i|)|B_{P^2}^i|
\end{aligned} \tag{2.22}$$

Where the fourth equality is due to $P_i^1 P_i^2$.

Case (iv): $P_i^1(\boldsymbol{x}) = 1 \wedge P_i^2(\boldsymbol{x}) = 1$, therefore $P_i(\boldsymbol{x}) = P_i^1(\boldsymbol{x}) \wedge P_i^2(\boldsymbol{x}) = 1 \wedge 1 = 1$. Hence:

$$\begin{aligned}
&\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \wedge P_i^2(\tilde{\boldsymbol{x}})) \neq 1\right] \\
&= \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \wedge P_i^2(\tilde{\boldsymbol{x}})) = 0\right] \\
&= 1 - \Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) = 1) \wedge (P_i^2(\tilde{\boldsymbol{x}}) = 1))\right] \\
&= 1 - \Pr(P_i^1(\tilde{\boldsymbol{x}}) = 1)) \Pr(P_i^2(\tilde{\boldsymbol{x}}) = 1)) \\
&= 1 - \big(1 - \Pr(P_i^1(\tilde{\boldsymbol{x}}) \neq P_i^1(\boldsymbol{x}))\big)\big(1 - \Pr(P_i^2(\tilde{\boldsymbol{x}}) \neq P_i^2(\boldsymbol{x}))\big) \\
&= 1 - (1 - |B_{P^1}^i|)(1 - |B_{P^2}^i|) \\
&= |B_{P^1}^i| + |B_{P^2}^i| - |B_{P^1}^i||B_{P^2}^i|
\end{aligned} \tag{2.23}$$

Where the fourth equality is due to $P_i^1 P_i^2$. □

Lemma 2.2. *Consider predicates P_i^1 and P_i^2 and let $P = P_i^1 \vee P_i^2$, then, for any dataset $\boldsymbol{x} \in \mathcal{X}$,*

label=() *If $P_i^1(\boldsymbol{x}) = 0, P_i^2(\boldsymbol{x}) = 0$ then*

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = |B_{P^1}^i| + |B_{P^2}^i| - |B_{P^1}^i||B_{P^2}^i|$$

lbbel=() *If $P_i^1(\boldsymbol{x}) = 0, P_i^2(\boldsymbol{x}) = 1$ then*

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = (1 - |B_{P^1}^i|)|B_{P^2}^i|$$

lcbel=() *If $P_i^1(\boldsymbol{x}) = 1, P_i^2(\boldsymbol{x}) = 0$ then*

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = |B_{P^1}^i|(1 - |B_{P^2}^i|)$$

ldbel=() *If $P_i^1(\boldsymbol{x}) = 1, P_i^2(\boldsymbol{x}) = 1$ then*

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = |B_{P^1}^i||B_{P^2}^i|$$

where $\tilde{\boldsymbol{x}} = \mathcal{M}(\boldsymbol{x})$ is the privacy-preserving dataset.

The proof is similar to proof of Lemma 2.1.

Lemma 2.3. *Given $P(\boldsymbol{x}) = P^1(\boldsymbol{x}) \oplus P^2(\boldsymbol{x})$, then for any value of $P_i^1(\boldsymbol{x}), P_i^2(\boldsymbol{x}) \in \{0, 1\}$:*

$$\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = |B_{P^1}^i| + |B_{P^2}^i| - 2|B_{P^1}^i||B_{P^2}^i|.$$

Proof. The following hold for all four combination of binary boolean values for $P_i^1(\boldsymbol{x})_i, P_i^2(\boldsymbol{x}) \in \{0,1\}$:

$$\begin{aligned}
&\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) \\
=&\Pr\left[(P_i^1(\tilde{\boldsymbol{x}}) \oplus P_i^2(\tilde{\boldsymbol{x}})) \neq (P_i^1(\boldsymbol{x}) \oplus P_i^2(\boldsymbol{x}))\right] \\
=&1 - \Pr(P_i^1(\tilde{\boldsymbol{x}}) = P_i^1(\boldsymbol{x})) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) = P_i^2(\boldsymbol{x})) \\
&- \Pr(P_i^1(\tilde{\boldsymbol{x}}) \neq P_i^1(\boldsymbol{x})) \cdot \Pr(P_i^2(\tilde{\boldsymbol{x}}) \neq P_i^2(\boldsymbol{x})) \\
=&1 - |B^i_{P^1}||B^i_{P^2}| - (1 - |B^i_{P^1}|)(1 - |B^i_{P^2}|) \\
=&|B^i_{P^1}| + |B^i_{P^2}| - 2|B^i_{P^1}||B^i_{P^2}|.
\end{aligned}$$

Where the second equality is due to $P_i^1 P_i^2$. □

Theorem 2.8. *Consider predicates P_1 and P_2 such that $P^1 P^2$ and assume that mechanism $\mathcal{M}$ that is α_k-fair for predicate P^k ($k \in \{1,2\}$). Then $\mathcal{M}$ is α-fair for predicates $P^1 \vee P^2$ and $P^1 \wedge P^2$ with*

$$\alpha = \left(\alpha_1 + \underline{B}^1 + \alpha_2 + \underline{B}^2 - (\alpha_1 + \underline{B}^1)(\alpha_2 + \underline{B}^2) - \underline{B}^1 \underline{B}^2\right),$$

where $\overline{B}^k$ and $\underline{B}^k$ are the maximum and minimum absolute biases for $\mathcal{M}$ w.r.t. P^k (for $k = \{1,2\}$).

The proof focuses on the case $P^1 \wedge P^2$ while the proof for the disjunction is similar.

Proof. To prove the statement, first notice that, by Lemma 2.1 and by assumption of $\mathcal{M}$ being non-trivial, it follows that

$$\begin{aligned}
&|B^i_{P^1}||B^i_{P^2}| < |B^i_{P^1}|(1 - |B^i_{P^2}|), \\
&|B^i_{P^2}|(1 - |B^i_{P^1}|) < |B^i_{P^1}| + |B^i_{P^2}| - |B^i_{P^1}||B^i_{P^2}|.
\end{aligned} \tag{2.24}$$

due to that $0 \leq |B^i_{P^1}| \leq 0.5$ and $0 \leq |B^i_{P^2}| \leq 0.5$, and thus:

$$|B^i_{P^1}||B^i_{P^2}| \leq \Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) \tag{2.25}$$

$$\leq |B^i_{P^1}| + |B^i_{P^2}| - |B^i_{P^1}||B^i_{P^2}|, \tag{2.26}$$

From the above, the maximum absolute bias $\overline{B}_P$ can be upper bounded as:

$$\overline{B}_P = \max_i \Pr(P_i(\tilde{x}) \neq P_i(x)) \tag{2.27}$$

$$\leq \max_i |B^i_{P^1}| + |B^i_{P^2}| - |B^i_{P^1}||B^i_{P^2}| \tag{2.28}$$

$$= \overline{B^1} + \overline{B^2} - \overline{B^1 B^2}, \tag{2.29}$$

where the first inequality follows by Lemma 2.1 and the last equality follows by Property 2.1.

Similarly, the minimum absolute bias of $\underline{B}_P$ can be lower bounded as:

$$\underline{B}_P = \min_i \Pr(P_i(\tilde{x}) \neq P_i(x)) \tag{2.30}$$

$$\geq \min_i |B^i_{P^1}||B^i_{P^2}| = \underline{B^1 B^2} \tag{2.31}$$

where the first inequality is due to Lemma 2.1, and the last equality is due to Property 2.1. Hence, the level of unfairness α of problem P can be determined by:

$$\alpha = \overline{B}_P - \underline{B}_P \leq \overline{B^1} + \overline{B^2} - \overline{B^1 B^2} - \underline{B^1 B^2} \tag{2.32}$$

Substituting $\overline{B^1} = \alpha_1 + \underline{B^1}$ and $\overline{B^2} = \alpha_2 + \underline{B^2}$ into the Equation (2.32), gives the sought fairness bound. □

Theorem 2.9. *Consider predicates P_1 and P_2 such that $P^1 \mathbb{P}^2$ and assume that mechanism $\mathcal{M}$ that is α_k-fair for predicate P^k $(k \in \{1, 2\})$. Then $\mathcal{M}$ is α-fair for predicates $P^1 \oplus P^2$ with*

$$\alpha = \big(\alpha_1(1 - 2\underline{B}^2) + \alpha_2(1 - 2\underline{B}^1) - 2\alpha_1\alpha_2\big),$$

where $\underline{B}^k$ is the minimum absolute bias for $\mathcal{M}$ w.r.t. P^k (for $k = \{1, 2\}$).

Proof. To prove the statement, first, notice that the maximum absolute bias for $\mathcal{M}$ w.r.t. $P = P^1 \oplus P^2$ can be expressed as:

$$\begin{aligned}
&\max_i \Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) \\
&= \max_{|B^i_{P^1}|,|B^i_{P^2}|} |B^i_{P^1}| + |B^i_{P^2}| - 2|B^i_{P^1}||B^i_{P^2}| && (2.33) \\
&= \overline{B}^1 + \overline{B}^2 - 2\overline{B}^1\overline{B}^2, && (2.34)
\end{aligned}$$

where the first equality is due to Lemma 2.3, and the second due to Property 2.1.

Next, notice that the minimum absolute bias for $\mathcal{M}$ w.r.t. $P = P^1 \oplus P^2$ can be expressed as:

$$\begin{aligned}
&\min_i \Pr(P_i(\tilde{x}) \neq P_i(x)) \\
&= \min_{|B^i_{P^1}|,|B^i_{P^2}|} |B^i_{P^1}| + |B^i_{P^2}| - 2|B^i_{P^1}||B^i_{P^2}| && (2.35) \\
&= \underline{B}^1 + \underline{B}^2 - 2\underline{B}^1\underline{B}^2 && (2.36)
\end{aligned}$$

Since the fairness bound α is defined as the difference between the maximum and the minimum absolute biases, it follows:

$$\begin{aligned}
\alpha &= \max_i \Pr(P_i(\tilde{x}) \neq P_i(x)) - \min_i \Pr(P_i(\tilde{x}) \neq P_i(x)) && (2.37) \\
&= \overline{B}^1_i + \overline{B}^2 - 2\overline{B}^1_i\overline{B}^2 - \underline{B}^1_i + \underline{B}^2 - 2\underline{B}^1_i\underline{B}^2, && (2.38)
\end{aligned}$$

Replacing $\overline{B}^1_i = \underline{B}^1_i + \alpha_1$ and $\overline{B}^2 = \underline{B}^2 + \alpha_2$, gives the south fairness bound. □

Corollary 2.9. *Assume that mechanism $\mathcal{M}$ is fair w.r.t. problems P^1 and P^2. Then $\mathcal{M}$ is also fair w.r.t. $P^1 \oplus P^2$.*

The above is a direct consequence of Theorem 2.9 for $\alpha_1 = 0, \alpha_2 = 0$.

Corollary 2.10. *Consider an allocation problem P. Mechanism $\mathcal{M}$ is not fair w.r.t. P if there exist two entries $i, j \in [n]$ such that $\mathrm{Tr}(\boldsymbol{H}P_i)(\boldsymbol{x}) \neq \mathrm{Tr}(\boldsymbol{H}P_j)(\boldsymbol{x})$ for some dataset $\boldsymbol{x}$.*

The corollary is a direct consequence of Theorem 2.6 – see Equation (2.15).

Corollary 4 is not clear. First, Corollary 4 was mentioned in the main text to illustrate the unfairness of "non-convex function, as is the case for all the allocation problems considered in this chapter." However we are saying here this corollary is a direct consequence of Theorem 1, and in Theorem 1, we limit the mapping functions of quadratic form, i.e the second derivatives are constant.

I do not have the best solutions to fix the above issues because it might requires to reorganize the main text. If we can reorganize the main text, then one possible solution is to explain that sometimes we could not compute the level of unfairness α but can approximate it by Taylor approximation. Or if you or reviewers do not like the concept of approximation, then we can have a statement in Corollary 4 for general functions but the proof can not relies on Theorem 1.

Corollary 2.11. *Mechanism $\mathcal{M}$ is not fair w.r.t. $P_i(\langle x, y\rangle) = {}^{x}\!/_{y}$ and inputs x, y.*

This corollary is also a direct consequence of Theorem 2.6.

This should be a direct consequence of Corollary 4, not Theorem 1

Corollary 2.12. *Let mechanism $\mathcal{M}$ be α_k-fair w.r.t. to problem P^k ($k \in \{1, 2\}$). Then $\mathcal{M}$ is α-fair w.r.t. problems $P = P^1 \vee P^2$ and $P = P^1 \wedge P^2$, with $\alpha > \max(\alpha_1, \alpha_2)$.*

An analogous result holds for the logical connector $\wedge$.

Proof. The proof is provided for $P = P^1 \vee P^2$. By Theorem 2.8 it follows that:

$$\alpha = \left(\alpha_1 + \underline{B}^1 + \alpha_2 + \underline{B}^2 - (\alpha_1 + \underline{B}^1)(\alpha_2 + \underline{B}^2) - \underline{B}^1\underline{B}^2\right)$$

We need to show that $\alpha > \max(\alpha_1, \alpha_2)$.

First the proof shows that $\alpha > \alpha_1$. From the equation above, it follows,

$$\alpha - \alpha_1 = \underline{B}^1 + \alpha_2 + \underline{B}^2 - (\alpha_1 + \underline{B}^1)(\alpha_2 + \underline{B}^2) - \underline{B}^1\underline{B}^2 \tag{2.39}$$

$$= \underline{B}^1 + \alpha_2 + \underline{B}^2 - \alpha_1\alpha_2 - \alpha_1\underline{B}^2 - \alpha_2\underline{B}^1 \tag{2.40}$$

$$= \underline{B}^1(1 - \alpha_2) + \alpha_2(1 - \alpha_1) + \underline{B}^2(1 - \alpha_1) \tag{2.41}$$

Due to $\mathcal{M}$ being not trivial (by assumption), we have that $0 \leq \alpha_1, \alpha_2 < 0.5$. Thus, it follow:

$$\underline{B}^1(1 - \alpha_2) > 0 \tag{2.42}$$

$$\alpha_2(1 - \alpha_1) \geq 0 \tag{2.43}$$

$$\underline{B}^2(1 - \alpha_1) > 0. \tag{2.44}$$

Combining the three inequalities above with Equation 2.39, gives

$$\alpha - \alpha_1 > 0$$

which implies $\alpha > \alpha_1$. The same argument above follows for α_2. Therefore, $\alpha > \alpha_1$ and $\alpha > \alpha_2$, which asserts the claim. □

Theorem 2.10. *Let $\tilde{x} = x + (\lambda)$, with scale $\lambda > 0$, and $\hat{x} = PP^{\geq \ell}(\tilde{x})$, with $\ell < x$, be its post-processed value. Then,*

$$\mathbb{E}[\hat{x}] = x + \frac{\lambda}{2}\exp(\frac{\ell - x}{\lambda}).$$

Proof. The expectation of post-processed value $\hat{x}$ is given by:

$$\begin{aligned} E[\hat{x}] = \int_{-\infty}^{\infty} \max(\ell, \tilde{x})p(\tilde{x})d\tilde{x} = \int_{-\infty}^{\ell} \max(\ell, \tilde{x})p(\tilde{x})d\tilde{x} \\ + \int_{\ell}^{x} \max(\ell, \tilde{x})p(\tilde{x})d\tilde{x} + \int_{x}^{\infty} \max(\ell, \tilde{x})p(\tilde{x})d\tilde{x}, \end{aligned} \tag{2.45}$$

where $p(\tilde{x}) = \frac{1}{2\lambda}\exp(-\frac{|\tilde{x} - x|}{\lambda})$ is the pdf of Laplace. We will compute separately each of three

terms in Equation 2.45. We first have:

$$\begin{aligned}\int_{-\infty}^{\ell} \max(\ell, \tilde{x}) p(\tilde{x}) d\tilde{x} &= \int_{-\infty}^{\ell} \ell\, p(\tilde{x}) d\tilde{x} = \ell \int_{-\infty}^{\ell} p(\tilde{x}) d\tilde{x} \\ &= \ell \times 0.5 \exp(\frac{\ell - x}{\lambda}) = \frac{1}{2} \ell \exp(\frac{\ell - x}{\lambda})\end{aligned} \tag{2.46}$$

Second,

$$\begin{aligned}\int_{\ell}^{x} \max(\ell, \tilde{x}) p(\tilde{x}) d\tilde{x} &= \int_{\ell}^{x} \tilde{x} p(\tilde{x}) d\tilde{x} \\ &= \frac{1}{2}(x - \lambda) - \frac{1}{2}(\ell - \lambda) \exp(\frac{\ell - x}{\lambda})\end{aligned} \tag{2.47}$$

Finally

$$\begin{aligned}\int_{x}^{\infty} \max(\ell, \tilde{x}) p(\tilde{x}) d\tilde{x} &= \int_{x}^{\infty} \tilde{x} p(\tilde{x}) d\tilde{x} \\ &= \frac{1}{2}(x + \lambda)\end{aligned} \tag{2.48}$$

Combine Equations (2.46)–(5.3) with Equation (2.45), gives

$$E[\hat{x}] = x + \frac{\lambda}{2} \exp(\frac{\ell - x}{\lambda})$$

□

2.9.2 The nature of bias (Ext)

Predicates Composition When $\mathcal{M}$ is fair w.r.t P^1 and P^2 we have the following Property

Property 2.2. *Suppose mechanism $\mathcal{M}$ is fair w.r.t. P^1 and P^2, and consider predicate $P = P^1 \wedge P^2$. Let $|B_P(a, b)|$ denote the absolute bias for $\mathcal{M}$ w.r.t. P when predicate $P^1 = a$ and predicate $P^1 = b$, for $a, b \in \{\textsf{True}, \textsf{False}\}$. Then, $|B_P(\textsf{True}, \textsf{True})| \geq |B_P(a, b)|$ for any other $a, b \in \{\textsf{True}, \textsf{False}\}$.*

Proof. Since, by assumption, $\mathcal{M}$ is fair w.r.t. to both predicates P^1 and P^2, then its absolute bias

w.r.t. these predicates is constant, that is $\forall i \in [n]$,

$$|B^i_{P^1}| = B_1 \in (0, 0.5)$$
$$|B^i_{P^2}| = B_2 \in (0, 0.5)$$

By Lemma 2.1 and for every $x \in \mathcal{X}$, it follows:

1. $P^1_i(\boldsymbol{x}) = 0 \wedge P^2_i(\boldsymbol{x}) = 0 \Rightarrow$
   $\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = B_1 B_2$

2. $P^1_i(\boldsymbol{x}) = 0 \wedge P^2_i(\boldsymbol{x}) = 1 \Rightarrow$
   $\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = (1 - B_2) B_1$

3. $P^1_i(\boldsymbol{x}) = 1 \wedge P^2_i(\boldsymbol{x}) = 0 \Rightarrow$
   $\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = (1 - B_1) B_2$

4. $P^1_i(\boldsymbol{x}) = 1 \wedge P^2_i(\boldsymbol{x}) = 1 \Rightarrow$
   $\Pr(P_i(\tilde{\boldsymbol{x}}) \neq P_i(\boldsymbol{x})) = B_1 + B_2 - B_1 B_2$

We are interested in showing that the last quantity is the largest among the four quantities above. From the equalities above the following sequence of relations can be derived,

$$(4) > (2); (4) > 3; (2) > (1); (3) > 1$$

which shows that quantity (4) is the largest among all the other possibilities and concludes the proof. □

Property 2.3. *Suppose mechanism $\mathcal{M}$ is fair w.r.t. P^1 and P^2, and consider predicate $P = P^1 \vee P^2$. Let $|B_P(a, b)|$ denote the absolute bias for $\mathcal{M}$ w.r.t. P when predicate $P^1 = a$ and predicate $P^1 = b$, for $a, b \in \{\textsf{True}, \textsf{False}\}$. Then, $|B_P(\textsf{False}, \textsf{False})| \geq |B_P(a, b)|$ for any other $a, b \in \{\textsf{True}, \textsf{False}\}$.*

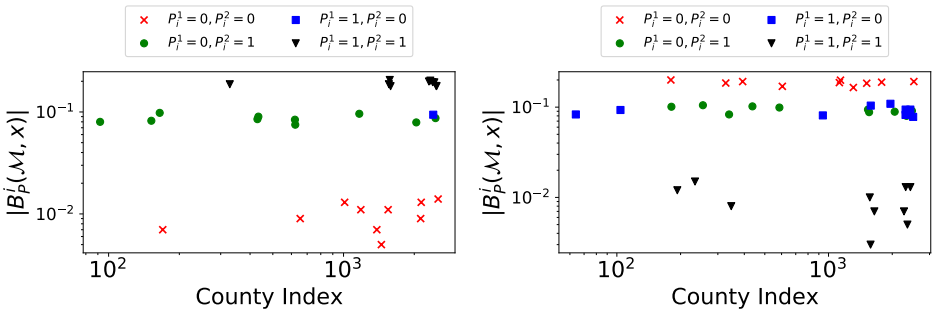


Fig. 2.9: Decision errors for four different groups of data under $P = P^1 \wedge P^2$ (left) and $P = P^1 \vee P^2$ (right)

The proof is similar to the proof of Property 2.2.

Figure 2.9 illustrates this result on the Minority Language dataset. It reports the decision errors on the y-axis (absolute bias). Notice that the red group is the least penalized in Figure 2.9 (left) and the most penalized in Figure 2.9 (right).

Post-processing This discussion analyzes positive results for two additional classes of post-processing, with respect to that analyzed in the main chapter: (1) The integrality constraint program $\text{PP}^{\mathbb{N}}(z)$ which makes sure the released data of integer data types (2) The sum-constrained constrained program $\text{PP}^{\sum_S}(z)$ which enforces the private data satisfies some equality constraints. The following results show that these post-processing do not contribute to further bias the decisions.

Integrality Post-processing The integrality post-processing $\text{PP}^{\mathbb{N}}(z)$ is used when the released data are integral quantities. To make sure that this processing step does not introduce additional bias, we can rely on the stochastic rounding technique:

$$\text{PP}^{\mathbb{N}}(z) = \begin{cases} z \text{ w.p.: } 1 - (z - z) \\ z + 1 \text{ w.p.: } z - z \end{cases} \tag{2.49}$$

The stochastic rounding guarantees that $\mathbb{E}[\text{PP}^{\mathbb{N}}(\tilde{x})] = \tilde{x}$ so no additional bias will introduce to $\text{PP}^{\mathbb{N}}(\tilde{x})$.

Sum-constrained Post-processing The sum-constrained post-processing $\text{PP}^{\sum_S}(z)$ can be formulated as the solution of following constrained optimization:

$$\min_{\hat{z}} \|\hat{z} - z\|_2^2 \text{ s.t : } \mathbf{1}^T z = S \tag{2.50}$$

The sum-constrained post-processing is of need in case when one requires the private outcome should satisfy some equality constraint, e.g the number of private representative in Apportionment of Legislative Representatives should sum up to $S = 543$. We have the following positive result regarding sum-constrained post-processing.

Theorem 2.11. *Consider a mechanism $\mathcal{M}$ which is α-fair w.r.t problem P, then $\mathcal{M}$ is also α-fair w.r.t problem $\text{PP}^{\sum_S}(P)$*

Note that our result is an extension of previous work [146].

Proof. For notation convenience, denote $z_i = P_i(\tilde{\boldsymbol{x}})$ so $\hat{z} = \text{PP}^{\sum_S}(z)$. It is easy to see that one unique solution for the convex optimization problem in Equation 2.11 is $\hat{z}_i = z_i + \eta$, where $\eta = \frac{S - \sum_i z_i}{n}$.

For an entity i by linearization of expectation, it follows:

$$\begin{aligned}
\mathbb{E}[\hat{z}_i] = \mathbb{E}[z_i + \eta] &= \mathbb{E}[z_i + \frac{S - \sum_{j \neq i} z_j}{n}] && (2.51)\\
&= \frac{n-1}{n}\mathbb{E}[z_i] - \frac{1}{n}\sum_{j \neq i}\mathbb{E}[z_j] + \frac{S}{n} && (2.52)\\
&= \frac{n-1}{n}(z_i + B_P^i) - \frac{1}{n}\big(\sum_{j \neq i} z_j + B_P^j)\big) + \frac{S}{n} && (2.53)\\
&= \frac{n-1}{n}(z_i + B_P^i) - \frac{1}{n}\big(S - z_i + \sum_{j \neq i} B_P^j\big) + \frac{S}{n} && (2.54)\\
&= z_i + \frac{\sum_{j \neq i}(B_P^i - B_P^j)}{n}. && (2.55)
\end{aligned}$$

The last equality indicates that bias for i-th entity under sum-constrained post-processing is $B^i_{\text{PP}^{\sum_S}(P)} =$

$\frac{\sum_{j\neq i}(B_P^i - B_P^j)}{n}$. Thus the fairness bound α' after post-processing can be determined by:

$$\alpha' = \max_{i,k} |B^i_{\text{PP}\Sigma_S(P)} - B^k_{\text{PP}\Sigma_S(P)}| \tag{2.56}$$

$$= \max_{i,k} \frac{\sum_{j\neq i}(B_P^i - B_P^j)}{n} - \frac{\sum_{j\neq k}(B_P^k - B_P^j)}{n} \tag{2.57}$$

$$= \max_{i,k} |B_P^i - B_P^k| = \alpha \tag{2.58}$$

The last equality asserts the sum-constrained post-processing does not introduce additional unfairness to mechanism $\mathcal{M}$. □

2.9.3 Mitigating solutions (Ext)

Learning piece-wise linear proxy-functions This subsection discusses a practical mitigating solution for decision rules of type P^M. Due to the discontinuities and non-linearities arising in this problem, approximating P^M using a single proxy-linear function may introduce high errors. The idea explored below relies to partition the inputs $\boldsymbol{x}$ into several groups $\boldsymbol{x}_1, \ldots \boldsymbol{x}_G$, e.g., by grouping individuals from the same state or by looking at similarity of counts. For each group $k \in [G]$, a linear-proxy function $\bar{P}^M_k(\boldsymbol{x}_k)$ can thus be constructed. The resulting problem $\bar{P}^M$ is a combination of these linear-proxy functions, and thus a piece-wise linear function that aims at approximating the original problem P^M.

Rather than using an ad-hoc method to linearize problem P^M, the chapter proposes to learn a linear function by fitting a Linear regression model or a Linear Support Vector Classifier model to the data $\boldsymbol{x}_k$ of each group $k \in [G]$. Specifically, each group is trained using features $\{x^{spe}, x^{sp}, x^s\}$ and the resulting coefficient are used to construct the proxy linear function. The proxy problem $\bar{P}^M$ is thus represented by a set of (piece-wise) linear functions whose coefficients are learned to approximate the original data. This is similar to the purpose that a decision rule has – capture some aspect of the true data.

The chapter uses the value x^{sp} to partition the dataset into 9 groups of approximately equal

size. To ensure privacy, the grouping is executed using privacy-preserving x^{sp} values.

Figure 2.10 compares the original problem P, a proxy-model $\bar{P}_{LR}$ whose pieces are learned using linear regression (LR) and a proxy model $\bar{P}_{SVM}$ whose pieces are learned using a linear SVM model. All three problem take as input the private data $\tilde{\boldsymbol{x}}$ and are compared with the original version of the problem P. The x-axis shows the range of x^{sp} that defines each group while the y-axis shows the fairness bound α (computed within each group). The positive effects of the proposed piece-wise linear proxy problem are dramatic. The fairness violations decrease significantly when compared to those obtained by the original model. We also noticed that the fairness violation of the SVM model is typically lower than that obtained by the LR model, and this may be due to the accuracy of the resulting model – with SVM reaching higher accuracy than LR in our experiments. Finally, as the population size increases, the fairness bound α decreases. This aspect was already shown in the main chapter, and emphasizes further the largest negative impact of the noise on the smaller counties.

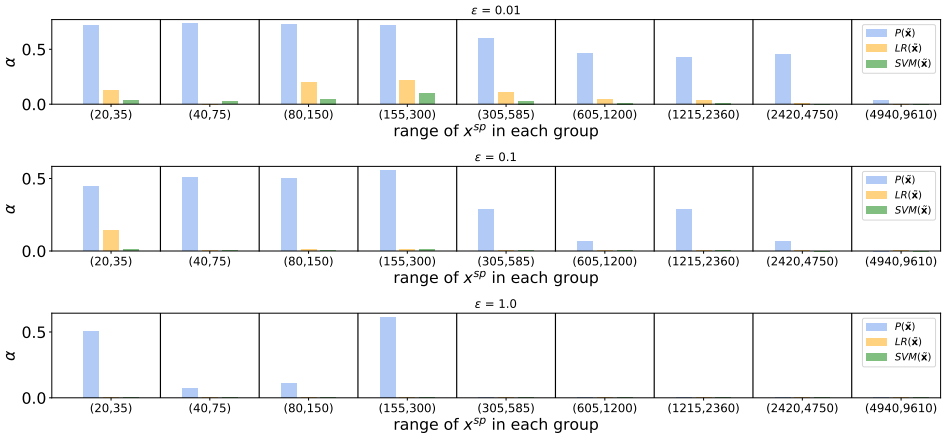


Fig. 2.10: fairness α under private grouping

2.9.4 Experimental details

General settings All experimental codes were written in Python 3.7. Some heavy computation tasks were performed on a cluster equipped with Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz and 8GB of RAM. We will release our codes upon chapter's acceptance.

Datasets

Title 1 School Allocation The dataset was uploaded as a supplemental materials of [6]. The dataset can be downloaded directly from `https://tinyurl.com/y6adjsyn`.

We processed the dataset by removing schools which contains NULL information, and keeping school districts with at least 1 students. The post-processed dataset left with 16441 school districts.

Minority language voting right benefits The dataset can be downloaded from `https://tinyurl.com/y2244gbt`.

The focus of the experiments is on Hispanic groups, which represent the largest minority population. There are 2774 counties that contain at least a Hispanic person.

Mechanism implementation

Linear Proxy Allocation $\bar{P}^F$ The linear proxy allocation method used in problem $\bar{P}^F$, is implement so that, for a given privacy parameter ϵ, the algorithm allocates $\epsilon_1 = \frac{\epsilon}{2}$ to release the normalization term Z. The remaining $\epsilon_2 = \frac{\epsilon}{2}$ budget is used to publish the population counts x_i.

Output Perturbation mechanism To implement the output perturbation method for problem P^F the chapter uses a Laplace mechanism. That is, the private outcome $\tilde{P}_i^F(\boldsymbol{x}) = P_i(x) + \text{Lap}(0, \frac{\Delta}{\epsilon})$ where the global sensitivity Δ is obtained from Theorem 2.12. In the experiments, we set the known public lower bound $L = 0.9Z$ for the normalization term Z in Theorem 2.12.

Theorem 2.12. *In the Title I School Allocation Problem, denote $a_{\max} = \max_i a_i$, and let $L \leq \sum_{i\in[n]} x_i a_i$ is a known public lower bound for the normalization term. The l_1 global sensitivity of the query $P^F = \{P_i^F\}_{i=1}^n$ where, $P_i^F = \left(\frac{x_i a_i}{\sum_{i\in[n]} x_i a_i}\right)$ is given by:*

$$\Delta = \max_{\boldsymbol{x},\boldsymbol{x'}} \left| P^F(\boldsymbol{x}) - P^F(\boldsymbol{x'}) \right|_1 = \frac{2a_{\max}}{L} \tag{2.59}$$

Proof. W.o.l.g assume we remove one individual from i-th district school in the dataset $\boldsymbol{x}$ to obtain the dataset $\boldsymbol{x}'$. This surely will affect the fraction of fund allocation of all district schools. Denote $Z = \sum_j x_j a_j$, then we have:

$$\begin{aligned} P_i^F(\boldsymbol{x}) - P_i^F(\boldsymbol{x'}) &= \frac{x_i a_i}{Z} - \frac{(x_i - 1)a_i}{Z - a_i} \\ P_j^F(\boldsymbol{x}) - P_j^F(\boldsymbol{x'}) &= \frac{x_j a_j}{Z} - \frac{x_j a_j}{Z - a_j} \; \forall j \neq i \end{aligned} \tag{2.60}$$

First, it is simple to show that:

$$\begin{aligned} P_i^F(\boldsymbol{x}) - P_i^F(\boldsymbol{x'}) &= \frac{a_i \left[Z - x_i a_i\right]}{Z(Z - a_i)} \\ &\leq \frac{a_i(Z - a_i)}{Z(Z - a_i)} = \frac{a_i}{Z} \leq \frac{a_{\max}}{Z} \\ &\leq \frac{a_{\max}}{L} \end{aligned} \tag{2.61}$$

The last inequality is due to the assumption that $Z = \sum_{j \in [n]} a_j \cdot x_j \geq L$. Second, $\forall j \neq i$ we have:

$$P_j^F(\boldsymbol{x}) - P_j^F(\boldsymbol{x'}) = \frac{-a_j x_j a_i}{Z(Z - a_i)} \tag{2.62}$$

Hence:

$$\begin{aligned} \sum_{j \neq i} \left| P_j^F(\boldsymbol{x}) - P_j^F(\boldsymbol{x'}) \right| &= a_i \frac{Z - a_i x_i}{Z(Z - a_i)} \\ \leq a_i \frac{Z - a_i}{Z(Z - a_i)} &= a_i \frac{1}{Z} \leq \frac{a_{\max}}{L} \end{aligned} \tag{2.63}$$

Combine Equation (2.61) and Equation (2.63), we obtain:

$$|P(\boldsymbol{x}) - P(\boldsymbol{x'})| \leq \left|P_i^F(D) - P_i^F(D')\right| + \sum_{j \neq i} \left|P_j^F(\boldsymbol{x}) - P_j^F(\boldsymbol{x'})\right| \leq \frac{2a_{\max}}{L}, \tag{2.64}$$

The last inequality asserts the corrects of Theorem 2.12

□

2.9.5 Related work

In literature the work on algorithmic fairness and differential privacy were mostly studied separately. However, there are several noticeable work which address the connection between these two concepts. One of the earliest work was [36] which showed individual fairness is a general notation of differential privacy. More recently, Cummings et. al [33] consider the tradeoffs when considering differential privacy and equal opportunity, a notion of fairness that restricts a classifier to produce equal true positive rates across different groups. The work claim that there is no classifier that achieves ϵ-differential privacy, satisfies equal opportunity, and has accuracy better than a constant classifier. Ekstrand et. al [40] raise questions about the tradeoffs involved between privacy and fairness and, finally, Jagielski et. al [63] shows two simple, yet effective algorithms that satisfy (ϵ, δ)-differential privacy and equalized odds. Finally, a recent line of work has also observed that private models may have a negative impact towards fairness. In particular, Pujol et. al [71] shows that differential privacy could disproportionately affect some groups on several Census resource allocation tasks. A similar observation was made by Bagdasaryan et. al [12] in the context of private deep learning models trained using DP-SDG. The authors observed disparity in performance across different sub-populations on several classification tasks. In addition, Chang et al [26] has showed that fairness comes at the cost of privacy and this cost is not distributed uniformly: the information leakage of fair models increases significantly on the unprivileged subgroups, which suffer from the discrimination in regular models. Finally, Khalili et. al [81], has indicated that

exponential mechanism [83] can be used as a post-processing step to improve fairness and privacy of the pre-trained supervised model.

CHAPTER 3
DISPARATE IMPACTS OF PRIVACY INTO LEARNING TASKS

Differential Privacy (DP) [37] is an important privacy-enhancing technology for private machine learning systems. It allows to measure and bound the risk associated with an individual participation in a computation. However, it was recently observed that DP learning systems may exacerbate bias and unfairness for different groups of individuals [12, 71, 137]. This chapter builds on these important observations and sheds light on the causes of the disparate impacts arising in the problem of differentially private empirical risk minimization. It focuses on the accuracy disparity arising among groups of individuals in two well-studied DP learning methods: output perturbation [27] and differentially private stochastic gradient descent [4]. The chapter analyzes which data and model properties are responsible for the disproportionate impacts, why these aspects are affecting different groups disproportionately, and proposes guidelines to mitigate these effects. The proposed approach is evaluated on several datasets and settings.

3.1 Introduction

While learning systems have become instrumental for many decisions and policy operations involving individuals, the use of rich datasets combined with the adoption of black-box algorithms has sparked concerns about how these systems operate. Two key concerns regard how these systems handle discrimination and how much information they leak about the individuals whose data is used as input.

Differential Privacy (DP) [37] has become the paradigm of choice for protecting data privacy and its deployments are growing at a fast rate. DP is appealing as it bounds the risks of disclosing sensitive information of individuals participating in a computation. However, it was recently observed that DP systems may induce biased and unfair outcomes for different groups of individuals [12, 71, 137]. The resulting outcomes can have significant societal and economic impacts on the involved individuals: classification errors may penalize some groups over others in important determinations including criminal assessment, landing, and hiring [12] or can result in disparities regarding the allocation of critical funds, benefits, and therapeutics [71]. *While these surprising observations have become apparent in several contexts, their causes are largely understudied and not fully understood.*

This chapter makes a step toward addressing this important knowledge gap. It builds on these key observations and sheds light on the causes of the disparate impacts arising in the problem of differentially private empirical risk minimization (ERM). It focuses on the accuracy disparity arising among groups of individuals in two well-studied DP learning methods: output perturbation [27] and differentially private stochastic gradient descent (DP-SGD) [4]. The chapter analyzes which properties of the model and the data are responsible for the disproportionate impacts, why these aspects are affecting different groups disproportionately, and proposes guidelines to mitigate these effects.

In summary, the chapter makes the following contributions:

1. It develops a notion of fairness under private training that relies on the concept of excessive

risk.

2. It analyzes this fairness notion in two DP learning methods: output perturbation and DP-SGD.

3. It isolates the relevant components related with noise addition and gradient clipping responsible for the disparate impacts.

4. It studies the behaviors and the causes for these components to affect different groups of individuals disproportionately during private training.

5. Based on these observations, it proposes a mitigation solution and evaluates its effectiveness on several standard datasets.

To the best of the authors knowledge, this work represents a first step toward a deeper understanding of the causes of the unfairness impacts in differentially private learning.

3.2 Related work

The research at the interface between differential privacy and fairness is receiving increasing attention and can be broadly categorized into three main lines of work. The first shows that DP is in alignment with fairness. Notable contribution in this direction include [36] seminal work, which highlights the relation between individual fairness and differential privacy, and [81], which shows that the private exponential mechanism can produce fair outcomes in some selection problems. Works in the second category study the setting under which a fair model can leak privacy [26,63,87,128,140]. These works propose learning frameworks that guarantee DP while also encouraging the satisfaction of different notions of fairness. For example, [137] proposes a private and fair variant of DP-SGD that uses separate clipping bounds for each groups of individuals. Such proposal encourages accuracy parity at the expense of an extra privacy cost (required to customize the clipping bound for each group). Works in the last category show that private mechanisms can have a negative impact towards fairness [12,43,71,128,137]. For example, Cummings et al. [33]

shows that it is impossible to achieve *exact* equalized odds while also satisfying pure DP. [71] observe that decisions made using a private version of a dataset may disproportionately affect some groups over others. Similar observations were also made in the context of model learning. [12] empirically observed that the accuracy of a DP model trained using DP-SGD drops disproportionately across groups causing larger negative impacts to the underrepresented groups. [43] reaches similar conclusions. The authors empirically show that the disparate impact of differential privacy on model accuracy is not limited to highly imbalanced data and can occur even in situations where the classes are slightly imbalanced.

This chapter builds on this body of work and their important empirical observations. It derives the conditions and studies the causes of unfairness in the context of private empirical risk minimization problems as well as it introduces mitigating guidelines.

3.3 Preliminaries

Differential privacy (DP) [37] is a strong privacy notion used to quantify and bound the privacy loss of an individual participation to a computation. Informally, it states that the probability of any output does not change much when a record is added or removed from a dataset, limiting the amount of information that the output reveals about any individual. The action of adding or removing a record from a dataset D, resulting in a new dataset D', defines the notion of *adjacency*, denoted $D \sim D'$.

Definition 3.1. *A mechanism $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ with domain $\mathcal{D}$ and range $\mathcal{R}$ is (ϵ, δ)-differentially private, if, for any two adjacent inputs $D \sim D' \in \mathcal{D}$, and any subset of output responses $R \subseteq \mathcal{R}$:*

$$\Pr[\mathcal{M}(D) \in R] \leq e^{\epsilon} \Pr[\mathcal{M}(D') \in R] + \delta.$$

Parameter $\epsilon > 0$ describes the *privacy loss* of the algorithm, with values close to 0 denoting strong privacy, while parameter $\delta \in [0, 1)$ captures the probability of failure of the algorithm to satisfy ϵ-DP. The global sensitivity Δ_ℓ of a real-valued function $\ell : \mathcal{D} \to \mathbb{R}^k$ is defined as the

maximum amount by which ℓ changes in two adjacent inputs: $\Delta_\ell = \max_{D \sim D'} \|\ell(D) - \ell(D')\|$. In particular, the Gaussian mechanism, defined by $\mathcal{M}(D) = \ell(D) + \mathcal{N}(0, \Delta_\ell^2\, \sigma^2)$, where $\mathcal{N}(0, \Delta_\ell^2\, \sigma^2)$ is the Gaussian distribution with 0 mean and standard deviation $\Delta_\ell^2\, \sigma^2$, satisfies (ϵ, δ)-DP for $\delta > \frac{4}{5} \exp(-(\sigma\epsilon)^2/2)$ and $\epsilon < 1$ [39].

3.4 Problem settings and goals

The chapter adopts boldface symbols to describe vectors (lowercase) and matrices (uppercase). Italic symbols are used to denote scalars (lowercase) and data features or random variables (uppercase). Notation $\|\cdot\|$ is used to denote the L_2 norm. The chapter considers datasets D consisting of n individuals' data points (X_i, A_i, Y_i), with $i \in [n]$ drawn i.i.d. from an unknown distribution. Therein, $X_i \in \mathcal{X}$ is a feature vector, $A_i \in \mathcal{A}$ is a protected group attribute, and $Y_i \in \mathcal{Y}$ is a label. For example, consider the case of a classifier that needs to predict the risks associated with a lending decision. The training example features X_i may describe the individual's demographics, education, credit score, and loan amount, the protected attribute A_i may describe the individual gender or ethnicity, and Y_i represents whether or not the individual will default on the loan. The goal is to learn a classifier $f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$, where $\boldsymbol{\theta}$ is a vector of real-valued parameters, that guarantees the *privacy* of each individual data (X_i, A_i, Y_i) in D. The model quality is measured in terms of a nonnegative *loss function* $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$, and the problem is that of minimizing the empirical risk (ERM) function:

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; D) = \frac{1}{n} \sum_{i=1}^{n} \ell(f_{\boldsymbol{\theta}}(X_i), Y_i). \tag{L}$$

For a group $a \in \mathcal{A}$, the chapter uses D_a to denote the subset of D containing exclusively samples whose group attribute $A = a$. The chapter focuses on learning classifiers that protect the disclosure of the individuals' data using the notion of differential privacy and it analyzes the fairness impact (as defined next) of privacy on different groups of individuals. Importantly, the chapter assumes that the attribute A is not part of the model input during inference.

Fairness The fairness analysis focuses on the notion of *excessive risk*, a widely adopted metric in private learning [132, 142]. It defines the difference between the private and non private risk functions:

$$R(\boldsymbol{\theta}, D) = \mathbb{E}_{\tilde{\boldsymbol{\theta}}}\left[\mathcal{L}(\tilde{\boldsymbol{\theta}}; D)\right] - \mathcal{L}(\boldsymbol{\theta}^*; D), \tag{3.1}$$

where the expectation is defined over the randomness of the private mechanism and $\tilde{\boldsymbol{\theta}}$ denotes the private model parameters while $\boldsymbol{\theta}^* = \operatorname{argmin}_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; D)$. The chapter uses shorthands $R(\boldsymbol{\theta})$ and $R_a(\boldsymbol{\theta})$ to denote, respectively, the population-level $R(\boldsymbol{\theta}, D)$ excessive risk and the group level $R(\boldsymbol{\theta}, D_a)$ excessive risk for group a. Fairness is measured with respect to the *excessive risk gap*:

$$\xi_a = |R_a(\boldsymbol{\theta}) - R(\boldsymbol{\theta})|. \tag{3.2}$$

(Pure) fairness is achieved when $\xi_a = 0$ for all groups $a \in \mathcal{A}$ and, thus, a private and fair classifier aims at minimizing the maximum excessive risk gap among all groups. The chapter assumes that the private mechanisms are non-trivial, i.e., they minimize the population-level excessive risk $R(\boldsymbol{\theta})$.

All proofs are reported in the Section, Section 3.11.1.

3.5 Warm up: output perturbation

The chapter starts with analyzing fairness under the DP setting induced by an output perturbation mechanism. In this setting the analysis restricts to twice differentiable and convex loss functions ℓ. Output perturbation is a standard DP paradigm in which noise calibrated to the function sensitivity is added directly to the output of the computation. In the context of the *regularized* ERM problem, adding noise drawn from a Gaussian distribution $\mathcal{N}(0, \Delta_\ell^2 \sigma^2)$ to the optimal model parameters $\boldsymbol{\theta}^*$ ensures (ϵ, δ)-differential privacy [27]. Therein, $\Delta_\ell = 2/n\lambda$ with regularization parameter λ. The following result sheds light on the unfairness induced by this mechanism.

Theorem 3.1. *Let ℓ be a twice differentiable and convex loss function and consider the output perturbation mechanism described above. Then, the excessive risk gap for group $a \in \mathcal{A}$ is approximated by:*

$$\xi_a \approx \frac{1}{2}\Delta_\ell^2 \sigma^2 \left|\mathrm{Tr}(\boldsymbol{H}_\ell^a) - \mathrm{Tr}(\boldsymbol{H}_\ell)\right|, \tag{3.3}$$

where $\boldsymbol{H}_\ell^a = \nabla^2_{\boldsymbol{\theta}^*} \sum_{(X,A,Y)\in D_a} \ell(f_{\boldsymbol{\theta}^*}(X), Y)$ is the Hessian matrix of the loss function, at the optimal parameters vector $\boldsymbol{\theta}^*$, computed using the group data D_a, $\boldsymbol{H}_\ell$ is the analogous Hessian computed using the population data D, and $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix.

The approximation above follows form a second order Taylor expansion of the loss function, linearity of expectation, and the properties of Gaussian distributions. It uses that fact that the excessive risk $R_a(\boldsymbol{\theta})$ for a group a can be approximated as $1/2\Delta_\ell^2\sigma^2\,\mathrm{Tr}(\boldsymbol{H}_\ell^a)$. The proof is reported in Section 3.11.1.

Theorem 3.1 sheds light on the relation between fairness and the difference in the local curvatures of the losses ℓ associated with a group and the population and provides a necessary condition to guarantee pure fairness. It suggests that output perturbation mechanisms may introduce unfairness when the local curvatures associated with the loss function of different groups differ substantially from one another. Additionally, the unfairness level is proportional to the amount of noise σ or, equivalently, inversely proportional to the privacy parameter ϵ, for a fixed δ. Finally, it also suggests that groups with larger Hessian traces $\mathrm{Tr}(\boldsymbol{H}_a^\ell)$ will have larger excessive risk compared to groups with smaller Hessian traces. An additional analysis on the reasons behind why different groups may have large differences in their associated Hessian traces is provided in Section 3.8.

Figure 3.1 illustrates Theorem 3.1. The plots show the correlation between the excessive risk [1] and the quantity $\mathrm{Tr}(\boldsymbol{H}_z^\ell)$ for each group $z \in \mathcal{A}$, at varying of the privacy loss $\epsilon \in [0.005, 0.5], \delta = 1e^{-5}$ on two datasets. Each data point represents the average of 100 runs of a DP Logistic Regression (obtained with output perturbation) on each group $z \in \mathcal{A}$. Details on dataset and experimental

[1] In all experiment presented, the excessive risk is approximated by sampling over 100 repetitions.

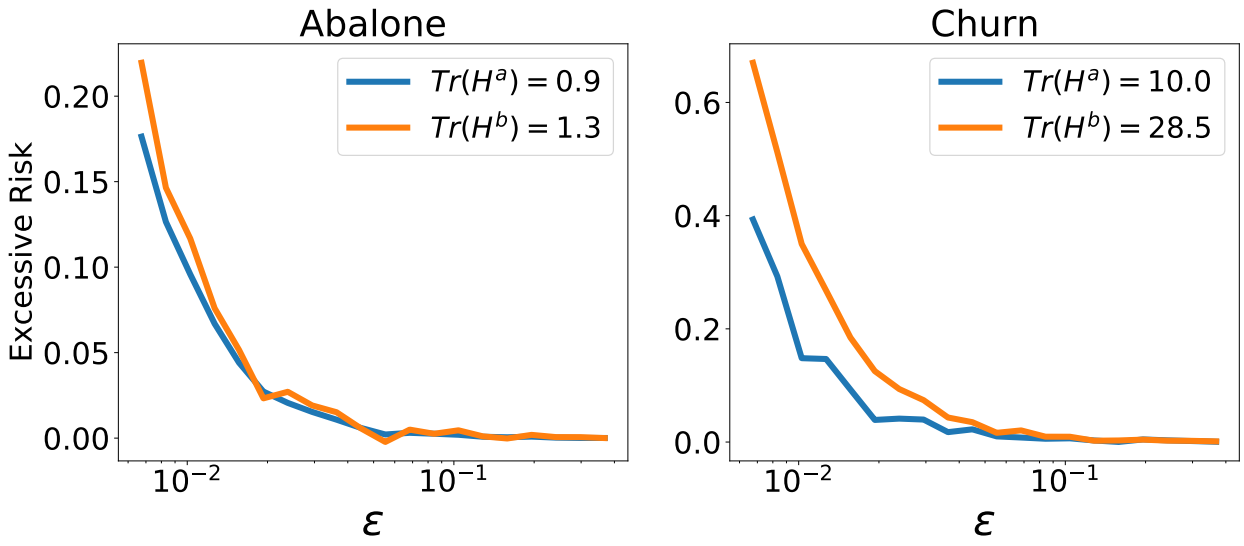


Fig. 3.1: Correlation between excessive risk and Hessian Traces at varying of the privacy loss ϵ.

setting are provided in Section 3.11.2 and additional experiments in Section 3.11.3. Note the positive correlation between the excessive risk and the Hessian trace: *Groups with larger Hessian traces tend to have larger excessive risks*. Note also the inverse correlation between ϵ and the dependency between the excessive risk and the Hessian trace. This is due to that larger ϵ values require smaller σ values, and thus, as shown in Equation 3.3, the dependency between the excessive risk and Hessian trace is attenuated.

The following illustrates that even a class of simple linear models may not to satisfy pure fairness.

Corollary 3.1. *Consider the ERM problem for a linear model* $f_{\boldsymbol{\theta}}(X) \stackrel{def}{=} \boldsymbol{\theta}^T X$*, with* L_2 *loss function i.e.,* $\ell(f_{\boldsymbol{\theta}}(X), Y) = (f_{\boldsymbol{\theta}}(X) - Y)^2$*. Then, output perturbation does not guarantee pure fairness.*

It follows from the observation that the Hessian of the L_2 loss for group $a \in \mathcal{A}$, i.e., $\mathrm{Tr}(\boldsymbol{H}_\ell^a) = \mathbb{E}_{X \sim D_a} \mathrm{Tr}(XX^T) = \mathbb{E}_{X \sim D_a} \|X\|^2$, depends solely on the input norms of the elements in D_a[2]. Interestingly, this result highlights the relation between fairness and the average input norms of different group elements. When these norms are substantially different one another they will impact their respective excessive risks differently. An additional analysis on this behavior is also discussed in Section 3.7.

[2]Throughout the chapter, we abuse notation and treat the dataset D_Z associated with group Z as distributions.

Algorithm 1: DP-SGD

input: Disjoint dataset D ; Sample prob. q; Iterations T; Noise variance σ^2; Clipping bound C; learning rate η

$\boldsymbol{\theta}_0 \leftarrow \mathbf{0}^T$

for *iteration* $t = 1, 2, \ldots T$ **do**

  $B \leftarrow$ random sub-sample of D with $\Pr q$

  foreach $(X_i, A_i, Y_i) \in B$ **do**

    $\boldsymbol{g}_i = \nabla \ell \left(f_{\boldsymbol{\theta}_t}(X_i), Y_i \right)$

  $\bar{\boldsymbol{g}}_B \leftarrow \frac{1}{|B|} \left(\sum_i \pi_C(\boldsymbol{g}^i) + \mathcal{N}(0, \boldsymbol{I} C^2 \sigma^2) \right)$

  $\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t - \eta \bar{\boldsymbol{g}}_B$

The following is a positive result.

Corollary 3.2. *If for any two groups* $a, b \in \mathcal{A}$ *their average group norms* $\mathbb{E}_{X_a \sim D_a} \|X_a\| = \mathbb{E}_{X \sim D_b} \|X_b\|$ *have identical values, then output perturbation with* L_2 *loss function provides pure fairness.*

The above is a direct consequence of Corollary 3.3. *Note also that pure fairness may be achieved, in this setting, by normalizing the input values for each group independently* (as shown in Section 3.11.3) although this solution requires accessing the sensitive group attributes at inference time.

3.6 Gradient perturbation: DP-SGD

Having identified the dependency between the Hessian of the model loss and the privacy parameters with the excessive risk gap in output perturbation mechanisms, this section extends the analysis to the context of DP Stochastic Gradient Descent (DP-SGD) [4]. In contrast to output perturbation, DP-SGD does not restrict focus on convex loss functions and the privacy analysis does not require optimality of the model parameters $\boldsymbol{\theta}$, rendering it an appealing framework for DP ERM problems.

In a nutshell, DP-SGD computes the gradients for each data sample in a random mini-batch B, clips their L_2-norm, adds noise to ensure privacy, and computes the average. Two key characteristics of DP-SGD are: **(1)** Clipping the gradients whose L_2 norm exceeds a given bound C,

and **(2)** Perturbing the averaged clipped gradients with 0-mean Gaussian noise with variance $\sigma^2 C^2$. The procedure is described in Algorithm 2. Therein, $\boldsymbol{g}_i$ represents the gradient of a data sample (X_i, A_i, Y_i), $\bar{\boldsymbol{g}}_B$ the average clipped noisy gradient of the samples in mini-batch B, and the function $\pi_C(\boldsymbol{x}) = \boldsymbol{x} \cdot \min(1, \frac{C}{\|\boldsymbol{x}\|})$.

The following theorem is an important result of this section. It connects the expected loss $\mathbb{E}[\mathcal{L}(\boldsymbol{\theta}; D_a)]$ of a group $a \in \mathcal{A}$ with its excessive risk $R_a(\boldsymbol{\theta})$, which is, in turn, used in our fairness analysis. It decomposes the expected loss during private training into three key components: The first relates with the model parameters update and it is not affected by the private training. The other two relate with gradient clipping and noise addition, and, combined, capture the notion of excessive risk.

Theorem 3.2. *Consider the ERM problem* (6.1) *with loss ℓ twice differentiable w.r.t. the model parameters. The expected loss $\mathbb{E}[\mathcal{L}(\boldsymbol{\theta}_{t+1}; D_a)]$ of group $a \in \mathcal{A}$ at iteration $t+1$, is approximated as:*

$$\mathbb{E}\left[\mathcal{L}(\boldsymbol{\theta}_{t+1}; D_a)\right] = \underbrace{\mathcal{L}(\boldsymbol{\theta}_t; D_a) - \eta \left\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D \right\rangle + \frac{\eta^2}{2} \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B\right]}_{\textit{non-private term}} \tag{3.4}$$

$$+ \underbrace{\eta \left(\left\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D \right\rangle - \left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_D \right\rangle\right) + \frac{\eta^2}{2} \left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B\right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B\right]\right)}_{\textit{private term due to clipping}} \qquad (R_a^{\textit{clip}})$$

$$+ \underbrace{\frac{\eta^2}{2} \operatorname{Tr}(\boldsymbol{H}_\ell^a) C^2 \sigma^2}_{\textit{private term due to noise}} \qquad (R_a^{\textit{noise}})$$

$$+ \; O(\|\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\|^3),$$

where the expectation is taken over the randomness of the private noise and the mini-batch selection, and the terms $\boldsymbol{g}_Z$ and $\bar{\boldsymbol{g}}_Z$ denote, respectively, the average non-private and private gradients over subset Z of D at iteration t (the iteration number is dropped for ease of notation).

The result in Theorem 5.1 follows from a second order Taylor expansion of the non-private and private ERM functions $\mathcal{L}(\boldsymbol{\theta}_t - \eta \boldsymbol{g}_B; D_a)$ and $\mathcal{L}(\boldsymbol{\theta}_t - \eta(\bar{\boldsymbol{g}}_B + \mathcal{N}(0, \boldsymbol{I} C^2 \sigma^2); D_a)$, respectively, around $\boldsymbol{\theta}_t$ and by comparing their differences. Once again, proofs are reported in Section 3.11.1.

The first term in the expression (Equation (4)) denotes the Taylor approximation of the (non-private) SGD loss. Terms (R_a^{clip}) and (R_a^{noise}) quantify, together, the excessive risk for group a. The last term $O(\|\boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t\|^3)$ captures for negligible higher order components. Therein, (R_a^{clip}) quantifies the effect of clipping to the excessive risk, and (R_a^{noise}) quantifies the effect of perturbing the average gradients to the excessive risk. Therefore, Theorem 5.1 shows that there are two main sources of disparate impact in DP-SGD training:

1. *Gradient clipping* (R_a^{clip}): which, in turn, depends of three factors: **(i)** The values of the Hessian matrix $\boldsymbol{H}_\ell^a$ of the loss function associated with group a; **(ii)** The gradients values $\boldsymbol{g}_{D_a}$ associated with the samples of group a; and **(iii)** The clipping bound C, which appears in $\bar{\boldsymbol{g}}_B$ and $\bar{\boldsymbol{g}}_D$.

2. *Noise addition* (R_a^{noise}): which, in turn, depends on two factors: **(i)** The values of the (trace of the) Hessian matrix $\boldsymbol{H}_\ell^a$ of the loss function associated with group a; and **(ii)** The privacy loss parameters $(\epsilon, \delta, \Delta_\ell)$ (which, in turn, are characterized by the noise variance $C^2\sigma^2$).

A schematic representation of these factors is shown in Figure 3.2. Therein, X_{D_a} denotes the features values $X \in \mathcal{X}$ of the subset D_a of D. *Theorem 5.1 entails that unfairness occurs whenever different groups have different values for any of the gradient clipping and noise addition excessive risk terms.*

The next sections analyze the reasons behind the disparity in excessive risk focusing, independently, on terms R_a^{clip} (Section 3.7) and R_a^{noise} (Section 3.8). Independently studying these terms is motivated by observation that the clipping value C regulates the dominance of a factor over the other. Indeed, for sufficiently large (small) C values R_a^{noise} will dominate (be dominated by) R_a^{clip}.[3]

3.7 Why gradient clipping causes unfairness?

As highlighted above, there are three factors influencing the clipping effect to the excessive risk R_a^{clip}: the *Hessian loss*, the *gradient values*, and the *clipping bound*. This section illustrates their

[3]This observation relates with the bias-variance trade-off typically observed in DP-SGD [128].

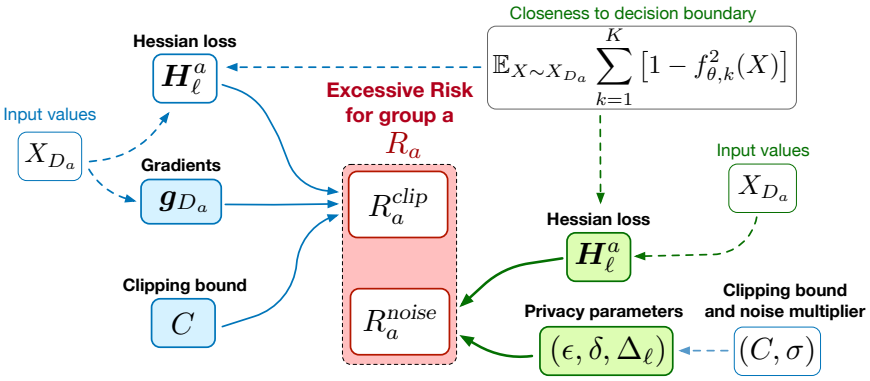


Fig. 3.2: Diagram of the factors affecting the excessive risk R_a for a group $a \in \mathcal{A}$ of individuals. Components affecting R_a in output perturbation involve exclusively the green boxes while those affecting R_a in DP-SGD involve both green and blue boxes. The main *direct* factors (e.g., those appearing in Eq. (4)) affecting the excessive risk clipping R^{clip}_a and noise R^{noise}_a components are highlighted within colored boxes. These direct factors are also regulated by *latent* factors, shown in white boxes, with dotted lines illustrating dependencies.

dependencies with the excessive risk, provides conditions to compare the disparate impacts between different groups, and shows the presence of an extra (latent) factor: the norm of the *input values* X_{D_a}, which plays a role to this disparate impacts by indirectly controlling the norms of gradient $\boldsymbol{g}_{D_a}$ (see the diagram illustrated in Figure 3.2).

The next results assume that the empirical loss function $\mathcal{L}(\boldsymbol{\theta}; D_a)$, associated with each group $a \in \mathcal{A}$, is convex and β_a-smooth. The analysis also consider learning rates $\eta \leq {}^{1}/_{\max_a \beta_a}$ and gradients $\boldsymbol{g}(B)$ and $\bar{\boldsymbol{g}}(B)$ with small variances. Note that this is not restrictive as the variance decreases as a function of the batch size B. Finally, for notational convenience, and w.l.o.g., the result focus on the case in which $|\mathcal{A}| = 2$. As shown in the empirical assessment (see Section 3.11.3), however, the conclusions carry on even in cases when the above assumptions may not hold.

Theorem 3.3. *Let* $p_z = {}^{|D_z|}/_{|D|}$ *be the fraction of training samples in group* $z \in \mathcal{A}$*. For groups* $a, b \in \mathcal{A}$*,* $R^{clip}_a > R^{clip}_b$ *whenever:*

$$\|\boldsymbol{g}_{D_a}\| \frac{p_a^2}{2} \geq \frac{5}{2} C + \|\boldsymbol{g}_{D_b}\| \left(1 + p_b + \frac{p_b^2}{2}\right). \tag{3.5}$$

Theorem 3.3 provides a sufficient condition for which a group may have larger excessive risk

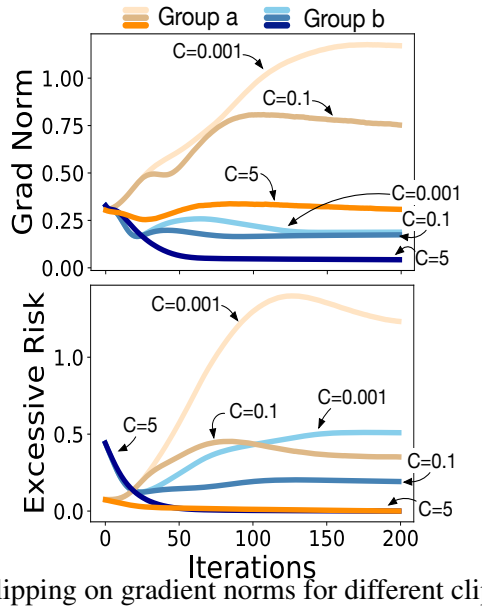


Fig. 3.3: Impact of gradient clipping on gradient norms for different clipping bounds on Bank dataset.

than another solely based on the clipping term analysis. *It relates unfairness with the average (non-private) gradient norms of the groups* $\boldsymbol{g}_{D_a}$ *and* $\boldsymbol{g}_{D_b}$ *and the clipping value* C. As shown in the diagram of Figure 3.2, this result relates two main factors to the excessive risk due to clipping $\boldsymbol{R}_a^{clip}$: **(1)** the *clipping bound* C, and **(2)** the (norm of the) *gradients* $\|\boldsymbol{g}_{D_a}\|$. While the *relative dataset size* $p_a = {}^{|D_a|}/_{|D|}$ of each group also appears in Equation (3.5), our extensive experiments showed that this factor may not play a prime role in controlling the disparate impacts (see Section 3.11.3).

The relation with these two factors is illustrated in Figure 3.3, which shows the impact of gradient clipping (for different C values) to the gradient norms (top) and to the excessive risk R_a (bottom). It shows that the gradient norms reduce as C increases and that the group with larger gradient norms have also larger excessive risk.

Finally, the diagram in Figure 3.2 also shows the presence of an additional factor affecting the gradient norms: *the input norms*, whose average is denoted $X_{D_a} = \mathbb{E}_{X\sim X_{D_a}}\|X\|$, in the figure. While this aspect is not directly evident in Theorem 3.3, the following examples highlight the positive correlation between input and gradients norms when considering a linear classifier and a feedforward neural network.

Example 3.1. *Consider the ERM problem* (6.1) *for a linear classifier* $f_{\boldsymbol{\theta}}(X) \stackrel{def}{=} softmax(\boldsymbol{\theta}^T X)$

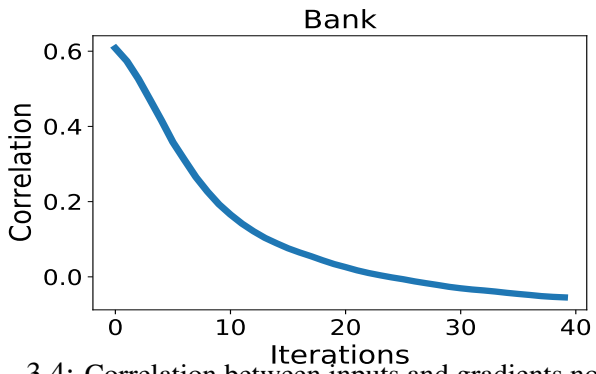


Fig. 3.4: Correlation between inputs and gradients norms.

and cross-entropy loss $\ell(f_{\boldsymbol{\theta}}(X), Y) = -\sum_{i=1}^{K} Y_i \log \boldsymbol{f}_{\boldsymbol{\theta}}^i(X)$ *where* K *is the number of classes. The gradient of the loss function at a given data point* (X, Y) *is:* $\boldsymbol{g}_X = \nabla\boldsymbol{\theta}\ell(f_{\boldsymbol{\theta}}(X), Y) = (\boldsymbol{Y} - \boldsymbol{f}) \otimes X$. *The result is by [21] and it suggests that the gradient norms are proportional to the input norms:* $\|\boldsymbol{g}_X\| \propto \|X\|$.

Example 3.2. *Next, consider a neural network with single hidden layer,* $f_{\boldsymbol{\theta}}(X) \stackrel{def}{=}$ *softmax* $\left(\boldsymbol{\theta}_1^T o(\boldsymbol{\theta}_2^T X)\right)$, *where* $o(\cdot)$ *is a proper activation function and* $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$ *are the model parameters. It can be seen that* $\|g_X\| \propto \|\nabla_{\boldsymbol{\theta}_1}\ell(f_{\boldsymbol{\theta}}(X), Y)\| + \|\nabla_{\boldsymbol{\theta}_2}\ell(f_{\boldsymbol{\theta}}(X), Y)\|$, *where* $\|\nabla_{\boldsymbol{\theta}_2}\ell(f_{\boldsymbol{\theta}}(X), Y)\| \propto \|X\|$. *The full derivations are reported in Section 3.11.8.*

Both examples illustrate a correlation between the gradients norms $\|\boldsymbol{g}_X\|$ and input norms $\|X\|$ for a given data sample X. This behavior is also illustrated in Figure 3.4, which highlights a positive correlation between the individual inputs and the gradients norms obtained while privately training a simple neural network (with one hidden layer) using DP-SGD on the Bank dataset. The experiment use $C = 0.1$ and $\sigma = 1$. The correlation decreases during training since the gradients norms reduce as training advances.

These observations imply that group data with large input norms—typically defining the tail of data distribution—result in large gradient norms and, thus, as shown in Theorem 3.3, may have larger disproportionate impacts than groups with smaller input norms, under DP-SGD. This analysis is in alignment with the empirical observation raised in [12], showing that samples at the tail of a distribution may experience larger accuracy losses, in private training, with respect to other samples.

While the above shows a dependency between gradients and clipping bound, as illustrated in the (R_a^{clip}) equation, the group excessive risk is also affected by the Hessian values. However, as shown in Section 3.11.3, the Hessian factor is almost always dominated by the other factors examined in this section. This is due to the presence of the multiplier $\eta^2/2$ which attenuate the impact of the Hessian value to the excessive risk due to clipping in conjunction with the smoothness assumptions, which prevents the Hessian values to grow too large.

In summary, the main factors affecting R_a^{clip} for a group $a \in \mathcal{A}$ are the norm of the group gradients $\boldsymbol{g}_{D_a}$, in turn controlled by the norm of the inputs X_{D_a}, and the clipping bound C.

3.8 Why noise addition causes unfairness?

Next, the chapter analyzes the factors influencing the noise effect to the excessive risk R_a^{noise}, which, as highlighted in Theorem 5.1, for DP-SGD and Theorem 3.1 for output perturbation, are the *Hessian loss*, and the *privacy loss parameters* $(\epsilon, \delta, \Delta_\ell)$ (see also Figure 3.2). Noting that the privacy parameters have a multiplicative effect on the Hessian loss (see Equations (R_a^{noise}) and (3.3)), the following analysis, treats them as constants, and restricts focus on the effects of the Hessian trace to the disparate impacts.

The following result provide a condition to compare the disparate impacts between different groups,

Theorem 3.4. *For groups $a, b \in \mathcal{A}$, $R_a^{noise} > R_b^{noise}$ whenever*

$$\mathrm{Tr}(H_\ell^a) > \mathrm{Tr}(H_\ell^b).$$

Note the connection of the result above with Theorem 3.1. Additionally, as illustrated in the diagram of Figure 3.2 the Hessian trace for a group is controlled by two (latent) factors: **(1)** The average distance of the group data to the decision boundary, and **(2)** The values of the group input norms. While these aspects are not directly evident in Theorem 3.9, the following highlights the positive correlation between these two factors and the Hessian Traces.

Example 3.3. *Consider the same setting of Example 3.1. The Hessian of the cross entropy loss of a sample $X \sim D$ is given by $H_\ell^X = [\big(diag(\boldsymbol{f}) - \boldsymbol{f}\boldsymbol{f}^T\big) \otimes XX^T]$, where $\otimes$ is the Kronecker product [21]. This result suggests that the trace of the Hessian for sample X is proportional to its input norm: $\mathrm{Tr}(H_\ell^X) \propto \|X\|^2$. Additionally it also shows that: $\mathrm{Tr}(H_\ell^X) \propto (1 - \sum_{k=1}^K \boldsymbol{f}_{\boldsymbol{\theta},k}^2(X))$, where K is the number of classes, whose term is connected to the distance to the decision boundary, as shown next.*

The following result highlights the connection between the term $(1 - \sum_{k=1}^K \boldsymbol{f}_{\boldsymbol{\theta},k}^2(X))$ and the distance of sample X to the decision boundary.

Theorem 3.5. *Consider a K-class classifier $\boldsymbol{f}_{\boldsymbol{\theta},k}$ ($k \in [K]$). For a given sample $X \sim D$, the term $\left(1 - \sum_{k=1}^K \boldsymbol{f}_{\boldsymbol{\theta},k}^2(X)\right)$ is maximized when $\boldsymbol{f}_{\boldsymbol{\theta},k}(X) = 1/K$ and minimized when $\exists k \in [K]$ s.t. $\boldsymbol{f}_{\boldsymbol{\theta},k}(X) = 1$ and $\boldsymbol{f}_{\boldsymbol{\theta},k'} = 0\ \forall k' \in [K], k \neq k$.*

That is, the term $\left(1 - \sum_{k=1}^K \boldsymbol{f}_{\boldsymbol{\theta},k}^2(X)\right)$ is maximized (minimized) when the sample X is close (far) to the decision boundary. Since, as shown in Example 3.3 this term can be proportional to the Hessian trace, then the aforementioned relation also indicates a connection between the Hessian trace value for a sample and its distance to the decision boundary: The closest (farther) is a sample X to the decision boundary the larger (smaller) is the associated Hessian trace value $\mathrm{Tr}(\boldsymbol{H}_\ell^X)$. This is intuitive as the model decision are less robust to the presence of noise in the model (e.g., as that introduced by a DP mechanism) for the samples which are close to the decision boundary w.r.t. those which are far from it.

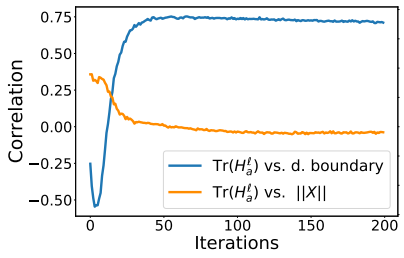


Fig. 3.5: Correlation between trace of Hessian with closeness to boundary (dark color) and input norm (light color).

An analogous behavior is also observed in Neural Networks and described in Section 3.11.8 due to space constraints. Figure 3.5 illustrates this behavior using the same setting adopted in Figure 3.4. It highlights the positive correlation between the input norm, the trace of Hessian, and the closeness to the decision boundary for a given sample X.

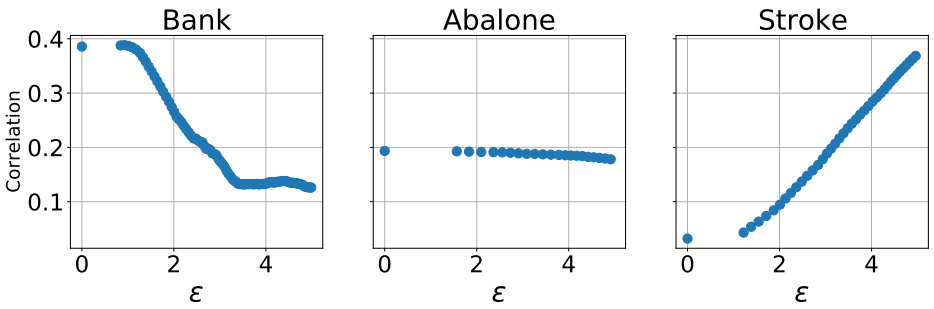


Fig. 3.6: Correlation between input norms and excessive risk; DP-SGD with $C = 0.1$ and $\sigma = 1.0$.

While the above discusses the relation between input norms and Hessian losses, Figure 3.6 illustrates this dependencies with the excessive risk, which is one of the main objective of the analysis, on three datasets. *Once again this observation recognizes the difference in input norms as a crucial proxy to unfairness: Groups with larger input norms will tend to have larger disproportionate impacts under private training than groups with smaller input norms.*

In summary, the main factor affecting R_a^{noise} for a group $a \in \mathcal{A}$ is the Hessian loss $\boldsymbol{H}_\ell^a$, which, in turn, is controlled by the group's distance to the decision boundary and by their inputs norm.

3.9 Mitigation solution

The previous sections showed that, in DP-SGD, the excessive risk R_a for a group $a \in \mathcal{A}$ could be decomposed into two factors R_a^{clip}, due to clipping, and R_a^{noise}, due to noise addition. In turn, it identified the gradients values $\boldsymbol{g}_{D_a}$ associated with the samples of group a and the clipping bound C as the main sources of disparate impact in component R_a^{clip}, and the (trace of the) Hessian $\boldsymbol{H}_\ell^a$ of the group a loss function as the main source of disparate impact in component R_a^{noise}.

A solution to mitigate the effects of these components to the excessive risk gap is to equalize the factors responsible for R_a^{clip} and R_a^{noise} among all group $a \in \mathcal{A}$ during private training. The

resulting empirical risk loss becomes:

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; D) + \sum_{a \in \mathcal{A}} \left(\gamma_1 \left|\langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_D, \boldsymbol{g}_D - \bar{\boldsymbol{g}}_D \rangle\right| + \gamma_2 \left|\mathrm{Tr}(\boldsymbol{H}_\ell^a) - \mathrm{Tr}(\boldsymbol{H}_\ell)\right|\right), \tag{3.6}$$

where the component multiplied by γ_1 comes for simplifying the expression $|\,\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D\rangle - \langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_D\rangle - \langle \boldsymbol{g}_D, \boldsymbol{g}_D\rangle - \langle \boldsymbol{g}_D, \bar{\boldsymbol{g}}_D\rangle\,|$ associated to the empirical risk gap ξ_a of the main factor affecting R_a^{clip}, and component multiplied by γ_2 by the analogous expression for the main factor affecting R_a^{noise}. Note that this last component involves computing the Hessian matrices of the loss functions during each training step, which is a computationally expensive process. The previous section, however, showed a strong dependency between the trace of the Hessian losses and the distance to the decision boundary (Theorem 3.5). Thus, in place of Equation (3.6) the proposed mitigating solution solves:

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; D) + \sum_{a \in \mathcal{A}} \left(\gamma_1 \left|\langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_D, \boldsymbol{g}_D - \bar{\boldsymbol{g}}_D \rangle\right| + \gamma_2 \left|\mathbb{E}_{X\sim D_a}[1 - \sum_{k=1}^{K} f_{\theta,k}^2(X)] - \mathbb{E}_{X\sim D}[1 - \sum_{k=1}^{K} f_{\theta,k}^2(X)]\right|\right).$$

Figure 3.7 illustrates this approach at work, for various multipliers γ_1 and γ_2 on the Bank dataset with two protected group (blue = majority; orange = minority). Similar trends are shown for other datasets as well in Section 3.11.3. The implementation uses a neural network with a single hidden layer and Suppose uses DP-SGD with $C = 0.1, \sigma = 5.0$. A clear trend arises: For appropriately selected values γ_1 and γ_2 the excessive risk gap between the majority and minority groups not only tends to be equalized, but it also decreases significantly for both groups. *These results imply that the proposed mitigating strategy may not only improve fairness but also the loss in utility of the private models.*

3.10 Limitations and conclusions

This work was motivated by the recent observations regarding the disparate impacts induced by DP in learning systems. The chapter introduced a notion of fairness that relies on the concept of ex-

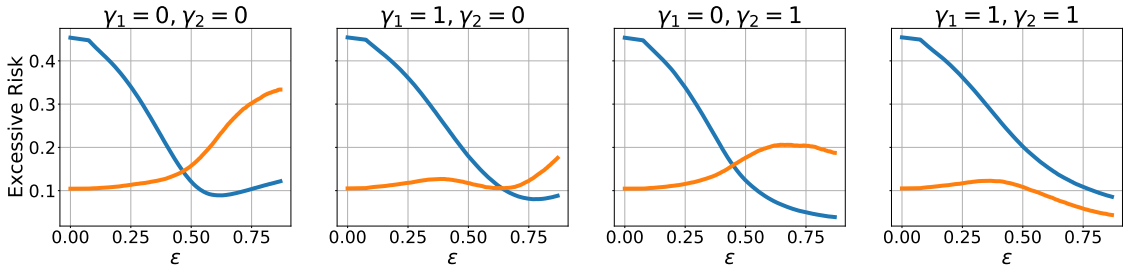


Fig. 3.7: Mitigating solution: Excessive risk gap at varying of the privacy loss ϵ on the Bank dataset for different values of γ_1 and γ_2. Majority (minority) group is shown in dark (light) colors.

cessive risk, analyzed this fairness notion in output perturbation and DP-SGD for ERM problems, it isolated the relevant components related with noise addition and gradient clipping responsible for the disparate impacts, studied the main factors affecting these components, and introduced a mitigation solution.

This study recognizes the following limitations: Firstly, the analyses in Section 3.7 requires the ERM losses to be smooth and convex. While these are common assumptions adopted in the analysis of private ERM [28, 142], the generalization to the non-convex case is an interesting open question. The second limitation regards the selection of the multipliers γ_1 and γ_2 in Equation 3.6. While the chapter does not investigate how to optimally selecting these values, the adoption of a Lagrangian Dual framework, as in [128], could a useful tool to the automatic selection of such parameters, for an extra privacy cost. Finally, the proposed mitigation solution negatively affects the training runtime and the design of more efficient solutions and implementations is an interesting challenge.

Despite these limitations, given the increasingly key role of differential privacy in machine learning, we believe that this work may represent an important and broadly useful step toward understanding the roots of the disparate impacts observed in differentially private learning systems.

3.11 Appendix

3.11.1 Missing proofs

Theorem 3.6. *Let ℓ be a twice differentiable and convex loss function and consider the output perturbation mechanism described above. Then, the excessive risk gap for group $a \in \mathcal{A}$ is approximated by:*

$$\xi_a \approx \frac{1}{2}\Delta_\ell^2 \sigma^2 \left|\mathrm{Tr}(\boldsymbol{H}_\ell^a) - \mathrm{Tr}(\boldsymbol{H}_\ell)\right|, \tag{3}$$

where $\boldsymbol{H}_\ell^a = \nabla^2_{\boldsymbol{\theta}^*} \sum_{(X,A,Y)\in D_a} \ell(f_{\boldsymbol{\theta}^*}(X), Y)$ is the Hessian matrix of the loss function at the optimal parameters vector $\boldsymbol{\theta}^*$, computed using the group data D_a, $\boldsymbol{H}^\ell$ is the analogous Hessian computed using the population data D, and $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix.

Proof. Recall that the output perturbation mechanism adds Gaussian noise directly to the non-private model parameters $\boldsymbol{\theta}^*$ to obtain the private parameters $\tilde{\boldsymbol{\theta}}$. Denote $\psi \sim \mathcal{N}(0, \mathbf{I}\Delta_\ell^2\sigma^2)$ the random noise vector with the same size as $\boldsymbol{\theta}^*$. Then $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta}^* + \psi$. Using a second order Taylor expansion around $\boldsymbol{\theta}^*$ the private risk function for group $a \in \mathcal{A}$ is approximated as follows:

$$\mathcal{L}(\tilde{\boldsymbol{\theta}}, D_a) = \mathcal{L}(\boldsymbol{\theta}^* + \psi, D_a) \approx \mathcal{L}(\boldsymbol{\theta}^*, D_a) + \psi^T \nabla_{\boldsymbol{\theta}^*} \mathcal{L}(\boldsymbol{\theta}^*, D_a) + \frac{1}{2}\psi^T \boldsymbol{H}_\ell^a \psi. \tag{3.7}$$

Taking the expectation with respect to ψ on both sides of the above equation results in:

$$\begin{aligned}
\mathbb{E}\left[\mathcal{L}(\tilde{\boldsymbol{\theta}}, D_a)\right] &\approx \mathbb{E}\left[\mathcal{L}(\boldsymbol{\theta}^*, D_a)\right] + \mathbb{E}\left[\psi^T \nabla_{\boldsymbol{\theta}^*} \mathcal{L}(\boldsymbol{\theta}^*, D_a)\right] + \frac{1}{2}\mathbb{E}\left[\psi^T \boldsymbol{H}_\ell^a \psi\right] && \text{(3.8a)}\\
&= \mathcal{L}(\boldsymbol{\theta}^*, D_a) + \frac{1}{2}\mathbb{E}\left[\psi^T \boldsymbol{H}_\ell^a \psi\right] && \text{(3.8b)}\\
&= \mathcal{L}(\boldsymbol{\theta}^*, D_a) + \frac{1}{2}\sum_{i,j}\mathbb{E}\left[\psi_i (\boldsymbol{H}_\ell^a)_{ij} \psi_j\right] && \text{(3.8c)}\\
&= \mathcal{L}(\boldsymbol{\theta}^*, D_a) + \frac{1}{2}\sum_{i}\mathbb{E}\left[\psi_i^2\right] (\boldsymbol{H}_\ell^a)_{ii} && \text{(3.8d)}\\
&= \mathcal{L}(\boldsymbol{\theta}^*, D_a) + \frac{1}{2}\Delta_\ell^2\sigma^2 \,\mathrm{Tr}\left(\boldsymbol{H}_\ell^a\right), && \text{(3.8e)}
\end{aligned}$$

where equation (3.8b) follows from linearity of expectation, by observing that $\nabla_{\boldsymbol{\theta}^*} \mathcal{L}(\boldsymbol{\theta}^*, D_a)$ is

a constant term, and that ψ is a 0-mean noise variable, thus, $\mathbb{E}[\psi] = \mathbf{0}^T \times \nabla_{\boldsymbol{\theta}^*}\mathcal{L}(\boldsymbol{\theta}^*, D_a) = \mathbf{0}^T$. Equation (3.8c) follows by definition of Hessian matrix, where $(H_\ell^a)_{ij}$ denotes the entry with indices i and j of the matrix. Equation (3.8d) follows from that $\psi_i \perp \psi_j$, for all $i \neq j$, and Equation (3.8e) from that for a random variable X, $\mathbb{E}[X^2] = (\mathbb{E}[X])^2 + \mathrm{Var}[X]$, and $\mathrm{Var}[\psi_i] = \Delta_\ell^2\sigma^2\ \forall i$ and definition of Trace of a matrix.

Therefore, the group and population excessive risks are approximated as:

$$R_a(\boldsymbol{\theta}) = \mathbb{E}\left[\mathcal{L}(\tilde{\boldsymbol{\theta}}, D_a)\right] - \mathcal{L}(\boldsymbol{\theta}^*, D_a) \approx \frac{1}{2}\Delta_\ell^2\sigma^2\,\mathrm{Tr}\left(\boldsymbol{H}_\ell^a\right) \tag{3.9}$$

$$R(\boldsymbol{\theta}) = \mathbb{E}\left[\mathcal{L}(\tilde{\boldsymbol{\theta}}, D)\right] - \mathcal{L}(\boldsymbol{\theta}^*, D) \approx \frac{1}{2}\Delta_\ell^2\sigma^2\,\mathrm{Tr}\left(\boldsymbol{H}_\ell\right). \tag{3.10}$$

The claim follows by definition of excessive risk gap (Equation 3.2) subtracting Equation (4.41) from (3.10) in absolute values. □

Corollary 3.3. *Consider the ERM problem for a linear model $f_{\boldsymbol{\theta}}(X) \stackrel{def}{=} \boldsymbol{\theta}^T X$, with L_2 loss function i.e., $\ell(f_{\boldsymbol{\theta}}(X), Y) = (f_{\boldsymbol{\theta}}(X) - Y)^2$. Then, output perturbation does not guarantee pure fairness.*

Proof. First, notice that for an L_2 loss function the trace of Hessian loss for a group $a \in \mathcal{A}$ is:

$$\mathrm{Tr}(\boldsymbol{H}_\ell^a) = \mathbb{E}_{x\sim D_a}\|X\|.$$

Therefore, from Theorem 3.6, the excessive risk gap ξ_a for group a is:

$$\xi_a \approx \frac{1}{2}\Delta_\ell^2\sigma^2\left|\mathbb{E}_{x\sim D_a}\|X\| - \mathbb{E}_{x\sim D}\|X\|\right|. \tag{3.11}$$

Notice that ξ_a is larger than zero only if the average input norm of group a is different with that of the population one. Since this condition cannot be guaranteed in general, the output perturbation mechanism for a linear ERM model under the L_2 loss does not guarantee pure fairness. □

Corollary 3.4. *If for any two groups $a, b \in \mathcal{A}$ their average group norms $\mathbb{E}_{X_a\sim D_a}\|X_a\| = \mathbb{E}_{X_b\sim D_b}\|X_b\|$ have identical values, then output perturbation with L_2 loss function provides pure*

fairness.

Proof. The above follows directly by observing that, when the average norms of any two groups have identical values, $\xi_a \approx 0$ for any group $a \in \mathcal{A}$ (see Equation (3.11)), and thus the average norm of each group also coincide with that of the population. □

The above indicates that as long as the average group norm is invariant across different groups, then output perturbation mechanism provides pure fairness.

Theorem 3.7. *Consider the ERM problem* (6.1) *with loss ℓ twice differentiable with respect to the model parameters. The expected loss $\mathbb{E}[\mathcal{L}(\boldsymbol{\theta}_{t+1}; D_a)]$ of group $a \in \mathcal{A}$ at iteration $t+1$, is approximated as:*

$$\mathbb{E}\left[\mathcal{L}(\boldsymbol{\theta}_{t+1}; D_a)\right] \approx \underbrace{\mathcal{L}(\boldsymbol{\theta}_t; D_a) - \eta \left\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D \right\rangle + \frac{\eta^2}{2}\mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B\right]}_{\textit{non-private term}} \tag{4}$$

$$+ \underbrace{\eta\left(\left\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D \right\rangle - \left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_D \right\rangle\right) + \frac{\eta^2}{2}\left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B\right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B\right]\right)}_{\textit{private term due to clipping}} \qquad (R_a^{\textit{clip}})$$

$$+ \underbrace{\frac{\eta^2}{2}\operatorname{Tr}(\boldsymbol{H}_\ell^a) C^2\sigma^2}_{\textit{private term due to noise}} \qquad (R_a^{\textit{noise}})$$

where the expectation is taken over the randomness of the private noise and the mini-batch selection, and the terms $\boldsymbol{g}_Z$ and $\bar{\boldsymbol{g}}_Z$ denote, respectively, the average non-private and private gradients over subset Z of D at iteration t (the iteration number is dropped for ease of notation).

Proof. The proof of Theorem 5.1 relies on the following two second order Taylor approximations: **(1)** The first approximates the ERM loss at iteration $t+1$ under non-private training, i.e., $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \boldsymbol{g}_B$, where $B \subseteq D$ denotes the minibatch. **(2)** The second approximates expected ERM loss under private-training, i.e $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta(\bar{\boldsymbol{g}}_B + \psi)$ where $\psi \sim \mathcal{N}(0, \boldsymbol{I}C^2\sigma^2)$. Finally, the result is obtained by taking the difference of these approximations under private and non-private training.

1. Non-private term. The non private term of Theorem 5.1 can be derived using second order Taylor approximation as follows:

$$\mathcal{L}(\theta_{t+1}, D_a) = \mathcal{L}(\theta_t - \eta \boldsymbol{g}_B, D_a) \approx \mathcal{L}(\theta_t, D_a) - \eta \langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_B \rangle + \frac{\eta^2}{2} \boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B \tag{3.12}$$

Taking the expectation with respect to the randomness of the mini-batch B selection on both sides of the above approximation, and noting that $\mathbb{E}[\boldsymbol{g}_B] = \boldsymbol{g}_D$ (as B is selected randomly from dataset D), it follows:

$$\mathbb{E}[\mathcal{L}(\theta_{t+1}, D_a)] \approx \mathcal{L}(\theta_t, D_a) - \eta \mathbb{E}[\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_B \rangle] + \frac{\eta^2}{2} \mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B] \tag{3.13a}$$

$$= \mathcal{L}(\theta_t, D_a) - \eta \langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D \rangle + \frac{\eta^2}{2} \mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B]. \tag{3.13b}$$

2. Private term (due to both clipping and noise). Consider the private update in DP-SGD, i.e., $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta(\bar{\boldsymbol{g}}_B + \psi)$. Again, applying a second order Taylor approximation around $\boldsymbol{\theta}_t$ allows us to estimate the expected private loss at iteration $t+1$ as:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\theta}_{t+1}, D_a) &= \mathcal{L}\left(\boldsymbol{\theta}_t - \eta(\bar{\boldsymbol{g}}_B + \psi), D_a\right) \\
&\approx \mathcal{L}\left(\boldsymbol{\theta}_t, D_a\right) - \eta \left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_B + \psi \right\rangle + \frac{\eta^2}{2} \left(\bar{\boldsymbol{g}}_B + \psi\right)^T \boldsymbol{H}_\ell^a \left(\bar{\boldsymbol{g}}_B + \psi\right) && \text{(3.14a)} \\
&= \mathcal{L}\left(\boldsymbol{\theta}_t, D_a\right) - \eta \left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_B \right\rangle - \eta \left\langle \boldsymbol{g}_{D_a}, \psi \right\rangle + \frac{\eta^2}{2} \bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B && \text{(3.14b)} \\
&\qquad + \frac{\eta^2}{2} \left(\psi^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B + \bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \psi + \psi^T \boldsymbol{H}_\ell^a \psi\right)
\end{aligned}$$

Taking the expectation with respect to the randomness of the mini-batch B selection and with

respect to the randomness of noise ψ on both sides of the above equation gives:

$$\mathbb{E}\left[\mathcal{L}(\boldsymbol{\theta}_{t+1}, D_a)\right] \approx \mathbb{E}\Big[\mathcal{L}\left(\boldsymbol{\theta}_t, D_a\right) - \eta\left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_B\right\rangle - \eta\left\langle \boldsymbol{g}_{D_a}, \psi\right\rangle + \frac{\eta^2}{2}\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B \tag{3.15a}$$

$$+ \frac{\eta^2}{2}\left(\psi^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B + \bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \psi + \psi^T \boldsymbol{H}_\ell^a \psi\right)\Big]$$

$$= \mathcal{L}(\boldsymbol{\theta}_t, D_a) - \eta\left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_B\right\rangle - \eta\left\langle \boldsymbol{g}_{D_a}, \mathbb{E}[\psi]\right\rangle + \frac{\eta^2}{2}\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B\right] \tag{3.15b}$$

$$+ \frac{\eta^2}{2}\Big(\mathbb{E}\left[\psi\right]^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B + \bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \mathbb{E}[\psi] + \mathbb{E}\left[\psi^T \boldsymbol{H}_\ell^a \psi\right]\Big)$$

$$= \mathcal{L}(\boldsymbol{\theta}_t, D_a) - \eta\left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_B\right\rangle + \frac{\eta^2}{2}\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B\right] + \frac{\eta^2}{2}\mathbb{E}\left[\psi^T \boldsymbol{H}_\ell^a \psi\right] \tag{3.15c}$$

$$= \mathcal{L}(\boldsymbol{\theta}_t, D_a) - \eta\left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_B\right\rangle + \frac{\eta^2}{2}\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B\right] + \frac{\eta^2}{2}\operatorname{Tr}\left(\boldsymbol{H}_\ell^a\right) C^2\sigma^2, \tag{3.15d}$$

where (3.15b), and (3.15c) follow from linearity of expectation and from that $\mathbb{E}[\psi] = 0$, since ψ is a 0-mean noise variable. Equation (3.15d) follows from that,

$$\mathbb{E}\left[\psi^T \boldsymbol{H}_\ell^a \psi\right] = \mathbb{E}\left[\sum_{i,j} \psi_i (\boldsymbol{H}_\ell^a)_{i,j} \psi_j\right] = \sum_i \mathbb{E}\left[\psi_i^2 (\boldsymbol{H}_\ell^a)_{i,i}\right] = \operatorname{Tr}\left(\boldsymbol{H}_\ell^a\right) C^2\sigma^2,$$

since $\mathbb{E}[\psi^2] = \mathbb{E}[\psi]^2 + \operatorname{Var}[\psi]$ and $\mathbb{E}[\psi] = 0$ while $\operatorname{Var}[\psi] = C^2\sigma^2$.

Note that in the above approximation (Equation (3.15)), the component

$$\mathcal{L}(\boldsymbol{\theta}_t, D_a) - \eta\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_B\rangle + \frac{\eta^2}{2}\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B\right] \tag{3.16}$$

is associated to the SGD update step in which gradients have been clipped to the clipping bound value C, i.e. $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta(\bar{\boldsymbol{g}}_B)$.

Next, the component

$$\frac{\eta^2}{2}\operatorname{Tr}\left(\boldsymbol{H}_\ell^a\right) C^2\sigma^2 \tag{3.17}$$

is associated to the SGD update step in which the noise ψ is added to the gradients.

If we take the difference between the approximation associated with the non-private loss term, obtained in Equation 3.13b, with that associated with the private loss term, obtained in Equation

3.15d, we can derive the effect of a single step of (private) DP-SGD compared to its non-private counterpart:

$$\mathbb{E}\left[\mathcal{L}(\boldsymbol{\theta}_{t+1}; D_a)\right] \approx \mathcal{L}(\boldsymbol{\theta}_t; D_a) - \eta \left\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D \right\rangle + \frac{\eta^2}{2}\mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B\right] \tag{3.18a}$$

$$+ \eta \left(\left\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D \right\rangle - \left\langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_D \right\rangle\right) + \frac{\eta^2}{2}\left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B\right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B\right]\right) \tag{3.18b}$$

$$+ \frac{\eta^2}{2}\operatorname{Tr}\left(\boldsymbol{H}_\ell^a\right) C^2 \sigma^2. \tag{3.18c}$$

In the above,

- The components in Equation (3.18a) are associated with the loss under non-private training (see again Equation 3.13b);
- The components in Equation (3.18b) is associated with for excessive risk due to gradient clipping;
- Finally, the components in Equation (3.18c) is associated with the excessive risk due to noise addition.

□

Next, the chapter proves Theorem 3.8. This result is based on the following assumptions.

Assumption 3.1. *[Convexity and Smoothness assumption] For a group $a \in \mathcal{A}$, its empirical loss function $\mathcal{L}(\boldsymbol{\theta}, D_a)$ is convex and β_a-smooth.*

Assumption 3.2. *Let $B \subseteq D$ be a subset of the dataset D, and consider a constant $\varepsilon \geq 0$. Then, the variance associated with the gradient norms of a random mini-batch B, $\sigma_B^2 = \operatorname{Var}\left[\|\boldsymbol{g}_B\|\right] \leq \varepsilon$ as well as that associated with its clipped counterpart, $\bar{\sigma}_B^2 = \operatorname{Var}\left[\|\bar{\boldsymbol{g}}_B\|\right] \leq \varepsilon$.*

The assumption above can be satisfied when the mini-batch size is large enough. For example, the variance is 0 when $|B| = |D|$.

Assumption 3.3. *The learning rate used in DP-SGD η is upper bounded by quantity $1/\max_{z \in \mathcal{A}} \beta_z$.*

Theorem 3.8. *Let $p_z = {}^{|D_z|}/_{|D|}$ be the fraction of training samples in group $z \in \mathcal{A}$. For groups $a, b \in \mathcal{A}$, $R_a^{clip} > R_b^{clip}$ whenever:*

$$\|\boldsymbol{g}_{D_a}\| \frac{p_a^2}{2} \geq \frac{5}{2}C + \|\boldsymbol{g}_{D_b}\| \left(1 + p_b + \frac{p_b^2}{2}\right). \tag{5}$$

To ease notation, the statement of the theorem above uses $\varepsilon = 0$ (See Assumption 3.2) but the theorem can be generalized to any $\varepsilon \geq 0$.

The following Lemmas are introduced to aid the proof of Theorem 3.8.

Lemma 3.1. *Consider the ERM problem* (6.1) *solved with DP-SGD with clipping value C. The following average clipped per-sample gradients $\bar{\boldsymbol{g}}_Z$, where $Z \subseteq D$, has norm at most C.*

Proof. The result follows by triangle inequality:

$$\begin{aligned}
\|\bar{\boldsymbol{g}}_{D_Z}\| &= \left\| \frac{1}{|D_Z|} \sum_{i \in D_Z} \bar{\boldsymbol{g}}_i \right\| \\
&\leq \frac{1}{|D_Z|} \sum_{i \in D_Z} \|\bar{\boldsymbol{g}}_i\| \\
&= \frac{1}{|D_Z|} \sum_{i \in D_Z} \left\| \boldsymbol{g}_i \min\left(1, \frac{C}{\|\boldsymbol{g}_i\|}\right) \right\| \\
&\leq \frac{1}{|D_Z|} \sum_{i \in D_Z} C = C.
\end{aligned}$$

□

The next Lemma derives a lower and an upper bound for the component $\mathbb{E}[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B] - \mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B]$, which appears in the excessive risk term due to clipping R_a^{clip} for some group $a \in \mathcal{A}$.

Lemma 3.2. *Consider the ERM problem* (6.1) *with loss ℓ, solved with DP-SGD with clipping value C. Further, let $\varepsilon = 0$ (see Assumption 3.2). For any group $a \in \mathcal{A}$, the following inequality holds:*

$$-\beta_a \|\boldsymbol{g}_D\|^2 \leq \mathbb{E}[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B] - \mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B] \leq \beta_a C^2 \tag{3.19}$$

Proof. Consider a group $a \in \mathcal{A}$. By the convexity assumption of the loss function, the Hessian $\boldsymbol{H}_\ell^a$ is a positive semi-definite matrix, i.e., for all real vectors of appropriate dimensions $\boldsymbol{v}$, it follows that $\boldsymbol{v}^T \boldsymbol{H}_\ell^a \boldsymbol{v} \geq 0$.

Therefore, for a subset $B \subseteq D$ the following inequalities hold:

- $\bar{\boldsymbol{g}}_B \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B \geq 0$,
- $\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B \geq 0$.

Additionally their expectations $\mathbb{E}[\bar{\boldsymbol{g}}_B \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B]$ and $\mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B]$ are non-negative.

By the smoothness property of the loss function, $\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B \leq \beta_a \|\bar{\boldsymbol{g}}_B\|^2$, thus:

$$\mathbb{E}[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B] \leq \beta_a \mathbb{E}\left[\|\bar{\boldsymbol{g}}_B\|^2\right] \tag{3.20a}$$

$$= \beta_a \big(\mathbb{E}[\|\bar{\boldsymbol{g}}_B\|]^2 + \mathrm{Var}\left[\|\bar{\boldsymbol{g}}_B\|\right]\big) \tag{3.20b}$$

$$\leq \beta_a \big(C^2 + \bar{\sigma}_B^2\big) \tag{3.20c}$$

$$\leq \beta_a (C^2 + \varepsilon), \tag{3.20d}$$

where Equation (3.20b) follows from that $\mathbb{E}[X^2] = (\mathbb{E}[X])^2 + \mathrm{Var}[X]$, Equation (3.20c) is due to Lemma 3.1, and finally, the last inequality is due to Assumption 3.2.

Therefore, since $\varepsilon = 0$ by assumption of the Lemma, the following upper bound holds:

$$\mathbb{E}[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B] - \mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B] \leq \beta_a C^2. \tag{3.21}$$

Next, notice that

$$\mathbb{E}[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B] - \mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B] \geq -\mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B] \tag{3.22a}$$

$$\geq -\mathbb{E}[\beta_a \|\boldsymbol{g}_B\|^2] \tag{3.22b}$$

$$= -\beta_a \left(\mathbb{E}[\|\boldsymbol{g}_B\|]^2 + \mathrm{Var}\left[\|\boldsymbol{g}_B\|\right]\right) \tag{3.22c}$$

$$= -\beta_a \|\boldsymbol{g}_D\|^2, \tag{3.22d}$$

where the inequality in Equation (3.22a) follows since both terms on the left hand side of the Equation are non negative. Equation (3.22b) follows by smoothness assumption of the loss function. Equation (3.22c) follows by definition of expectation of a random variable, since $\mathbb{E}[X]^2 = \mathbb{E}[X^2] + \mathrm{Var}[X]$. Finally, Equation (3.22d) follows from that $\mathrm{Var}[\boldsymbol{g}_B] \leq \varepsilon = 0$ by Assumption 3.2, and that $\varepsilon = 0$ by assumption of the Lemma, and thus the norms $\|g_B\| = \|g_D\|$ and, thus, $\mathbb{E}[\boldsymbol{g}_B] = \boldsymbol{g}_D$. Therefore if follows:

$$-\beta_a \|\boldsymbol{g}_D\|^2 \leq \mathbb{E}[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B] - \mathbb{E}[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B]. \tag{3.23}$$

which concludes the proof. □

Again, the above uses $\varepsilon = 0$ to simplify notation, but the results generalize to the case when $\varepsilon > 0$. In such a case, the bounds require slight modifications to involve the term ε.

Lemma 3.3. *Let $a, b \in \mathcal{A}$ be two groups. Consider the ERM problem* (6.1) *solved with DP-SGD with clipping value C and learning rate $\eta \leq 1/\max_{a\in\mathcal{A}} \beta_a$. Then, the difference on the excessive risk due to clipping $R^a_{clip} - R^b_{clip}$ is lower bounded as:*

$$R^a_{clip} - R^b_{clip} \geq \eta \left(\langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_{D_b}, \boldsymbol{g}_D - \bar{\boldsymbol{g}}_D \rangle - \frac{1}{2}(\|\boldsymbol{g}_D\|^2 + C^2) \right). \tag{3.24}$$

Proof. Recall that $B \subseteq D$ is the mini-batch during the resolution of DP-SGD. Using the lower and

upper bounds obtained from Lemma 3.2, it follows:

$$
\begin{aligned}
R^a_{clip} - R^b_{clip} &= \eta\left(\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D\rangle - \langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_D\rangle\right) + \frac{\eta^2}{2}\left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}^a_\ell \bar{\boldsymbol{g}}_B\right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}^a_\ell \boldsymbol{g}_B\right]\right) && \text{(3.25a)}\\
&\quad - \eta\left(\langle \boldsymbol{g}_{D_b}, \boldsymbol{g}_D\rangle - \langle \boldsymbol{g}_{D_b}, \bar{\boldsymbol{g}}_D\rangle\right) - \frac{\eta^2}{2}\left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}^b_\ell \bar{\boldsymbol{g}}_B\right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}^b_\ell \boldsymbol{g}_B\right]\right)\\
&= \eta\langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_{D_b}, \boldsymbol{g}_D - \bar{\boldsymbol{g}}_D\rangle + \frac{\eta^2}{2}\left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}^a_\ell \bar{\boldsymbol{g}}_B\right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}^a_\ell \boldsymbol{g}_B\right]\right) && \text{(3.25b)}\\
&\quad - \frac{\eta^2}{2}\left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}^b_\ell \bar{\boldsymbol{g}}_B\right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}^b_\ell \boldsymbol{g}_B\right]\right)\\
&\geq \eta\langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_{D_b}, \boldsymbol{g}_D - \bar{\boldsymbol{g}}_D\rangle - \frac{\eta^2}{2}\beta_a\|\boldsymbol{g}_D\|^2 - \frac{\eta^2}{2}\beta_b C^2 && \text{(3.25c)}\\
&\geq \eta\langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_{D_b}, \boldsymbol{g}_D - \bar{\boldsymbol{g}}_D\rangle - \frac{\eta^2}{2}\max_{z\in\mathcal{A}}\beta_z\big(\|\boldsymbol{g}_D\|^2 + C^2\big) && \text{(3.25d)}\\
&\geq \eta\left(\langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_{D_b}, \boldsymbol{g}_D - \bar{\boldsymbol{g}}_D\rangle - \frac{1}{2}(\|\boldsymbol{g}_D\|^2 + C^2)\right), && \text{(3.25e)}
\end{aligned}
$$

where the inequality (3.25c) follows as a consequence of Lemma 3.2, and the inequality (3.25e) since $\eta \leq \frac{1}{\max_{a\in\mathcal{A}}\beta_a}$. □

Proof of Theorem 3.8. We want to show that $R^a_{clip} > R^b_{clip}$ given Equation (4.27). Since, by Lemma 3.3 the difference $R^a_{clip} - R^b_{clip}$ is lower bounded – see Equation (3.24), the following shows that the right hand side of Equation (3.24) is positive, that is:

$$
\langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_{D_b}, \boldsymbol{g}_D - \bar{\boldsymbol{g}}_D\rangle - \frac{1}{2}\left(\|\boldsymbol{g}_D\|^2 + C^2\right) > 0. \tag{3.26}
$$

First, observe that the gradients at the population level can be expressed as a combination of the gradients of the two groups a and b in the dataset: $\boldsymbol{g}_D = p_a\boldsymbol{g}_{D_a} + p_b\boldsymbol{g}_{D_b}$ and $\bar{\boldsymbol{g}} = p_a\bar{\boldsymbol{g}}_{D_a} + p_b\bar{\boldsymbol{g}}_{D_b}$.

By algebraic manipulation, and the above, Equation (3.26) can thus be expressed as:

$$
\begin{aligned}
(3.26) &= \langle \boldsymbol{g}_{D_a} - \boldsymbol{g}_{D_b}, p_a\boldsymbol{g}_{D_a} + p_b\boldsymbol{g}_{D_b} - p_a\bar{\boldsymbol{g}}_{D_a} - p_b\bar{\boldsymbol{g}}_{D_b}\rangle - \frac{1}{2}\big(\|\boldsymbol{g}_{D_a}p_a + \boldsymbol{g}_{D_b}p_b\|^2 + C^2\big) && \text{(3.27a)}\\
&= (p_a\|\boldsymbol{g}_{D_a}\|^2 + p_b\boldsymbol{g}_{D_a}^T\boldsymbol{g}_{D_b} - p_a\boldsymbol{g}_{D_a}^T\bar{\boldsymbol{g}}_{D_a} - p_b\boldsymbol{g}_{D_a}^T\bar{\boldsymbol{g}}_{D_b} - p_a\boldsymbol{g}_{D_b}^T\boldsymbol{g}_{D_a} - p_b\|\boldsymbol{g}_{D_b}\|^2 && \text{(3.27b)}\\
&\quad + p_a\boldsymbol{g}_{D_b}^T\bar{\boldsymbol{g}}_{D_a} + p_b\boldsymbol{g}_{D_b}^T\bar{\boldsymbol{g}}_{D_b} - \frac{1}{2}\big(p_a^2\|\boldsymbol{g}_{D_a}\|^2 + 2p_ap_b\boldsymbol{g}_{D_a}\boldsymbol{g}_{D_b} + p_b^2\|\boldsymbol{g}_{D_b}\|^2 + C^2\big).
\end{aligned}
$$

Noting that for any vector $\boldsymbol{x}, \boldsymbol{y}$ the following inequality hold: $\boldsymbol{x}^T\boldsymbol{y} \geq -\|\boldsymbol{x}\|\|\boldsymbol{y}\|$, all the inner products in the above expression can be replaced by their lower bounds:

$$
\begin{aligned}
(3.26) &\geq \|\boldsymbol{g}_{D_a}\| \left(\|\boldsymbol{g}_{D_a}\| p_a (1 - \frac{p_a}{2}) - p_b \|\boldsymbol{g}_{D_b}\| - p_a C - p_b C - p_a \|\boldsymbol{g}_{D_b}\| - p_a p_b \|\boldsymbol{g}_{D_b}\| \right) && \text{(3.28a)} \\
&\quad - \|\boldsymbol{g}_{D_b}\| - p_a p_b \|\boldsymbol{g}_{D_b}\| \left(\|\boldsymbol{g}_{D_b}\| p_b (1 + \frac{p_b}{2}) + p_a C + p_b C \right) - \frac{1}{2} C^2 \\
&= \|\boldsymbol{g}_{D_a}\| \left(\|\boldsymbol{g}_{D_a}\| p_a (1 - \frac{p_a}{2}) - p_a p_b \|\boldsymbol{g}_{D_b}\| (p_b + p_a)(\|\boldsymbol{g}_{D_b}\| + C) \right) && \text{(3.28b)} \\
&\quad - \|\boldsymbol{g}_{D_b}\| \left(\|\boldsymbol{g}_{D_b}\| p_b (1 + \frac{p_b}{2}) + (p_a + p_b) C \right) - \frac{1}{2} C^2 \\
&= \|\boldsymbol{g}_{D_a}\| \left(\|\boldsymbol{g}_{D_a}\| p_a (1 - \frac{p_a}{2}) - p_a p_b \|\boldsymbol{g}_{D_b}\| - \|\boldsymbol{g}_{D_b}\| - C) \right) - \|\boldsymbol{g}_{D_b}\| \left(\|\boldsymbol{g}_{D_b}\| p_b (1 + \frac{p_b}{2}) + C \right) - \frac{1}{2} C^2 && \text{(3.28c)}
\end{aligned}
$$

where the last equality is because $p_a + p_b = 1$, by assumption of the dataset having exactly two groups.

By theorem assumption, $\|\boldsymbol{g}_{D_a}\| \frac{p_a^2}{2} \geq \frac{5}{2} C + \|\boldsymbol{g}_{D_b}\| (1 + p_b + \frac{p_b^2}{2})$. It follows that $\|\boldsymbol{g}_{D_a}\| > \|\boldsymbol{g}_{D_b}\|$ and $\|\boldsymbol{g}_{D_a}\| > C$. Combined with Equation (3.28c) it follows that:

$$
\begin{aligned}
(3.28c) &= \|\boldsymbol{g}_{D_a}\| \left(\|\boldsymbol{g}_{D_a}\| p_a (1 - \frac{p_a}{2}) - p_a p_b \|\boldsymbol{g}_{D_b}\| - \|\boldsymbol{g}_{D_b}\| - C - \|\boldsymbol{g}_{D_b}\| p_b (1 + \frac{p_b}{2}) - C \right) - \frac{1}{2} C^2 && \text{(3.29a)} \\
&\geq \|\boldsymbol{g}_{D_a}\| \left(\|\boldsymbol{g}_{D_a}\| p_a (1 - \frac{p_a}{2}) - p_a p_b \|\boldsymbol{g}_{D_a}\| - 2C - \|\boldsymbol{g}_{D_b}\| (1 + p_b + \frac{p_b^2}{2}) \right) - \frac{1}{2} C^2 && \text{(3.29b)} \\
&\geq \|\boldsymbol{g}_{D_a}\| \left(\|\boldsymbol{g}_{D_a}\| p_a (1 - \frac{p_a}{2} - p_b) - 2C - \|\boldsymbol{g}_{D_b}\| (1 + p_b + \frac{p_b^2}{2}) \right) - \frac{1}{2} C^2 && \text{(3.29c)} \\
&= \|\boldsymbol{g}_{D_a}\| \left(\|\boldsymbol{g}_{D_a}\| \frac{p_a^2}{2} - 2C - \|\boldsymbol{g}_{D_b}\| (1 + p_b + \frac{p_b^2}{2}) \right) - \frac{1}{2} C^2 && \text{(3.29d)} \\
&\geq \|\boldsymbol{g}_{D_a}\| \frac{C}{2} - \frac{1}{2} C^2 && \text{(3.29e)} \\
&> 0, && \text{(3.29f)}
\end{aligned}
$$

where the last equality is because $\|\boldsymbol{g}_{D_a}\| > C$. □

Theorem 3.9. *For groups $a, b \in \mathcal{A}$, $R_a^{noise} > R_b^{noise}$ whenever*

$$\mathrm{Tr}(\boldsymbol{H}_\ell^a) > \mathrm{Tr}(\boldsymbol{H}_\ell^b).$$

Proof. Suppose $\mathrm{Tr}(\boldsymbol{H}_\ell^a) > \mathrm{Tr}(\boldsymbol{H}_\ell^b)$. By definition of R_a^{noise} and R_b^{noise} from Theorem 5.1 it follows that:

$$R_a^{noise} = \frac{\eta^2}{2}\,\mathrm{Tr}(\boldsymbol{H}_\ell^a)C^2\sigma^2 > \frac{\eta^2}{2}\,\mathrm{Tr}(\boldsymbol{H}_\ell^b)C^2\sigma^2 = R_b^{noise},$$

which concludes the proof.

□

Theorem 3.10. *Consider a K-class classifier $\boldsymbol{f}_{\boldsymbol{\theta},k}$ ($k \in [K]$). For a given sample $X \sim D$, the term $\left(1 - \sum_{k=1}^K \boldsymbol{f}_{\boldsymbol{\theta},k}^2(X)\right)$ is maximized when $\boldsymbol{f}_{\boldsymbol{\theta},k}(X) = 1/K$ and minimized when $\exists k \in [K]$ s.t. $\boldsymbol{f}_{\boldsymbol{\theta},k}(X) = 1$ and $\boldsymbol{f}_{\boldsymbol{\theta},k'} = 0\ \forall k' \in [K], k' \neq k$.*

Proof. Fix an input X of D and denote $y_k = \boldsymbol{f}_{\boldsymbol{\theta},k}(X) \in [0, 1]$. Recall that y_k represents the likelihood of the prediction of input X to be associated with label k.

Note that, by Cauchy–Schwarz inequality

$$1 - \sum_{k=1}^K y_k^2 \leq 1 - K\left(\frac{\sum_i^K y_k}{K}\right)^2 \tag{3.30a}$$

$$= 1 - \frac{1}{K}, \tag{3.30b}$$

where Equation (3.30b) follows since $\sum_i^K y_k(X) = 1$. The above expression is maximized when

$$y_k = \boldsymbol{f}_{\theta,k}(X) = \frac{1}{K}.$$

Additionally, since $y_k \in [0, 1]$ it follows that $y_k^2 \leq y_k$. Hence,

$$1 - \sum_{k=1}^{K} y_k^2 \geq 1 - \sum_{i=1}^{K} y_k = 0. \tag{3.31}$$

To hold, the equality above, it must exists $k \in [K]$ such that $y_k = \boldsymbol{f}_{\theta,k}(X) = 1$ and for any other $k' \in [K]$ with $k' \neq k$, $y_{k'} = \boldsymbol{f}_{\theta,k'} = 0$. □

Given the connection of the term $1 - \sum_{k=1}^{K}(1 - f_{\theta,k}^2(X))$ and the associated (trace of the) Hessian loss $\boldsymbol{H}_f$, the result above suggests that the trace of the Hessian is minimized (maximized) when the classifier is very confident (uncertain) about the prediction of $X \sim D$, i.e., when X is far (close) to the decision boundary.

3.11.2 Experimental settings

Datasets The chapter uses the following UCI datasets to support its claims:

1. **Adult** (Income) dataset, where the task is to predict if an individual has low or high income, and the group labels are defined by race: *White* vs *Non-White* [18].

2. **Bank** dataset, where the task is to predict if a user subscribes a term deposit or not and the group labels are defined by age: *people whose age is less than 60 years old vs the rest* [86].

3. **Wine** dataset, where the task is to predict if a given wine is of good quality, and the group labels are defined by wine color: *red vs white* [18].

4. **Abalone** dataset, where the task is to predict if a given abalone ring exceeds the median value, and the group labels are defined by gender: *female vs male* [18].

5. **Parkinsons** dataset, where the task is to predict if a patient has total UPDRS score that exceeds the median value, and the group labels are defined by gender: *female vs male* [77].

6. **Churn** dataset, where the task is to predict if a customer churned or not. The group labels are defined by on gender: *female vs male* [31].

7. **Credit Card** dataset, where the task is to predict if a customer defaults a loan or not. The group labels are defined by gender: *female vs male* [23].

8. **Stroke** dataset, where the task is to predict if a patient have had a stroke based on their physical conditions. The group labels are defined by gender: *female vs male* [1].

All datasets were processed by standardization so each feature has zero mean and unit variance.

Settings For output perturbation, the chapter uses a Logistic regression model to obtain the optimal model parameters (we set the regularization parameter $\lambda = 1$) and add Gaussian noise to achieve privacy. The standard deviation of the noise required to the mechanism is determined following [13].

For DP-SGD, the chapter uses a neural network with single hidden layer with *tanh* activation function for the different datasets. The batch size $|B|$ is fixed to 32 and the learning rate $\eta = 1e-4$. Unless specified we set the clipping bound $C = 0.1$ and noise multiplier $\sigma = 5.0$. The experiments consider 100 runs of DP-SGD with different random seeds for each configuration. We employ the Tensorflow Privacy toolbox to compute the privacy loss ϵ spent during training.

Computing infrastructure All experiments were performed on a cluster equipped with Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz and 8GB of RAM.

Software and libraries All models and experiments were written in Python 3.7 and in Pytorch 1.5.0.

Code The code used for this submission is attached as supplemental material. All implementation of the experiments and proposed mitigation solution will be released upon publication.

3.11.3 Additional experiments

3.11.4 More on "Warm up: output perturbation"

Correlation between Hessian trace and excessive risk The following provides additional empirical support for the claims of the main chapter: *Groups with larger Hessian trace tend to*

have larger excessive risks in this subsection.

The experiments in this sub-section use output perturbation. Figure 3.8 reports the excessive risk and Hessian traces for the two groups defined in the datasets (as described in Section 3.11.2. The figure clearly illustrates that the groups with larger Hessian traces have larger excessive risk (i.e., experienced more unfairness) under private output perturbation when compared with the groups with smaller Hessian traces. These empirical findings are again a strong support for the claims of Theorem 3.1.

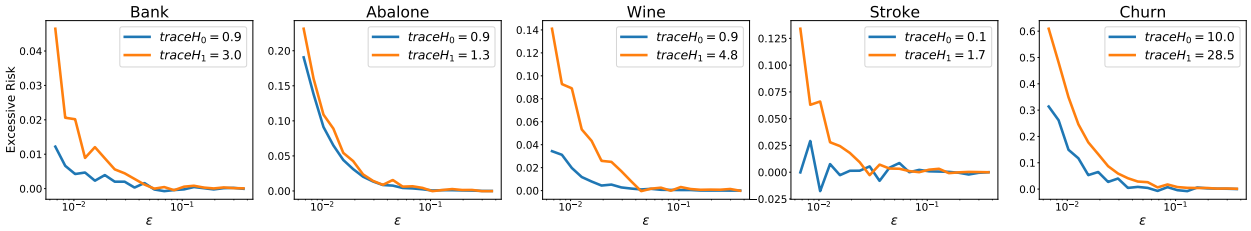


Fig. 3.8: Correlation between excessive risk gap and Hessian Traces at varying of the privacy loss ϵ.

Impact of data normalization by group The next results provide evidence to support the following claim raised in Section 3.5: *Given the impact of gradient norms to unfairness, normalizing data independently for each group can help improve fairness*. Figure 3.9 shows the evolution of the excessive risk R_a and R_b for the dataset groups during training. The top plots present the results with standard data normalization (e.g., each sample data is normalized independently from its group membership) while the bottom plots show the counterpart results for models trained when the data was normalized within the group datasets D_a and D_b. Note that the normalization adopted ensures that the data is 0-mean and of unit variance in each group dataset, which is a required condition to achieve the desired property.

The results clearly show that this strategy can not only reduce unfairness, but also the excessive risk gaps.

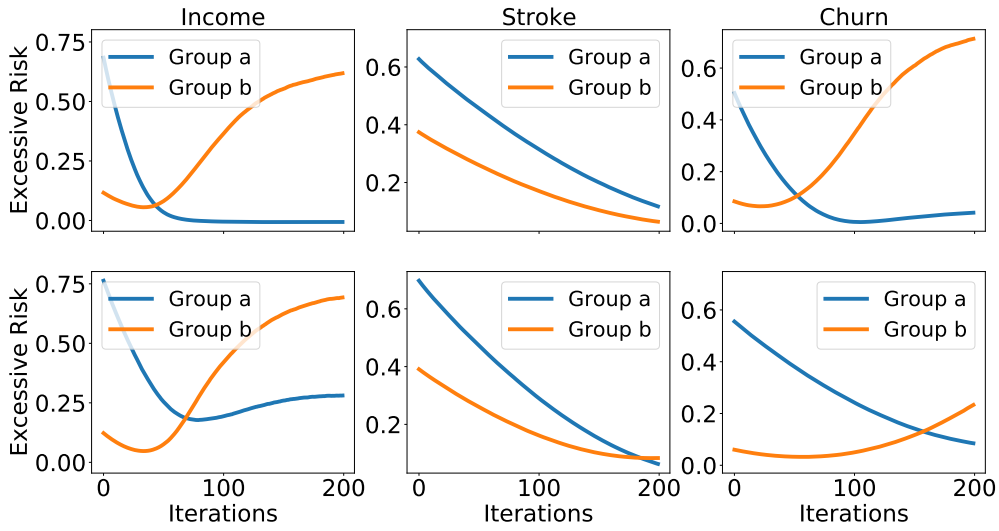
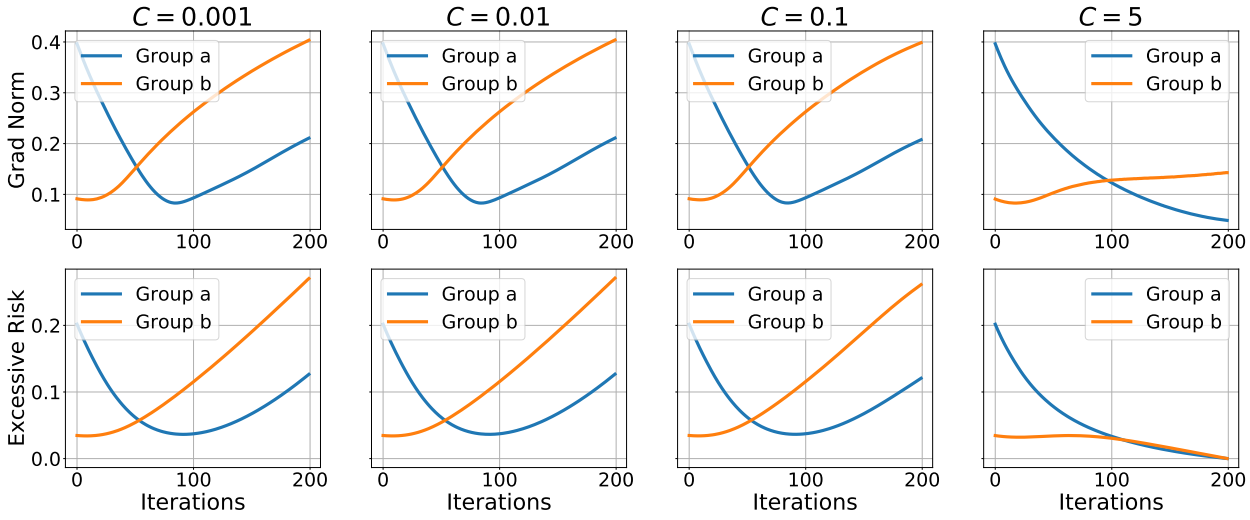


Fig. 3.9: Excessive risk for each group without group normalization (top) and with group normalization (bottom).

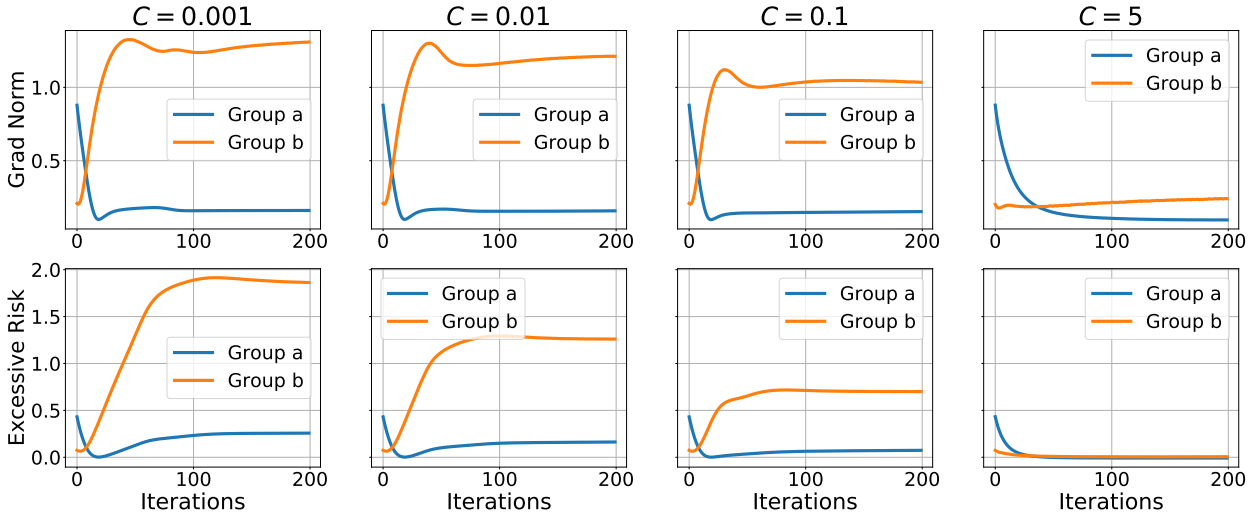
3.11.5 More on "Why gradient clipping causes unfairness?"

This section provides additional empirical evidence to support the claim made in Section 3.7 specifying the three direct factors influencing the clipping effect to the excessive risk: **(1)** the Hessian loss, **(2)** the gradient values, and **(3)** the clipping bound. Among these three factors, the gradient values and clipping bound are the dominant ones.

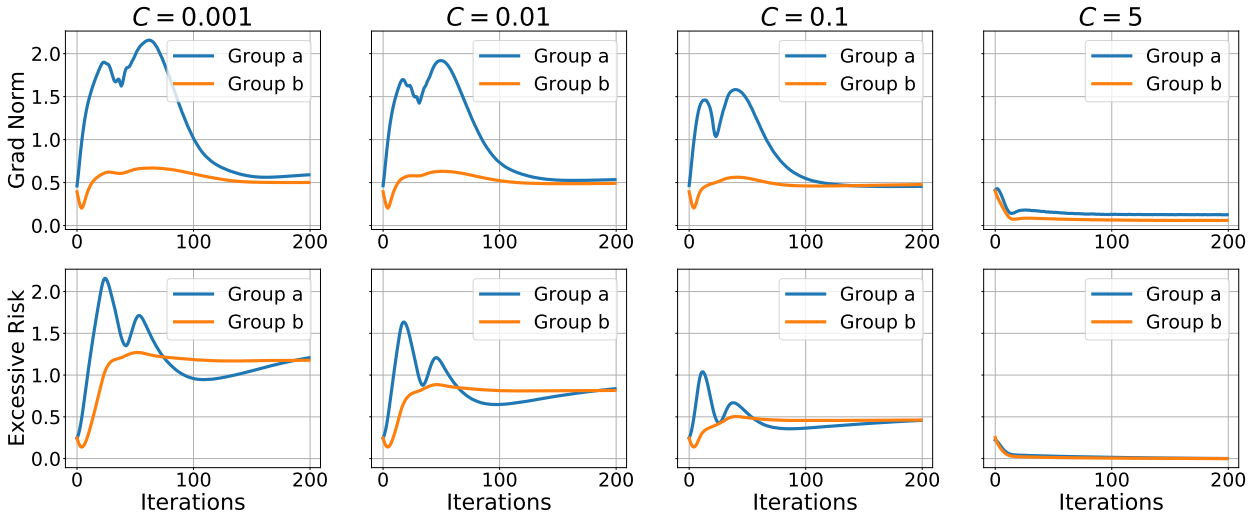
Impact of gradient values and clipping bound C Figure 3.10 provides the relation between the gradient norm and the different choices of clipping bounds to the excessive risks. The results are shown for the Abalone, Churn and Credit Card datasets. The experiments show that gradient norms reduce as C increases and that the group with larger gradient norms have also larger excessive risk. Similar results were achieved for other datasets as well (not reported to avoid redundancy).



(a) Abalone dataset



(b) Churn dataset



(c) Credit card dataset

Fig. 3.10: Impact of gradient clipping with different clipping bound values C to the excessive risk.

The Hessian loss is a minor impact factor to the excessive risk. As showed in the main text, the excessive risk associated to the gradient clipping for a particular group $a \in \mathcal{A}$ can be decomposed as:

$$R_a^{clip} = \eta \left(\langle \boldsymbol{g}_{D_a}, \boldsymbol{g}_D \rangle - \langle \boldsymbol{g}_{D_a}, \bar{\boldsymbol{g}}_D \rangle \right) + \frac{\eta^2}{2} \left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B \right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B \right] \right) \tag{3.32}$$

Denote $\psi_a = \left(\mathbb{E}\left[\bar{\boldsymbol{g}}_B^T \boldsymbol{H}_\ell^a \bar{\boldsymbol{g}}_B \right] - \mathbb{E}\left[\boldsymbol{g}_B^T \boldsymbol{H}_\ell^a \boldsymbol{g}_B \right] \right)$. This quantity clearly depends on the Hessian loss $\boldsymbol{H}_\ell^a$. However, under the assumptions in Theorem 3.8: convexity and smoothness of the loss function and the magnitude of the learning rate (i.e., that is small enough), the term ψ_a will be a negligible component in R_a^{clip}.

While this is evident under those assumption, our empirical analysis has reported a similar behavior for loss function for which those conditions do not generally apply. In the following experiment we run DP-SGD on a neural network with single hidden layer and tracked the values of R_a^{clip} and ψ_a for each group $a \in \mathcal{A}$ during private training. These values are reported in Figure 3.11 for different datasets. It can be seen that the components ψ_a (dotted lines) constitute a negligible amount to the excessive risk under gradient clipping R_a^{clip}.

Relative group data size is a minor impact factor to the excessive risk. Section 3.7 also observed that the relative group data size, p_b/p_a for two groups $a, b \in \mathcal{A}$ had a minor impact on unfairness. Figure 3.12 provides empirical evidence to support this observation. It shows the effects of varying the relative group data p_b/p_a to the gradient norms (top rows) and excessive risk (bottom rows) in three datasets: Abalone, Bank, and Income. The different relative group data ratios were obtained through subsampling. Notice that changing the relative group sizes does not result in a noticeable effect in the group gradient norms and excessive risk. These experiments demonstrate that the relative group data size might play a minor role in affecting unfairness.

These observation are also in alignment with the those raised by [43], who showed that the disparate impact of DP on model accuracy is not limited to highly imbalanced data and can occur in situations where the groups are slightly imbalanced.

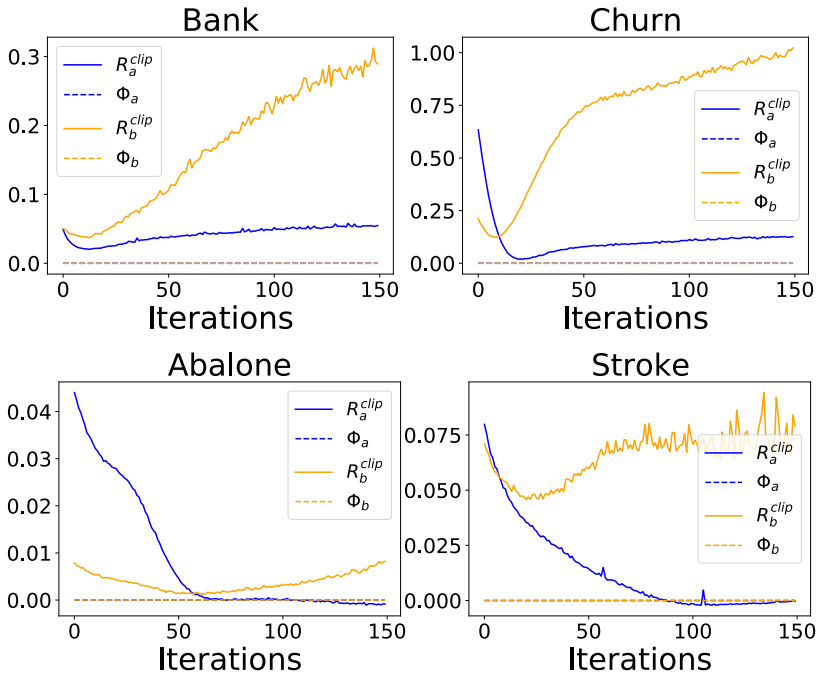


Fig. 3.11: Values of R_a^{clip} and ψ_a during private training for a neural network classifier.

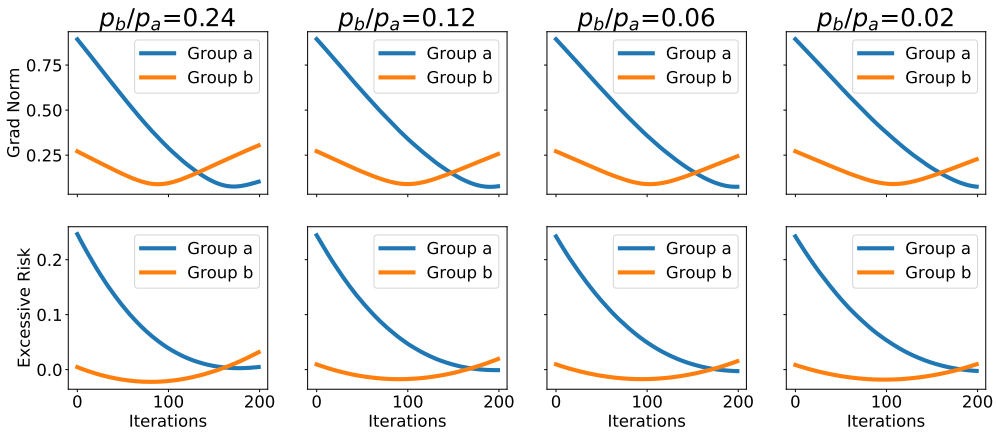
3.11.6 More on "Why noise addition causes unfairness?"

Figure 3.13 illustrates the connection between the trace of the Hessian of the loss function at some sample $X \in D$ and its distance to the decision boundary. The figure clearly show that the closest (father) is a sample X to the decision boundary, the larger (smaller) is the associated Hessian trace value $\mathrm{Tr}(\boldsymbol{H}_\ell^X)$. The experiments are reported for datasets Parkinson, Stroke, Wine, and Churn, but once again they extend to other datasets as well. b

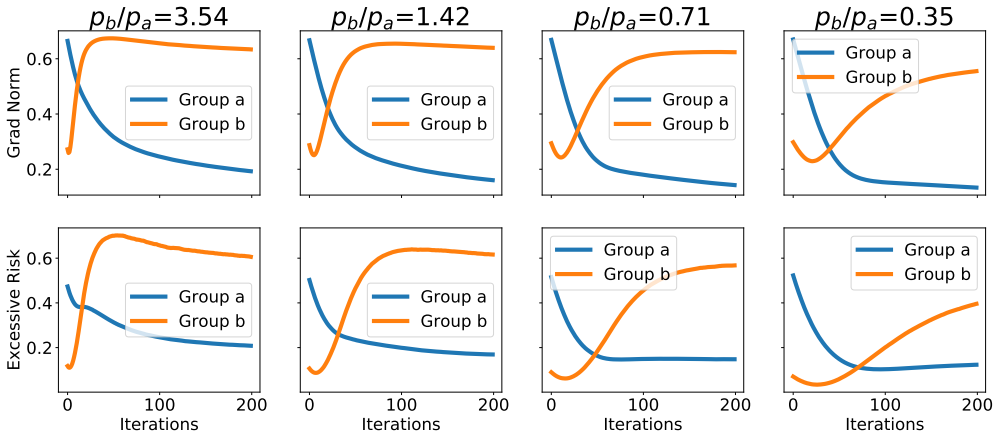
3.11.7 More on mitigation solutions

Next, this section demonstrates the benefits of the proposed mitigation solution on additional datasets. Figure 3.14 illustrates the excessive risk for each group in the reported datasets (recall that better fairness is achieved when the excessive risk curves values are small and similar) at varying of the privacy parameter ϵ (i.e., the excessive risk is tracked during private training).

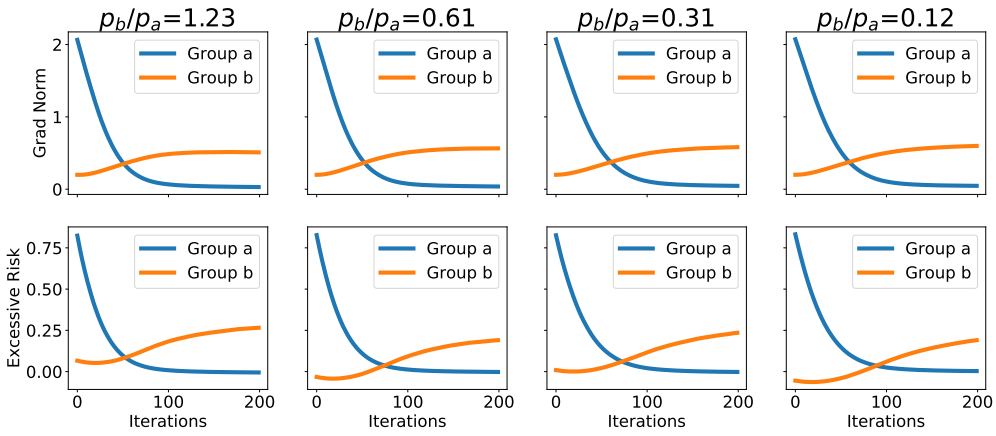
The leftmost column in each sub-figure present the results for the baseline model, which runs DP-SGD without the proposed fairness-mitigating constraints. Observe the positive effects in re-



(a) Abalone dataset



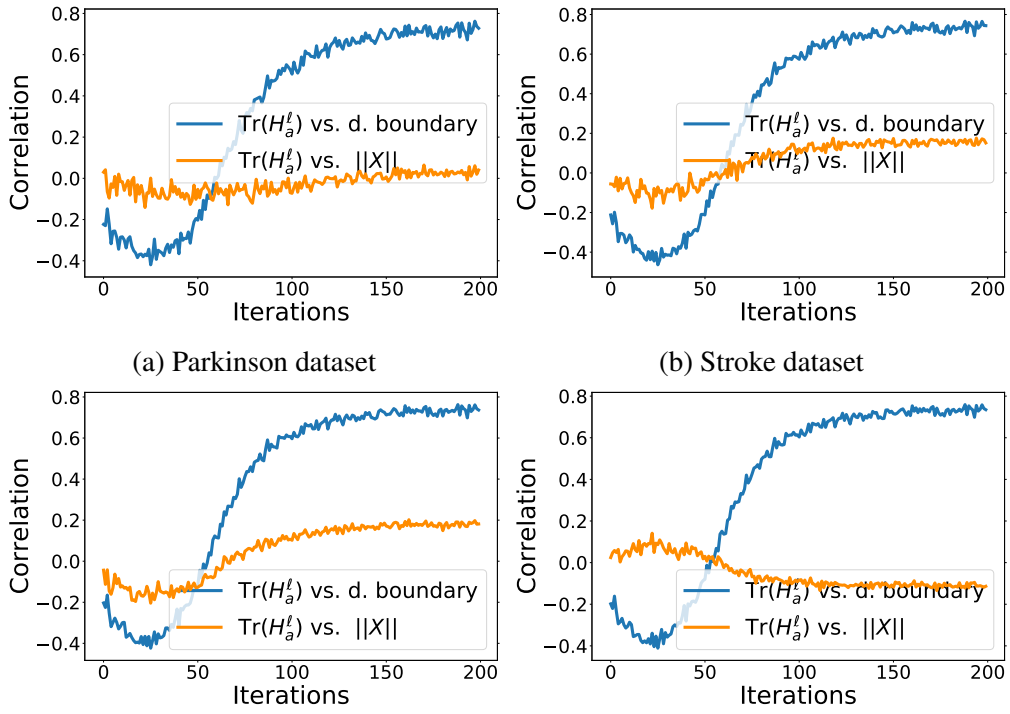
(b) Bank dataset



(c) Income dataset

Fig. 3.12: Impact of the relative group data size towards unfairness under DP-SGD (with $C = 0.1, \sigma = 5.0$).

ducing the inequality between the excessive risks between the groups when the solution activates both γ_1 (which regulates the component associated with R^{clip}) and γ_2 (which regulates the compo-



(a) Parkinson dataset

(b) Stroke dataset

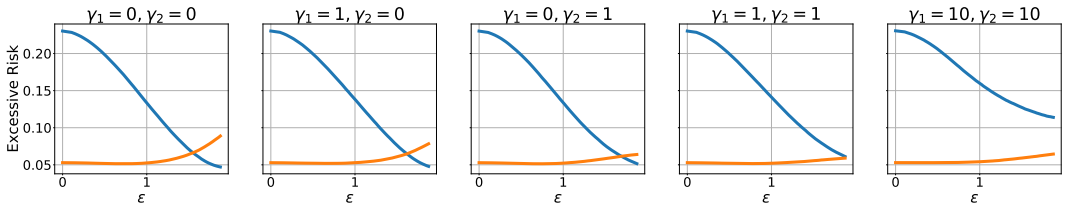
(c) Wine dataset

(d) Churn dataset

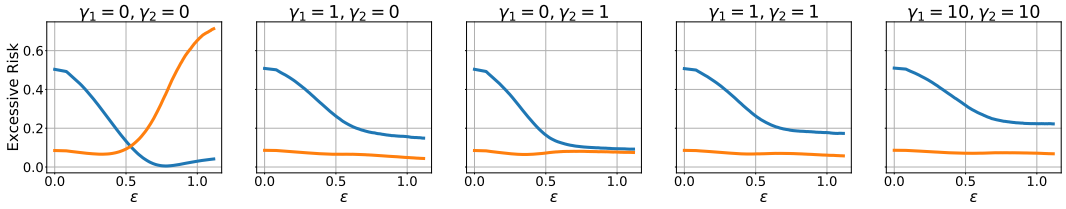
Fig. 3.13: Correlation between the trace of the Hessian of the loss function for a data sample X with its distance to the decision boundary (dark colors) and input norm (light colors).

nent associated with R^{noise}). In the reported experiments hyper-parameters $\gamma_1 = 1, \gamma_2 = 1$ were found to be good values for all our benchmark datasets. Smaller γ_1 and γ_2 values may not reduce unfairness. Likewise, large values could even exacerbate unfairness. Using the above setting, the proposed mitigation solution was able not only to reduce unfairness in 6 out 8 cases studied, but also to increase the utility of the private models.

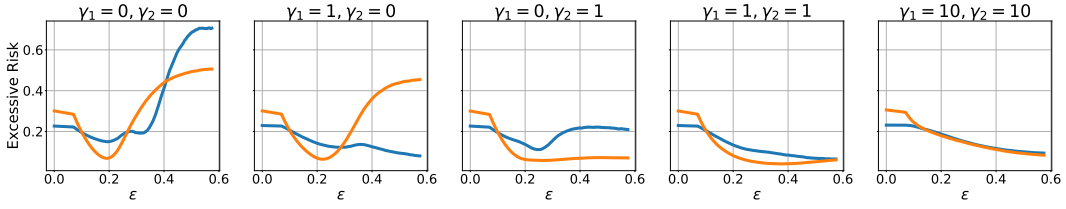
Once again, we mention that the design of optimal hyper-parameters is an interesting open challenge.



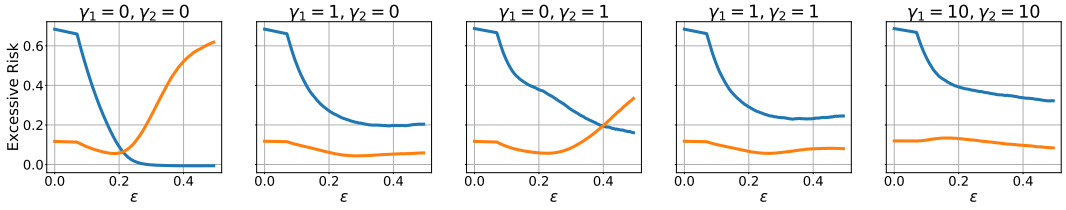
(a) Abalone dataset



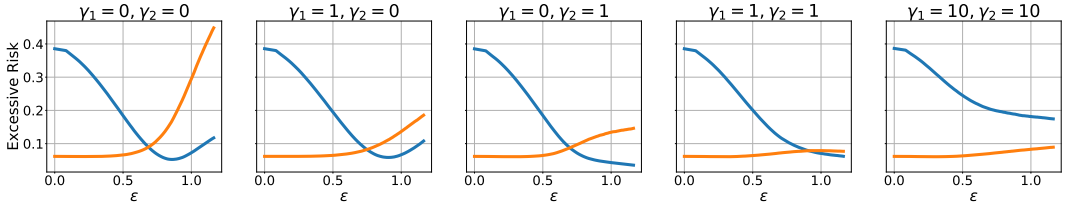
(b) Churn dataset



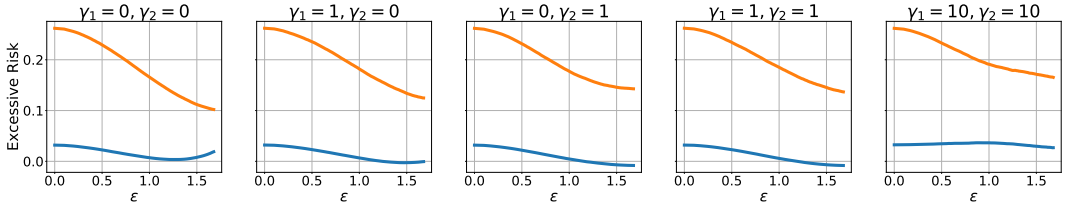
(c) Credit Card dataset



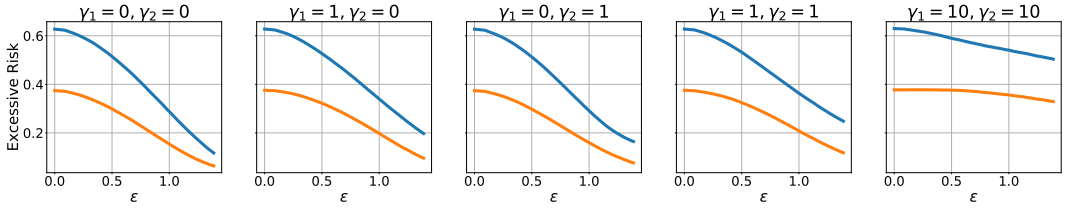
(d) Income dataset



(e) Wine dataset



(f) Parkinsons dataset



(g) Stroke dataset

Fig. 3.14: Mitigating solution: Excessive risk at varying of the privacy loss ϵ for different γ_1, γ_2.

3.11.8 Additional examples

3.11.9 More on gradient and Hessian loss of neural networks

This section focuses on two tasks: The first is to demonstrate the connection between the gradient norm $\|\boldsymbol{g}_X\|$ for some input X with its input norm $\|X\|$. The second is to demonstrate the relation between the trace of the Hessian loss at a sample X with input norm $\|X\|$ and the closeness of X to the decision boundary. We do so by providing a derivation of the gradients and the Hessian trace of a neural networks with one hidden layer.

Settings Consider a neural network model $f_{\boldsymbol{\theta}}(X) \stackrel{\text{def}}{=} \textit{softmax}\left(\boldsymbol{\theta}_1^T \sigma(\boldsymbol{\theta}_2^T X)\right)$ where $X \in \mathbb{R}^d, \theta_2 \in \mathbb{R}^{d\times H}, \theta_1 \in \mathbb{R}^{H\times K}$ and the cross-entropy loss $\ell(f_{\boldsymbol{\theta}}(X), Y) = -\sum_{k=1}^{K} Y_i \log \mathbf{f}_{\boldsymbol{\theta},\mathbf{k}}(\mathbf{X})$ where K is the number of classes, and $\sigma(\cdot)$ is the a proper activation function, e.g. a sigmoid function. Let $\boldsymbol{O} = \sigma(\boldsymbol{\theta}_2^T X) \in \mathbb{R}^H$ be the vector $(O_1, \ldots, O_H)$ of H hidden nodes of the network. Denote with $h_j = \sum_i \theta_{ji} X_i$ as the j-th hidden unit before the activation function. Next, denote $\boldsymbol{\theta}_{1,j,k} \in \mathbb{R}$ as the weight parameter that connects the j-th hidden unit h_j with the k-th output unit f_k and $\theta_{2,i,j} \in \mathbb{R}$ as the weight parameter that connects the i-th input X_i unit with the j-th hidden unit h_j.

Gradients Norm First notice that we can decompose the gradients norm of this neural network into two layers as follows:

$$\|\nabla_{\boldsymbol{\theta}} \ell(f_{\boldsymbol{\theta}}(X), Y)\|^2 = \|\nabla_{\boldsymbol{\theta}_1} \ell(f_{\boldsymbol{\theta}}(X), Y)\|^2 + \|\nabla_{\boldsymbol{\theta}_2} \ell(f_{\boldsymbol{\theta}}(X), Y)\|^2. \tag{3.33}$$

We will show that $\nabla_{\boldsymbol{\theta}_2} \ell(f_{\boldsymbol{\theta}}(X), Y)\| \propto \|X\|$.

Notice that:

$$\|\nabla_{\boldsymbol{\theta}_2} \ell(f_{\boldsymbol{\theta}}(X), Y)\|^2 = \sum_{i,j} \|\nabla_{\boldsymbol{\theta}_{2,i,j}} \ell(f_{\boldsymbol{\theta}}(X), Y)\|^2.$$

Applying, Equation (14) from [108], it follows that:

$$\nabla_{\theta_{2,i,j}} \ell(f_{\boldsymbol{\theta}}(X), Y) = \sum_{k=1}^{K} \left(Y_k - \boldsymbol{f}_{\theta,k}(X)\right) \theta_{1,j,k} \left(O_j(1 - O_j)\right) X_i, \tag{3.34}$$

which highlights the dependency of the gradient norm $\|\nabla_{\boldsymbol{\theta}_2} \ell(f_{\boldsymbol{\theta}}(X), Y)\|$ and the input norm $\|X\|^2$.

Hessian trace For the connections between the Hessian trace of the loss function at a sample X with the closeness of X to the decision boundary and the input norm $\|X\|$, the analysis follows the derivation provided by [16]. First, notice that:

$$\mathrm{Tr}(\boldsymbol{H}_{\ell}^{X}) = \mathrm{Tr}(\nabla^2_{\boldsymbol{\theta}_1} \ell(f_{\boldsymbol{\theta}}(X), Y)) + \mathrm{Tr}(\nabla^2_{\boldsymbol{\theta}_2} \ell(f_{\boldsymbol{\theta}}(X), Y)) \tag{3.35}$$

The following shows that:

1. $\mathrm{Tr}\left(\nabla^2_{\boldsymbol{\theta}_2} \ell(f_{\boldsymbol{\theta}}(X), Y)\right) \propto \left(1 - \sum_{k=1}^{K} \boldsymbol{f}^2_{\boldsymbol{\theta},k}(X)\right)$
2. $\mathrm{Tr}\left(\nabla^2_{\boldsymbol{\theta}_1} \ell(f_{\boldsymbol{\theta}}(X), Y)\right) \propto \|X\|^2$.

The former follows from Equation (26) of [16], since:

$$\nabla^2_{\boldsymbol{\theta}_{1,j,k}} \ell(f_{\boldsymbol{\theta}}(X), Y)) = f_k(1 - f_k) O_j^2, \tag{3.36}$$

and thus,

$$\mathrm{Tr}(\nabla^2_{\boldsymbol{\theta}_1} \ell(f_{\boldsymbol{\theta}}(X), Y)) = \sum_{j=1}^{H} \sum_{k=1}^{K} f_k(1 - f_k) O_j^2 = \sum_{j=1}^{H} \big(\sum_{k=1}^{K} f_k - \sum_{k=1}^{K} f_k^2 \big) O_j^2 = (1 - \sum_{k=1}^{K} f_k^2) \sum_{j=1}^{H} O_j^2.$$

The above shows the connection between the trace of Hessian loss at a sample X for the second layer of the neural network and the quantity $1 - \sum_{k=1}^{K} f_k^2(X)$ which measures how close is the sample X to the decision boundary. This result relates with Theorem 3.10.

Regarding point (2), by applying Equation (27) of [16] we obtain:

$$\nabla^2_{\boldsymbol{\theta}_{2,i,j}} \ell(f_{\boldsymbol{\theta}}(X), Y)) = X_i^2 \Gamma_j, \tag{3.37}$$

where $\Gamma_j = \sigma''(h_j) \sum_{k=1}^{K} \theta_{2,j,k}(Y_k - f_k) + \sigma'(h_j)^2 \sum_{k=1}^{K} \boldsymbol{\theta}^2_{2,j,k} f_k(1 - f_k)$, where σ' and σ'' are, respectively, the first and second derivative of the activation σ with respect to the hidden node h_j.

Thus:

$$\mathrm{Tr}(\nabla^2_{\boldsymbol{\theta}_2} \ell(f_{\boldsymbol{\theta}}(X), Y)) = \sum_{j=1}^{H} \sum_{i=1}^{d} \nabla^2_{\boldsymbol{\theta}_{2,i,j}} \ell(f_{\boldsymbol{\theta}}(X), Y) = \sum_{j=1}^{H} \left(\sum_{i=1}^{d} X_i^2 \right) \Gamma_j \propto \|X\|^2,$$

which shows the dependency of the trace of the Hessian of the loss function in the first layer at sample X and the data input norm.

CHAPTER 4
ON THE FAIRNESS IMPACTS OF PRIVATE ENSEMBLES MODELS

The Private Aggregation of Teacher Ensembles (PATE) is a machine learning framework that enables the creation of private models through the combination of multiple "teacher" models and a "student" model. The student model learns to predict an output based on the voting of the teachers, and the resulting model satisfies differential privacy. PATE has been shown to be effective in creating private models in semi-supervised settings or when protecting data labels is a priority. This chapter explores whether the use of PATE can result in unfairness, and demonstrates that it can lead to accuracy disparities among groups of individuals. The chapter also analyzes the algorithmic and data properties that contribute to these disproportionate impacts, why these aspects are affecting different groups disproportionately, and offers recommendations for mitigating these effects.

4.1 Introduction

The widespread adoption of machine learning (ML) systems in decision-making processes have raised concerns about bias and discrimination, as well as the potential for these systems to leak sensitive information about the individuals whose data is used as input. These issues are partic-

ularly relevant in contexts where ML systems are used to assist in decisions processes impacting individuals' lives, such as criminal assessment, lending, and hiring.

Differential Privacy (DP) [37] is an algorithmic property that bounds the risks of disclosing sensitive information of individuals participating in a computation. In the context of machine learning, DP ensures that algorithms can learn the relations between data and predictions while preventing them from memorizing sensitive information about any specific individual in the training data. While this property is appealing, it was recently observed that DP systems may induce biased and unfair outcomes for different groups of individuals [12, 137]. The resulting outcomes can have significant impacts on the life of the individuals with negative effects on financial, criminal, or job-hiring decisions [45]. *While these surprising observations have become apparent in several contexts, their causes are largely understudied.*

This chapter makes a step toward filling this important gap and investigates the unequal impacts that can occur when training a model using Private Aggregation of Teacher Ensembles (PATE), a state-of-the-art privacy-preserving ML framework [93]. PATE involves combining multiple agnostic models, referred to as *teachers*, to create a *student* model that is able to predict an output based on noisy voting among the teachers. This approach satisfies differential privacy and has been demonstrated to be effective for learning high-quality private models in semi-supervised settings. The chapter examines which algorithmic and data properties contribute to disproportionate impacts, why these aspects are affecting different groups of individuals disproportionately, and proposes a solution for mitigating these effects.

In summary, the chapter makes several key contributions: **(1)** It introduces a fairness measure that extends beyond accuracy parity and assesses the direct impact of privacy on model outputs for different groups. **(2)** It examines this fairness measure in the context of PATE, a leading privacy-focused ML framework. **(3)** It identifies key components of model parameters and data properties that contribute to disproportionate impacts on different groups during private training. **(4)** It investigates the circumstances under which these components disproportionately affect different groups. **(5)** Finally, based on these findings, the chapter proposes a method for reducing these unfair im-

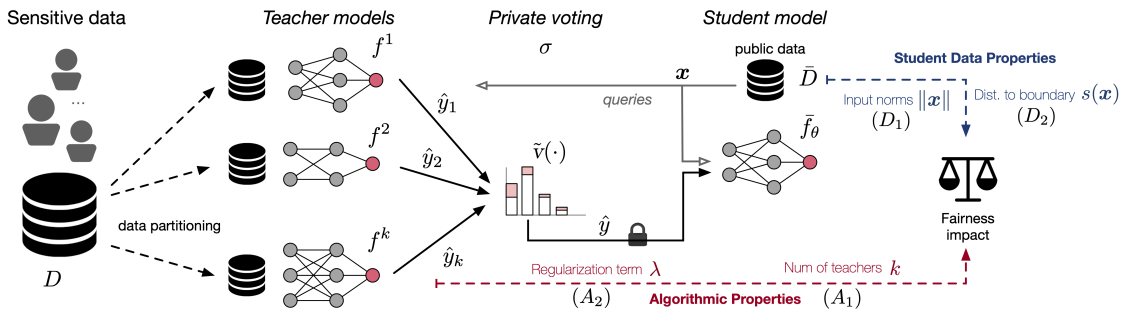


Fig. 4.1: Illustration of PATE and aspects contributing to fairness.

pacts while maintaining high accuracy.

The empirical advantages of privacy-preserving ensemble models over other frameworks, such as DP-SGD [3, 49, 130], make this work a significant and widely relevant contribution to understanding and addressing the disproportionate impacts observed in semi-supervised private learning systems. As far as we are aware, this is the first study to examine the causes of disparate impacts in privacy-preserving ensemble models.

4.2 Related work

The relationship between privacy and fairness has been a topic of recent debate, as recently surveyed by [46], with several researchers raising questions about the tradeoffs involved [41]. [34] specifically studied the tradeoffs between differential privacy and equal opportunity, a fairness criterion that requires a classifier to have equal true positive rates for different groups. They demonstrated that it is not possible to simultaneously achieve $(\epsilon, 0)$-differential privacy, satisfy equal opportunity, and have accuracy better than a constant classifier. Additionally, it has been proven that when training data has a long-tailed distribution, it is impossible to develop a private learning algorithm that has high accuracy for minority groups [109]. These findings have led to the question of whether fair models can be created while preserving sensitive information, and have spurred the development of various approaches such as those presented in [87, 128].

[71] were the first to show, empirically, that decision tasks made using DP datasets may disproportionately affect some groups of individuals over others. These studies were complemented

theoretically by [126]. Similar observations were also made in the context of model learning. [12] empirically observed that the accuracy of a DP model trained using DP-Stochastic Gradient Descent (DP-SGD) decreased disproportionately across groups causing larger negative impacts to the underrepresented groups. [43] and [130] reached similar conclusions and showed that this disparate impact was not limited to highly imbalanced data.

This chapter builds on this body of work and their important empirical observations. It provides an analysis of the causes of unfairness in the context of private learning ensembles, a significant privacy-enhancing ML system, and introduces guidelines for mitigating these effects.

4.3 Preliminaries: differential privacy

Differential privacy (DP) is a strong privacy notion stating that the probability of any output does not change much when a record is added or removed from a dataset, limiting the amount of information that the output reveals about any individual. The action of adding or removing a record from a dataset D, resulting in a new dataset D', defines the notion of *adjacency*, denoted $D \sim D'$.

Definition 4.1 ([37]). *A mechanism $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ with domain $\mathcal{D}$ and range $\mathcal{R}$ satisfies (ϵ, δ)-differential privacy, if, for any two adjacent inputs $D \sim D' \in \mathcal{D}$, and any subset of output responses $R \subseteq \mathcal{R}$:*

$$\Pr[\mathcal{M}(D) \in R] \leq e^{\epsilon} \Pr[\mathcal{M}(D') \in R] + \delta.$$

Parameter $\epsilon > 0$ describes the *privacy loss* of the algorithm, with values close to 0 denoting strong privacy, while parameter $\delta \in [0, 1)$ captures the probability of failure of the algorithm to satisfy ϵ-DP. The global sensitivity Δ_ℓ of a real-valued function $\ell : \mathcal{D} \to \mathbb{R}$ is defined as the maximum amount by which ℓ changes in two adjacent inputs: $\Delta_\ell = \max_{D \sim D'} \|\ell(D) - \ell(D')\|$. In particular, the Gaussian mechanism, defined by $\mathcal{M}(D) = \ell(D) + \mathcal{N}(0, \Delta_\ell^2\, \sigma^2)$, where $\mathcal{N}(0, \Delta_\ell^2\, \sigma^2)$ is the Gaussian distribution with 0 mean and standard deviation $\Delta_\ell^2\, \sigma^2$, satisfies (ϵ, δ)-DP for $\delta > \frac{4}{5} \exp(-(\sigma\epsilon)^2/2)$ and $\epsilon < 1$ [39].

4.4 Problem settings and goals

This chapter considers a *private* dataset D consisting of n individuals' data $(\boldsymbol{x}_i, y_i)$, with $i \in [n]$, drawn i.i.d. from an unknown distribution Π. Therein, $\boldsymbol{x}_i \in \mathcal{X}$ is a sensitive feature vector containing a protected group attribute $\boldsymbol{a}_i \in \mathcal{A} \subset \mathcal{X}$, and $y_i \in \mathcal{Y} = [C]$ is a C-class label. For example, consider a classifier that needs to predict criminal defendants' recidivism. The data features $\boldsymbol{x}_i$ may describe the individual's demographics, education, and crime committed, the protected attribute $\boldsymbol{a}_i$ may describe the individual's gender or ethnicity, and y_i whether the individual has high risk to reoffend.

This chapter studies the fairness implications arising when training private semi-supervised transfer learning models. The setting is depicted in Figure 4.1. We are given an ensemble of *teacher* models $\boldsymbol{T} = \{f^j\}_{j=1}^k$, with each $f^j : \mathcal{X} \to \mathcal{Y}$ trained on a non-overlapping portion D_i of D. This ensemble is used to transfer knowledge to a *student* model $\bar{f}_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$, where $\boldsymbol{\theta}$ is a vector of real-valued parameters.

The student model $\bar{f}$ is trained using a *public* dataset $\bar{D} = \{\boldsymbol{x}_i\}_{i=1}^m$ with samples drawn i.i.d. from the same distribution Π considered above but whose labels are unrevealed. We focus on learning classifier $\bar{f}_{\boldsymbol{\theta}}$ using knowledge transfer from the teacher model ensemble $\boldsymbol{T}$ while guaranteeing the privacy of each individual's data $(\boldsymbol{x}_i, y_i) \in D$. The sought model is learned by minimizing the regularized empirical risk function with loss $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$:

$$\boldsymbol{\theta}^* = \operatorname*{argmin}_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; \bar{D}, \boldsymbol{T}) + \lambda \|\boldsymbol{\theta}\|^2 \tag{4.1}$$

$$= \sum_{\boldsymbol{x} \in \bar{D}} \ell\left(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), v\left(\boldsymbol{T}(\boldsymbol{x})\right)\right) + \lambda \|\boldsymbol{\theta}\|^2, \tag{4.2}$$

where $v : \mathcal{Y}^k \to \mathcal{Y}$ is a *voting scheme* used to decide the prediction label from the ensemble $\boldsymbol{T}$, with $\boldsymbol{T}(\boldsymbol{x})$ used as a shorthand for $\{f^j(\boldsymbol{x})\}_{j=1}^k$, and $\lambda > 0$ is a regularization term.

We focus on DP classifiers that protect the disclosure of the individual's data and analyzes the fairness impact (as defined below) of privacy on different groups of individuals.

Privacy. *Privacy* is achieved by using a DP version $\tilde{v}$ of the voting function v:

$$\tilde{v}(\boldsymbol{T}(\boldsymbol{x})) = \text{argmax}_c\{\#_c(\boldsymbol{T}(\boldsymbol{x})) + \mathcal{N}(0, \sigma^2)\} \tag{4.3}$$

which perturbs the reported counts $\#_c(\boldsymbol{T}(\boldsymbol{x})) = |\{j : j \in [k], f^j(\boldsymbol{x}) = c\}|$ for class $c \in \mathcal{C}$ with zero-mean Gaussian and standard deviation σ. The overall approach, called *PATE* [97], guarantees (ϵ, δ)-DP, with privacy loss scaling with the magnitude of the standard deviation σ and the size of the public dataset $\bar{D}$. A detailed review of the privacy analysis of PATE is reported in Section 4.10.3. Throughout the chapter, the privacy-preserving parameters of the model $\bar{f}$ trained with noisy voting $\tilde{v}(\boldsymbol{T}(\boldsymbol{x}))$ are denoted with $\tilde{\boldsymbol{\theta}}$.

Fairness. One widely used metric for measuring utility in private learning is the *excess risk* [142], which is defined as the difference between the private and non-private risk functions:

$$R(S, \boldsymbol{T}) \stackrel{\text{def}}{=} \mathbb{E}_{\tilde{\boldsymbol{\theta}}}\left[\mathcal{L}(\tilde{\boldsymbol{\theta}}; S, \boldsymbol{T})\right] - \mathcal{L}(\boldsymbol{\theta}^*; S, \boldsymbol{T}), \tag{4.4}$$

where the expectation is taken over the randomness of the private mechanism, S is a subset of $\bar{D}$, $\tilde{\boldsymbol{\theta}}$ is the private student model's parameters, and $\boldsymbol{\theta}^* = \text{argmin}_{\boldsymbol{\theta}}\, \mathcal{L}(\boldsymbol{\theta}; \bar{D}, \boldsymbol{T}) + \lambda\|\boldsymbol{\theta}\|^2$.

In this chapter, the unfairness introduced by privacy in the learning task is measured using the difference in excess risks of each protected subgroup. This notion is significant because it captures the unintended impact of privacy on task accuracy for a given group, and it chapters to the concept of accuracy parity, a standard metric in fair and private learning. More specifically, the chapter focuses on measuring the excess risk $R(\bar{D}_{\leftarrow a}, \boldsymbol{T})$ for groups $a \in \mathcal{A}$, where $\bar{D}_{\leftarrow a}$ is the subset of $\bar{D}$ containing only samples from a group a. We use the shorthand $R(\bar{D}_{\leftarrow a})$ to refer to $R(\bar{D}_{\leftarrow a}, \boldsymbol{T})$ and assume that the private mechanisms are non-trivial, i.e., they minimize the population-level excess risk $R(\bar{D})$.

Definition 4.2. *Fairness is measured as the highest excess risk difference among all groups:*

$$\xi(\bar{D}) = \max_{a,a' \in \mathcal{A}} R(\bar{D}_{\leftarrow a}) - R(\bar{D}_{\leftarrow a'}). \tag{4.5}$$

Notice how this definition of fairness chapters to the concept of accuracy parity [12], which measures the disparity of task accuracy across groups, when the adopted loss ℓ is a 0/1-loss. All the experiments in the chapter use, in fact, this 0/1-loss, while the theoretical analysis considers general differentiable loss functions.

4.5 PATE fairness analysis: roadmap

The objective of this chapter is to identify the factors that cause unfairness in PATE and understand why they have this effect. The following sections isolate these key factors, which will be divided into two categories: *algorithm parameters* and *public student data characteristics*. The theoretical analysis assumes that, for a group $a \in \mathcal{A}$, the group loss function $\mathcal{L}(\boldsymbol{\theta}; D_{\leftarrow a}, \boldsymbol{T})$ is convex and β_a-smooth with respect to the model parameters $\boldsymbol{\theta}$ for some $\beta_a \geq 0$. However, the evaluation does not impose any restrictions on the form of the loss function. A detailed description of the experimental settings can be found in Section 4.10.4, and the proofs of all theorems are included in Section 4.10.2.

A fairness bound. We start by introducing a bound on the model disparity, which will be crucial for identifying the algorithm and data characteristics that contribute to unfairness in PATE. Throughout the chapter, we refer to the quantity $\Delta_{\tilde{\boldsymbol{\theta}}} \stackrel{\text{def}}{=} \|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|$ as to *model deviation due to privacy*, or simply *model deviation*, as it captures the effect of the private teachers' voting on the student learned model. Here, $\boldsymbol{\theta}^*$ and $\tilde{\boldsymbol{\theta}}$ represent the parameters of student model $\bar{f}$ learned using a clean or noisy voting scheme, respectively.

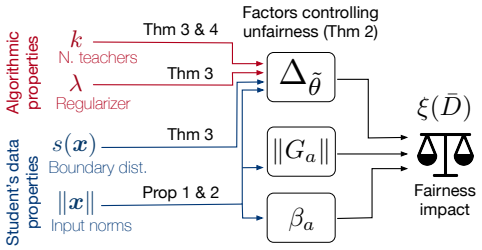


Fig. 4.2: Factors impacting PATE fairness.

Theorem 4.1. *The model fairness is upper bounded as:*

$$\xi(\bar{D}) \leq 2 \max_a \|G_a\| \, \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] + 1/2 \max_a \beta_a \, \mathbb{E}\left[\Delta^2_{\tilde{\boldsymbol{\theta}}}\right], \tag{4.6}$$

where $G_a = \mathbb{E}_{\boldsymbol{x} \sim \bar{D}_{\leftarrow a}}\left[\nabla_{\boldsymbol{\theta}^*} \ell(\bar{f}_{\boldsymbol{\theta}^*}(\boldsymbol{x}), y)\right]$ *is the gradient of the group loss evaluated at* $\boldsymbol{\theta}^*$*, and* $\Delta_{\tilde{\boldsymbol{\theta}}}$ *and* $\Delta^2_{\tilde{\boldsymbol{\theta}}}$ *capture the first and second order statistics of the model deviation.*

The above illustrates that the model unfairness is proportionally regulated by three direct factors: **(1)** the model deviation $\Delta_{\tilde{\boldsymbol{\theta}}}$, **(2)** the maximum gradient norm $\max_a \|G_a\|$ among all groups, and **(3)** the largest smoothness parameter $\max_a \beta_a$ among all groups.

The chapter delves into which **A**lgorithms' parameters and **D**ata characteristics affect the factors that contribute to model unfairness. Within the **A**lgorithm's parameters, in addition to the privacy variable ϵ (captured by the noise parameter σ), the chapter identifies two factors having a direct impact on fairness: ($\boldsymbol{A_1}$) the regularization term λ associated with the student risk function and ($\boldsymbol{A_2}$) the size k of the teachers' ensemble. Regarding the public student **D**ata's characteristics, the chapter shows that ($\boldsymbol{D_1}$) the magnitude of the sample input norms $\|\boldsymbol{x}\|$ and ($\boldsymbol{D_2}$) the distance of a sample to the decision boundary (denoted $s(\boldsymbol{x})$) are key factors that can exacerbate the excess risks induced by the student model. The relationships between these factors and how they impact model fairness are illustrated in Figure 4.2.

Several aspects of the analysis in this chapter rely on the following definition.

Definition 4.3. *Given a data sample* $(\boldsymbol{x}, y) \in D$*, for an ensemble* $\boldsymbol{T}$ *and voting scheme* v*, the*

flipping probability *is:*

$$p_{\boldsymbol{x}}^{\leftrightarrow} \stackrel{def}{=} \Pr\left[\tilde{\mathrm{v}}(\boldsymbol{T}(\boldsymbol{x})) \neq \mathrm{v}(\boldsymbol{T}(\boldsymbol{x}))\right].$$

It connects the *voting confidence* of the teacher ensemble with the perturbation induced by the private voting scheme and will be useful in the fairness analysis introduced below.

The theoretical results presented in the following sections are supported and corroborated by empirical evidence from tabular datasets (UCI Adults, Credit card, Bank, and Parkinsons) and an image dataset (UTKFace). These results were obtained using feed-forward networks with two hidden layers and nonlinear ReLU activations for both the ensemble and student models for tabular data, and CNNs for image data. All reported metrics are the average of 100 repetitions used to compute empirical expectations and report 0/1 losses, *which capture the concept of accuracy parity*. While the chapter provides a brief overview of the empirical results to support the theoretical claims, extended experiments and more detailed descriptions of the datasets can be found in Section 4.10.4.

4.6 Algorithm's parameters

This section analyzes the algorithm's parameters that affect the disparate impact of the student model outputs. The fairness analysis reported in this section assumes that the student model loss $\ell(\cdot)$ is convex and *decomposable*:

Definition 4.4. *A function $\ell(\cdot)$ is* decomposable *if there exists a parametric function $h_{\boldsymbol{\theta}}: \mathcal{X} \to \mathbb{R}$, a constant real number c, and a function $z: \mathbb{R} \to \mathbb{R}$, such that, for $\boldsymbol{x} \in \mathcal{X}$, and $y \in \mathcal{Y}$:*

$$\ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = z(h_{\boldsymbol{\theta}}(\boldsymbol{x})) + c\, y\, h_{\boldsymbol{\theta}}(\boldsymbol{x}). \tag{4.7}$$

A number of loss functions commonly adopted in ML, including the logistic loss (used in our experiments) or the least square loss function, are decomposable [99]. Additionally, while restrictions are commonly imposed on the loss functions to render the analysis tractable, our findings are

empirically validated on non-linear models.

It is important to recall that the model deviation is a central factor that proportionally controls the unfairness of PATE (Theorem 4.1). In the following, we provide a useful bound on the model deviation and highlight its relationship with key algorithm parameters.

Theorem 4.2. *Consider a student model $\bar{f}_{\boldsymbol{\theta}}$ trained with a convex and decomposable loss function $\ell(\cdot)$. Then, the first order statistics of the model deviation is upper bounded as:*

$$\mathbb{E}\Big[\Delta_{\tilde{\boldsymbol{\theta}}}\Big] \leq \frac{|c|}{m\lambda}\left[\sum_{\boldsymbol{x}\in\bar{D}} p_{\boldsymbol{x}}^{\leftrightarrow}\|G_{\boldsymbol{x}}^{\max}\|\right], \tag{4.8}$$

where c is a real constant and $G_{\boldsymbol{x}}^{\max} = \max_{\boldsymbol{\theta}} \|\nabla_{\boldsymbol{\theta}} h_{\boldsymbol{\theta}}(\boldsymbol{x})\|$ represents the maximum gradient norm distortion introduced by a sample $\boldsymbol{x}$. Both c and h are defined as in Equation 4.7.

The proof relies on λ-strong convexity of the loss function $\mathcal{L}(\cdot) + \lambda\|\boldsymbol{\theta}\|$ (see Section 4.10.2) and its tightness is demonstrated empirically in Section 4.10.6. Theorem 4.2 reveals how the student model changes due to privacy and chapters it with two mechanism-dependent components: **(1)** the regularization term λ of the empirical risk function $\mathcal{L}(\boldsymbol{\theta}, \bar{D}, \boldsymbol{T})$ (see (4.1)), and **(2)** the flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$, which, as it will be shown later, is heavily controlled by the size k of the teacher ensemble. These mechanisms-dependent components and the focus of this section, while data-dependent components, including those chapterd to the maximum gradient norm distortion $G_{\boldsymbol{x}}^{\max}$ are discussed to Section 4.7.

A_1: The impact of the regularization term λ. The first immediate observation of Theorem 4.2 is that variations of the regularization term λ can increase or decrease the difference between the private and non-private student model parameters. Since the model deviation $\mathbb{E}[\Delta_{\tilde{\boldsymbol{\theta}}}]$ has adirect relationship with the fairness goal (see the first term of RHS of (4.6) in Theorem 4.1) *the regularization term affects the disparate impact of the privacy-preserving student model*. These effects are further illustrated in Figure 4.3 (top). The figure shows how increasing λ reduces the expected difference between the privacy-preserving and original model parameters $\mathbb{E}[\Delta_{\tilde{\boldsymbol{\theta}}}]$ (left), as well as

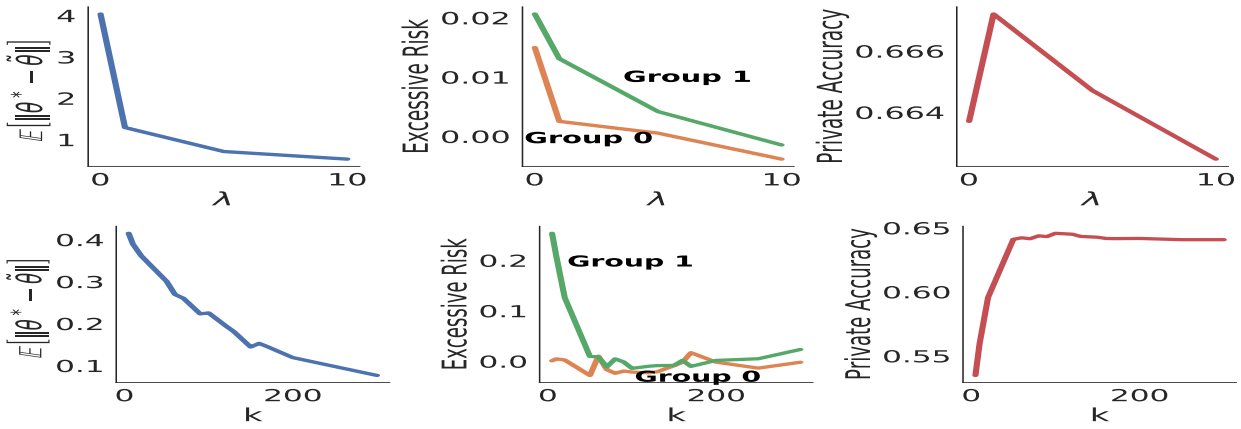


Fig. 4.3: Credit card dataset with $\sigma = 50, k = 150$ (top) and $\lambda = 100$ (bottom). Expected model deviation (left), excess risk (middle), and model accuracy (right) as a function of the regularization term (top) and ensemble size (bottom).

the excess risk $R(\bar{D}_{\leftarrow a})$ difference between groups $a = 0$ and $a = 1$ (middle). Note, however, that while larger λ values may reduce the model unfairness, they can hurt the model's accuracy, as shown in the right plot. The latter is an intuitive and recognized effect of large regularizers [82].

***A*$_2$: The impact of the teachers ensemble size k.** Next, we consider the relationship between the ensemble size k and the resulting private model's fairness. The following result chapters the size of the ensemble with its voting confidence.

Theorem 4.3. *For a sample $\boldsymbol{x} \in \bar{D}$ let the teacher models outputs $f^i(\boldsymbol{x})$ be in agreement, $\forall i \in [k]$. The flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ is given by $p_{\boldsymbol{x}}^{\leftrightarrow} = 1 - \Phi(\frac{k}{\sqrt{2}\sigma})$, where $\Phi(\cdot)$ is the CDF of the standard Normal distribution and σ is the standard deviation in the Gaussian mechanism.*

The proof is based on the properties of independent Gaussian random variables. This analysis shows that the ensemble size k (as well as the privacy parameter σ) directly affects the outcome of the teacher voting and, therefore, the model deviation and its disparate impact. The theorem shows that larger k values correspond to smaller flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$. *In conjunction with Theorem 4.1, this suggests that the model deviation due to privacy and the excess risks for various groups are inversely proportional to the ensemble size k.*

Figure 4.4 (top) illustrates the relationship between the number k of teachers and the flipping

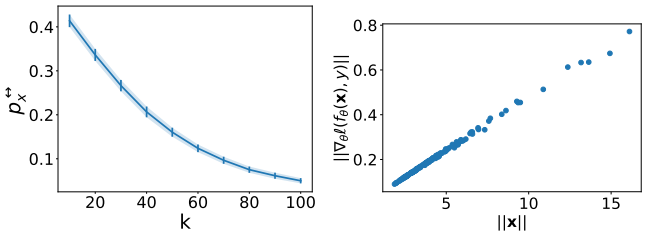


Fig. 4.4: Credit-card: Average flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ for samples $\boldsymbol{x} \in \bar{D}$ as a function of the ensemble size k (left) and the relation between gradient and input norms (right).

probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ of the ensemble. The plot shows a clear trend indicating that larger ensembles result in smaller flipping probabilities. It is worth noting that in these experiments, *different teachers may have different agreements on each sample*, thus this result generalizes the one presented in Theorem 4.3. Additionally, Figure 4.3 (bottom) shows that increasing k reduces the expected model deviation (left), reduces the group excess risk difference (middle), and increases the accuracy of the model $\bar{f}$ (right). Similar to theregularization term λ, large values k can decrease the accuracy of the (private and non-private) models. This behavior is chapterd to the bias-variance tradeoff imposed on the growing ensemble with less training data available to each teacher.

This section concludes with a useful corollary of Theorem 4.2.

Corollary 4.1 (Theorem 4.2)**.** *For a* logistic regression *classifier* $\bar{f}_{\boldsymbol{\theta}}$*, the model deviation is upper bounded as:*

$$\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] \leq \frac{1}{m\lambda}\left[\sum_{\boldsymbol{x}\in\bar{D}} p_{\boldsymbol{x}}^{\leftrightarrow}\|\boldsymbol{x}\|\right]. \tag{4.9}$$

This result highlights the presence of a relationship between gradient norms and input norms, which is further illustrated in Figure 4.4 (bottom). The plot shows a strong correlation between inputs and their associated gradient norms. The result also shows that samples with large norms can significantly impact fairness, emphasizing the importance of considering the characteristics of the student data, which are the subject of study in the next section.

In summary, the regularization parameter λ and the ensemble size k are two key algorithmic parameters that, by bounding the model deviation $\Delta_{\tilde{\theta}}$, can control the disparate impacts of the student model. These relations are further illustrated in the causal graph in Figure 4.1.

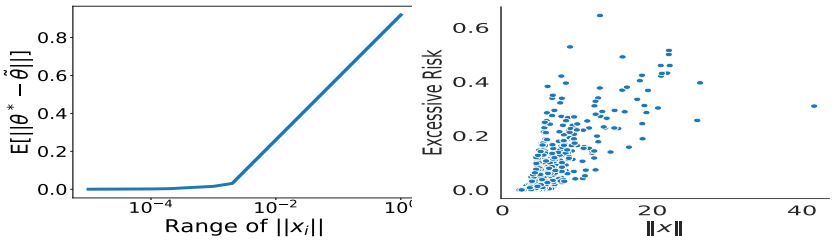


Fig. 4.5: *Credit*: Relation between input norms and model deviation (top) and Spearman correlation between input and excess risk (bottom).

4.7 Student's data properties

Having examined the algorithmic properties of PATE affecting fairness, this section turns on analyzing the role of certain characteristics of the student data in regulating the disproportionate impacts of of the algorithm. The results below will show that the norms of the student's data samples and their distance to the decision boundary can significantly impact the excess risk in PATE. This is particularly interesting as it dispels the notion that unfairness in these models is solely due to imbalanced training data. The following is a second corollary of Theorem 4.2 and bounds the second order statistics of the model deviation to privacy.

Corollary 4.2 (Theorem 4.2)**.** *Given the same settings and assumption of Theorem 4.2, it follows:*

$$\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}^2\right] \leq \frac{|c|^2}{m\lambda^2}\left[\sum_{\boldsymbol{x}\in\bar{D}} p_{\boldsymbol{x}}^{\leftrightarrow 2}\|G_{\boldsymbol{x}}^{\max}\|^2\right]. \tag{4.10}$$

Note that, similarly to what shown by Corollary 5.1, when $\bar{f}_{\boldsymbol{\theta}}$ is a logistic regression model, the gradient norm $\|G_{\boldsymbol{x}}^{\max}\|$ above can be substituted with the input norm $\|\boldsymbol{x}\|$.

The rest of the section focuses on logistic regression models, however, as our experimental results illustrate, the observations extend to complex nonlinear models as well.

(D_1): The impact of the data input norms. First notice that the norm $\|\boldsymbol{x}\|$ of a sample $\boldsymbol{x}$ strongly influences the model deviation controlling quantity $\Delta_{\tilde{\boldsymbol{\theta}}}$ as already observed by Corollaries 5.1 and 4.2. This aspect is further highlighted in Figure 4.5 (top), which illustrates that samples with high input norms have a significant impact on the model deviation. *As a result, these samples*

may contribute to the unfairness of the model, as per Theorem 4.1.

Next, recall that the group gradient norms G_a have a proportional effect on the upper bound of the model unfairness, as shown in Theorem 4.1. These norms also have an effect on the excess risk $R(\bar{D}_{\leftarrow a})$, as shown in Lemma 4.1, Section 4.10.2. The following results reveal a connection between the gradient norm for a sample $\boldsymbol{x} \in \bar{D}$ and its associated input norm, and how these factors chapter to the unfairness observed in the student model.

Proposition 4.1. *Consider a logistic regression binary classifier $\bar{f}_{\boldsymbol{\theta}}$ with cross entropy loss function ℓ. For a given sample $(\boldsymbol{x}, a, y) \in \bar{D}$, the gradient $\nabla_{\boldsymbol{\theta}^*} \ell(\bar{f}_{\boldsymbol{\theta}^*}(\boldsymbol{x}), y)$ is given by:*

$$\nabla_{\boldsymbol{\theta}^*} \ell(\bar{f}_{\boldsymbol{\theta}^*}(\boldsymbol{x}), y) = \big(\bar{f}_{\boldsymbol{\theta}^*}(\boldsymbol{x}) - y\big) \otimes \boldsymbol{x},$$

where $\otimes$ expresses the Kronecker product.

Thus, the relation above suggests that the *input norm* of data samples play a key role in controlling their associated excess risk, and, thus, that of the group in which they belong to. This aspect can be appreciated in Figure 4.5 (bottom), which shows a strong correlation between the input norms and excess risk. This observation is significant because it challenges the common belief that unfairness is solely caused by imbalances in group sizes. Instead, it suggests that the properties of the data itself directly contribute to unfairness.

Finally, it should be noted that the smoothness parameter β_a reflects the local flatness of the loss function in relation to samples from a group a. An extension of the results from [114] is provided to derive β_a for logistic regression classifiers, further illustrating the connection between the input norms $\|\boldsymbol{x}\|$ of a group $a \in \mathcal{A}$ and the smoothness parameters β_a.

Proposition 4.2. *Consider again a binary logistic regression as in Proposition 4.1. The smoothness parameter β_a for a group $a \in \mathcal{A}$ is given by: $\beta_a = 0.25 \max_{\boldsymbol{x} \in D_a} \|\boldsymbol{x}\|^2$.*

Therefore, Propositions 4.1 and 4.2 show that groups with large (small) inputs' norms tend to have large (small) gradient norms and smoothness parameters. Since these factors influence the

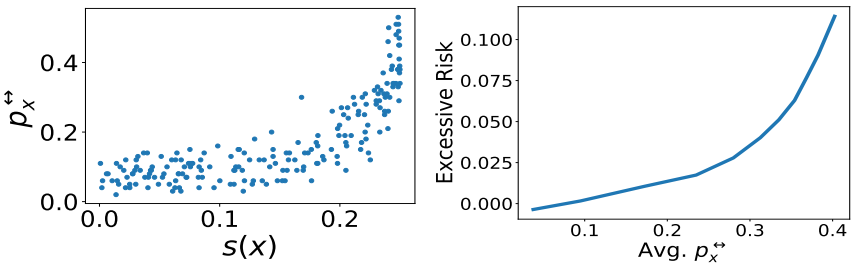


Fig. 4.6: *Credit*: Spearman correlation between closeness to boundary $s(\boldsymbol{x})$ and flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ (top) and relation between input norms and excess risk (bottom).

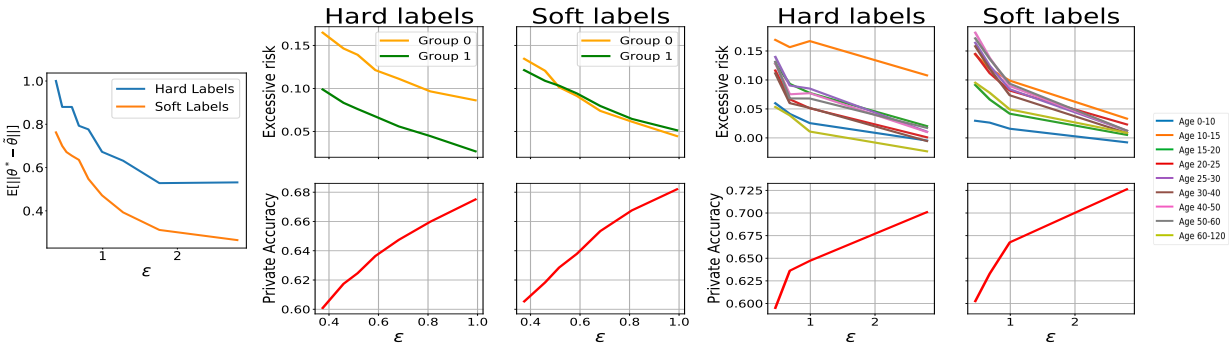


Fig. 4.7: Training privately PATE with hard and soft labels: Model deviation at varying of the privacy loss (left) on Credit dataset and excess risk at varying of the privacy loss for Credit (middle) and UTKFace (right) datasets.

model deviation, they also affect the associated excess risk, leading to larger disparate impacts. An extended analysis of the above claim is provided in Section 4.10.11.

(D_2): The impact of the distance to decision boundary. As mentioned in Theorem 4.2, the flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ of a sample $\boldsymbol{x} \in \bar{D}$ directly controls the model deviation $\Delta_{\tilde{\theta}}$. Intuitively, samples close to the decision boundary are associated to small ensemble voting confidence and vice-versa. Thus, groups with samples close to the decision boundary will be more sensitive to the noise induced by the private voting process. To illustrate this intuition the chapter reports the concept of *closeness to boundary*.

Definition 4.5 ([124]). *Let $f_{\boldsymbol{\theta}}$ be a C-classes classifier trained using data $\bar{D}$ with its true labels. The closeness to the decision boundary $s(\boldsymbol{x})$ is defined as: $s(\boldsymbol{x}) \stackrel{def}{=} 1 - \sum_{c=1}^{C} f_{\boldsymbol{\theta}^*,c}(\boldsymbol{x})^2$, where $f_{\boldsymbol{\theta},c}$ denotes the softmax probability for class c.*

The above discussion chapters large (small) values of $s(\boldsymbol{x})$ to projections of point $\boldsymbol{x}$ that are close (distant) to the model decision boundary. *The concept of closeness to decision boundary provides a way to indirectly quantify the flipping probability of a sample.* Empirically, the correlation between the distance of sample $\boldsymbol{x}$ to the decision boundary and its flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ is illustrated in Figure 4.6 (top). The plots are once again generated using a neural network with nonlinear objective and the relation holds for all datasets analyzed. The plot indicates that the samples that are close to the decision boundary have a higher probability of "flipping" their label, leading to a worse excess risk and unfairness. Finally, Figure 4.6 (bottom) further illustrates the strong proportional effect of the flipping probability on the excess risk.

To summarize, the norms $\|\boldsymbol{x}\|$ of a group's samples and their associated distance to boundary $s(\boldsymbol{x})$ are two key characteristics of the student data that influence fairness through their control of the model deviation $\Delta_{\tilde{\theta}}$, the smoothness parameters β_a, and the group gradients G_a, (see Figure 4.2 for a schematic representation).

4.8 Mitigation solution

The previous sections have identified a number of algorithmic and data-chapterd factors that can influence the disparate impact of the student model. These factors often affect the model deviation $\Delta_{\tilde{\boldsymbol{\theta}}}$, which is chapterd to the excess risk of different groups (as shown in Theorem 4.1), whose difference we would like to minimize. With this in mind, this section proposes a strategy to reduce the deviation of the private model parameters. To do so, we exploit the idea of *soft labels* instead of traditional *hard labels* in the voting process. Hard labels may be significantly affected by small perturbations due to noise, especially when the teachers have low confidence in their votes. For example, consider a binary classifier where for a sample $\boldsymbol{x}$, $k/2+1$ teachers vote label 0 and $k/2-1$, label 1, for some even ensemble size k. If perturbations are introduced to these counts to esnure privacy, the process may incorrectly report label ($\hat{y}=1$) with high probability, causing causing the student model's private parameters to deviate significantly from the non-private ones. This issue

can be partially addressed by the introduction of soft labels:

Definition 4.6 (Soft label)**.** *The* soft label *of a sample* $\boldsymbol{x}$ *is:* $\alpha(\boldsymbol{x}) = \left(\frac{\#_c(\boldsymbol{T}(\boldsymbol{x}))}{k}\right)_{c=1}^{C}$ *and their privacy-preserving counterparts* $\tilde{\alpha}(\boldsymbol{x})$ *adds Gaussian noise* $\mathcal{N}(0, \sigma^2)$ *in the numerator of* $\alpha(\boldsymbol{x})$*.*

To exploit soft labels, the training step of the student model uses loss $\ell'(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), \tilde{\alpha}) = \sum_{c=1}^{C} \tilde{\alpha}_c \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), c)$, which can be considered as a weighted version of the original loss function $\ell(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), c)$ on class label c, whose weight is its confidence $\tilde{\alpha}_c$. Note that $\ell'(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), \tilde{\alpha}) = \ell(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}))$ when all teachers in the ensemble chose the same label. The privacy loss for this model is equivalent to that of classical PATE. The analysis is reported in Section 4.10.3.

The effectiveness of this scheme is demonstrated in Figure 4.7. The experiment settings are reported in detail in the Section and reflect those described in Section 4.5. The left subplot shows the relation between the model deviation $\mathbb{E}\left[\Delta_{\bar{\boldsymbol{\theta}}}\right]$ at varying of the privacy loss ϵ (dictated by the noise level σ). Notice how the student models trained using soft labels reduce their model deviation ($E[\Delta_{\bar{\boldsymbol{\theta}}}]$) when compared to the counterparts that use hard labels.

The middle and right plots of Figure 4.7 show the impact of the proposed solution on the private student model in terms of the utility/fairness tradeoff. The top subplots illustrate the group excess risks $R(\bar{D}_{\leftarrow a})$ associated with each group $a \in \mathcal{A}$ for Credit (left) and UTKFace datasets (right), respectively. The bottom subplots illustrate the accuracy of the models, which include a simple ReLU network for the tabular dataset and a more complex CNN for the image dataset. Recall that the fairness goal $\xi(\bar{D})$ is captured by the gap between excess risk curves in the figures. Notice how soft labels can reduce the disparate impacts in private training (top). Notice also that while fairness is improved there is seemingly no cost in accuracy. On the contrary, using soft labels produces comparable or better models than the counterparts produced with hard labels.

Additional experiments, including illustrating the behavior of the mitigating solution at varying of the number k of teachers are reported in Section 4.10.4 and the trends are all consistent with what is described above. It is important to note that the proposed solution preserves the original privacy budget. In contrast, mitigating solutions that would consider explicitly the number of teachers K or the smoothness parameter λ will inevitably introduce further privacy/fairness tradeoffs as would

require costly privacy-preserving hyper-parameter optimization [98].

Finally, an important benefit of this solution is that it *does not* uses the protected group information ($a \in \mathcal{A}$) during training. Thus, it is applicable in challenging situations when it is not feasible to collect or use protected features (e.g., under the General Data Protection Regulation (GDPR) [72]). *These results are significant. They suggest that this mitigating solution can be effective for improving the disparate impact of private model ensembles without sacrificing accuracy.*

4.9 Discussion, limitations, and conclusions

This study highlights two key messages. First, the proposed mitigating solution chapters to concepts in robust machine learning. In particular, [96] showed that training a classifier with soft labels can increase its robustness against adversarial samples. This connection is not coincidental, as the deviation of the model is influenced by the voting outcomes of the teacher ensemble (as demonstrated in Theorems 4.1 and 4.2). In the same way that robust ML models are insensitive to input perturbations, an ensemble that strongly agrees will be less sensitive to noise and vice versa. This raises the question of the relationship between robustness and fairness in private models, which is an important open question. Second, we also note that more advanced voting schemes, such as interactive GNMAX [97], may produce different fairness results. While this is an interesting area for further analysis, these sophisticated voting schemes may introduce sampling bias (e.g., interactive GNMAX may exclude samples with low ensemble voting agreement), which could create its own fairness issues.

Given the growing use of privacy-preserving data-driven algorithms in consequential decisions, this work represents a significant and widely applicable step towards understanding the causes of disparate impacts in differentially private learning systems.

4.10 Appendix

4.10.1 Related work

The study of the disparate impacts caused by privacy-preserving algorithms has recently seen several important developments. [41] raise questions about the tradeoffs involved between privacy and fairness. [34] study the tradeoffs arising between differential privacy and equal opportunity, a fairness notion requiring a classifier to produce equal true positive rates across different groups. They show that there exists no classifier that simultaneously achieves $(\epsilon, 0)$-DP, satisfies equal opportunity, and has accuracy better than a constant classifier. This development has risen the question of whether one can practically build fair models while retaining sensitive information private, which culminated in a variety of proposals, including [62, 87, 128].

[71] were the first to show, empirically, that decision tasks made using DP datasets may disproportionately affect some groups of individuals over others. These studies were complemented theoretically by [126]. Similar observations were also made in the context of model learning. [12] empirically observed that the accuracy of a DP model trained using DP-Stochastic Gradient Descent (DP-SGD) decreased disproportionately across groups causing larger negative impacts to the underrepresented groups. [43, 130] reached similar conclusions and showed that this disparate impact was not limited to highly imbalanced data.

This chapter builds on this body of work and their important empirical observations. It provides an analysis for the reasons of unfairness in the context of semi-supervised private learning ensembles, an important privacy-enhancing ML system, as well as introduces mitigating guidelines.

4.10.2 Missing proofs

This section contains the missing proofs associated with the theorems and corollaries presented in the main chapter. The theorems are restated for completeness.

First we provide the upper bound on the excess risk per group $a \in \mathcal{A}$ in the following Lemma 4.1. This helps to understand what factors control the excess risk for a particular group.

Lemma 4.1. *The excess risk $R(\bar{D}_{\leftarrow a})$ of a group $a \in \mathcal{A}$ is upper bounded as:*

$$R(D_{\leftarrow a}) \leq \|G_a\| \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] + 1/2 \; \beta_a \mathbb{E}\left[\Delta^2_{\tilde{\boldsymbol{\theta}}}\right], \tag{4.11}$$

where $G_a = \mathbb{E}_{\boldsymbol{x} \sim \bar{D}_{\leftarrow a}}\left[\nabla_{\boldsymbol{\theta}^*} \ell(\bar{f}_{\boldsymbol{\theta}^*}(\boldsymbol{x}), y)\right]$ is the gradient of the group loss evaluated at $\boldsymbol{\theta}^*$, and $\Delta_{\tilde{\boldsymbol{\theta}}}$ and $\Delta^2_{\tilde{\boldsymbol{\theta}}}$ capture the first and second order statistics of the model deviation.

Proof. By β_a smoothness assumption on the loss function defined over a group $a \in \mathcal{A}$ it follows that:

$$\mathcal{L}(\tilde{\boldsymbol{\theta}}; D_{\leftarrow a}, \boldsymbol{T}) \leq \mathcal{L}(\boldsymbol{\theta}^*; D_{\leftarrow a}, \boldsymbol{T}) + \left(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right)^T G_a + \frac{\beta_a}{2} \|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|^2. \tag{4.12}$$

By taking the expectation on both sides of the above equation w.r.t. the randomness of the noise, we obtain:

$$\begin{aligned} \mathbb{E}[\mathcal{L}(\tilde{\boldsymbol{\theta}}; D_{\leftarrow a}, \boldsymbol{T}] &\leq \mathcal{L}(\boldsymbol{\theta}^*; D_{\leftarrow a}, \boldsymbol{T}) + G_a^T \mathbb{E}[(\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}})] + \frac{\beta_a}{2} \mathbb{E}[\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|^2] & (4.13) \\ &\leq \mathcal{L}(\boldsymbol{\theta}^*; D_{\leftarrow a}, \boldsymbol{T}) + \|G_a\| \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] + \frac{1}{2} \beta_a \mathbb{E}[\Delta^2_{\tilde{\boldsymbol{\theta}}}], & (4.14) \end{aligned}$$

where the last inequality is by Cauchy-Schwarz inequality on vectors. Next, by substituting $R(\bar{D}_{\leftarrow a}) = \mathbb{E}[\mathcal{L}(\tilde{\boldsymbol{\theta}}; D_{\leftarrow a}, \boldsymbol{T}] - \mathcal{L}(\boldsymbol{\theta}^*; D_{\leftarrow a}, \boldsymbol{T})$ into eq:final$_i neq we obtain the Lemma statement.$

□

Theorem 4.1. *The model fairness is upper bounded as:*

$$\xi(\bar{D}) \leq \max_a 2\|G_a\| \mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] + \max_a 1/2 \; \beta_a \mathbb{E}\left[\Delta^2_{\tilde{\boldsymbol{\theta}}}\right]. \tag{4.15}$$

Proof. By convexity assumption on the loss function defined over a group $a \in \mathcal{A}$ it follows that:

$$\mathcal{L}(\boldsymbol{\theta}^*; D_{\leftarrow a}, \boldsymbol{T}) + \left(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right)^T \boldsymbol{G}_a \leq \mathcal{L}(\tilde{\boldsymbol{\theta}}; D_{\leftarrow a}, \boldsymbol{T}) \tag{4.16}$$

By taking the expectation on both sides of the above equation w.r.t. the randomness of the noise, we obtain:

$$\mathbb{E}[\mathcal{L}(\tilde{\boldsymbol{\theta}}; D_{\leftarrow a}, \boldsymbol{T})] \geq \mathcal{L}(\boldsymbol{\theta}^*; D_{\leftarrow a}, \boldsymbol{T}) + \mathbf{E}[\left(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right)^T]\boldsymbol{G}_a \tag{4.17}$$

By combining Equation 4.17 and Equation 4.13 we obtain the following:

$$\mathbf{E}[\left(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right)^T]\boldsymbol{G}_a \leq R(\bar{D}_{\leftarrow a}) \leq \mathbb{E}\left[\left(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right)^T\right]\boldsymbol{G}_a + \frac{\beta_a}{2}\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}^2\right] \tag{4.18}$$

Based on the definition of fairness in Equation 4.5, it follows that:

$$\xi(\bar{D}) = \max_{a,a' \in \mathcal{A}} R(\bar{D}_{\leftarrow a}) - R(\bar{D}_{\leftarrow a'}) \leq \max_{a,a' \in \mathcal{A}} \mathbf{E}[\left(\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\right)^T]\left(G_a - G_{a'}\right) + \max_{a \in \mathcal{A}} \frac{\beta_a}{2}\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}^2\right] \tag{4.19}$$

$$\leq \max_a 2\|G_a\|\mathbf{E}[\|\tilde{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|] + \max_a \frac{\beta_a}{2}\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}^2\right] = 2\max_a \|G_a\|\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] + \max_a \frac{\beta_a}{2}\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}^2\right] \tag{4.20}$$

□

Theorem 4.2. *Consider a student model $\bar{f}_{\boldsymbol{\theta}}$ trained with a convex and decomposable loss function $\ell(\cdot)$. Then, the expected difference between the private and non-private model parameters is upper bounded as follows:*

$$\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] \leq \frac{|c|}{m\lambda}\left[\sum_{\boldsymbol{x} \in \bar{D}} p_{\boldsymbol{x}}^{\leftrightarrow} \|G_{\boldsymbol{x}}^{\max}\|\right], \tag{4.21}$$

where c is a real constant and $G_{\boldsymbol{x}}^{\max} = \max_{\boldsymbol{\theta}} \|\nabla_{\boldsymbol{\theta}} h_{\boldsymbol{\theta}}(\boldsymbol{x})\|$ represents the maximum gradient norm distortion introduced by a sample $\boldsymbol{x}$. Both c and h are defined as in Equation (4.7)*.*

Proof of Theorem 4.2 requires the following Lemma 4.2 from [113] on the property of strongly convex functions.

Lemma 4.2 ([113]). *Let $\mathcal{L}(\boldsymbol{\theta})$ be a differentiable function. Then $\mathcal{L}(\boldsymbol{\theta})$ is λ-strongly convex* iff *for all vectors $\boldsymbol{\theta}, \boldsymbol{\theta}'$:*

$$(\nabla_{\boldsymbol{\theta}}\mathcal{L} - \nabla_{\boldsymbol{\theta}'}\mathcal{L})^T(\boldsymbol{\theta} - \boldsymbol{\theta}') \geq \lambda\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|^2. \tag{4.22}$$

Proof of Theorem 4.2. Let us denote with $\hat{y}_i = v(\boldsymbol{T}(\boldsymbol{x}_i))$ to indicate the non-private voting label associated with $\boldsymbol{x}_i$ and $\tilde{y}_i = \tilde{v}(\boldsymbol{T}(\boldsymbol{x}_i))$ for the private voting label counterpart. The regularized empirical risk function with the non-private voting labels from 4.1 can be rewritten as follows:

$$\mathcal{L} = \frac{1}{m}\sum_{i=1}^{m} \ell(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_i), \hat{y}_i) + \lambda\|\boldsymbol{\theta}\| \tag{4.23}$$

$$= \frac{1}{m}\sum_{i=1}^{m} \big[z(h_{\boldsymbol{\theta}}(\boldsymbol{x}_i)) + c\hat{y}_i h_{\boldsymbol{\theta}}(\boldsymbol{x}_i)\big] + \lambda\|\boldsymbol{\theta}\|^2, \tag{4.24}$$

where the second equality is due to the decomposable loss assumption. Likewise, define $\tilde{\mathcal{L}}$ to be the regularized empirical risk function with private voting labels $\tilde{y}_i$:

$$\tilde{\mathcal{L}} = \frac{1}{m}\sum_{i=1}^{m} \big[z(h_{\boldsymbol{\theta}}(\boldsymbol{x}_i)) + c\tilde{y}_i h_{\boldsymbol{\theta}}(\boldsymbol{x}_i)\big] + \lambda\|\boldsymbol{\theta}\|^2, \tag{4.25}$$

Based on Equation 4.24 and Equation 4.25, it follows that: $\tilde{\mathcal{L}} = \mathcal{L} + \Delta_{\mathcal{L}}$ where $\Delta_{\mathcal{L}} = \frac{c}{m}\sum_{i=1}^{m}(\tilde{y}_i - \hat{y}_i)h_{\boldsymbol{\theta}}(\boldsymbol{x}_i)$.

Furthermore, since each individual loss function $\ell(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_i), \tilde{y}_i)$ or $\ell(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_i), \hat{y}_i)$ is convex for all i from the given assumption, then $\tilde{\mathcal{L}}$ and $\mathcal{L}$ both are λ-strongly convex.

Next, from the definition of $\tilde{\boldsymbol{\theta}} = \operatorname{argmin}_{\boldsymbol{\theta}} \tilde{\mathcal{L}}$, and $\overset{\star}{\boldsymbol{\theta}} = \operatorname{argmin}_{\boldsymbol{\theta}} \mathcal{L}$ it follows that:

$$\nabla_{\tilde{\boldsymbol{\theta}}}\tilde{\mathcal{L}} = \mathbf{0} \text{ and } \nabla_{\overset{\star}{\boldsymbol{\theta}}}\mathcal{L} = \mathbf{0}. \tag{4.26}$$

By Lemma 4.2, it follows that:

$$\big(\nabla_{\tilde{\boldsymbol{\theta}}}\tilde{\mathcal{L}} - \nabla_{\overset{\star}{\boldsymbol{\theta}}}\tilde{\mathcal{L}}\big)^T(\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}) \geq \lambda\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|^2. \tag{4.27}$$

Now since $\nabla_{\tilde{\boldsymbol{\theta}}}\tilde{\mathcal{L}} = \mathbf{0}$ by eq:4, we can rewrite Equation 4.27 as

$$\big(- \nabla_{\overset{\star}{\boldsymbol{\theta}}}\tilde{\mathcal{L}}\big)^T(\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}) \geq \lambda\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|^2, \tag{4.28}$$

since $\nabla_{\overset{\star}{\boldsymbol{\theta}}}\tilde{\mathcal{L}} = \nabla_{\overset{\star}{\boldsymbol{\theta}}}\mathcal{L} + \nabla_{\overset{\star}{\boldsymbol{\theta}}}\Delta_{\mathcal{L}} = \mathbf{0} + \nabla_{\overset{\star}{\boldsymbol{\theta}}}\Delta_{\mathcal{L}} = \nabla_{\overset{\star}{\boldsymbol{\theta}}}\Delta_{\mathcal{L}}$. In addition, by applying the Cauchy-Schwartz inequality to the L.H.S of eq:5b we obtain

$$\|\nabla_{\overset{\star}{\boldsymbol{\theta}}}\Delta_{\mathcal{L}}\|\|(\overset{\star}{\boldsymbol{\theta}} - \tilde{\boldsymbol{\theta}})\| \geq -(\nabla_{\overset{\star}{\boldsymbol{\theta}}}\Delta_{\mathcal{L}})^T(\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}) \geq \lambda\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|^2, \tag{4.29}$$

and thus,

$$\|\nabla_{\overset{\star}{\boldsymbol{\theta}}}\Delta_{\mathcal{L}}\| \geq \lambda\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|. \tag{4.30}$$

By definition of $\nabla_{\overset{\star}{\boldsymbol{\theta}}}\Delta_{\mathcal{L}}$ we can rewrite the above inequality as follows:

$$\|\nabla_{\overset{\star}{\boldsymbol{\theta}}}\Delta_{\mathcal{L}}\| = \|\frac{c}{m}\sum_{i=1}^{m}(\tilde{y}_i - \hat{y}_i)\nabla_{\overset{\star}{\boldsymbol{\theta}}}h_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}_i)\| \geq \lambda\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|^2. \tag{4.31}$$

Next, let $\rho_i = \hat{y}_i - \tilde{y}_i$, applying this substitution to the above and by triangle inequality it follows that:

$$\frac{|c|}{m}\sum_{i=1}^{m}|\rho_i|\|g_i\| \geq \frac{|c|}{m}\sum_{i=1}^{m}|\rho_i|\|\nabla_{\tilde{\boldsymbol{\theta}}}h_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{x}_i)\| \tag{4.32}$$

$$\geq \|\frac{c}{m}\sum_{i=1}^{m}\rho_i\nabla_{\tilde{\boldsymbol{\theta}}}h_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{x}_i)\| \geq \lambda\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|, \tag{4.33}$$

where the first inequality is due to definition of $g_{\boldsymbol{x}_i} = \max_{\boldsymbol{\theta}}\|\nabla_{\boldsymbol{\theta}}h_{\boldsymbol{\theta}}(\boldsymbol{x}_i)\|$ and the second inequality is due to the general triangle inequality . Since $|\rho_i|$ is a Bernoulli random variable, in which $|\rho_i| = 1$ w.p. $p^{\leftrightarrow}_{\boldsymbol{x}_i}$ and $|\rho_i| = 0$ w.p of $1 - p^{\leftrightarrow}_{\boldsymbol{x}_i}$. Therefore $\mathbb{E}[|\rho_i|] = p^{\leftrightarrow}_{\boldsymbol{x}_i}$. Thus, it follows that:

$$\mathbb{E}\big[\frac{|c|}{m}\sum_{i=1}^{m}|\rho_i|\|g_{\boldsymbol{x}_i}\|\big] = \frac{|c|}{m}\sum_{i=1}^{m}p^{\leftrightarrow}_{\boldsymbol{x}_i}\|g_{\boldsymbol{x}_i}\| \geq \lambda\mathbb{E}\big[\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|\big] = \mathbb{E}\big[\Delta_{\tilde{\boldsymbol{\theta}}}\big], \tag{4.34}$$

which concludes the proof. □

Theorem 4.3. *For a sample $\boldsymbol{x} \in \bar{D}$ let the teacher models outputs $f^i(\boldsymbol{x})$ be in agreement, $\forall i \in [k]$. The flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ is given by $p_{\boldsymbol{x}}^{\leftrightarrow} = 1 - \Phi(\frac{k}{\sqrt{2}\sigma})$, where $\Phi(\cdot)$ is the CDF of the std. Normal distribution and σ is the standard deviation in the Gaussian mechanism.*

For simplicity of exposition Theorem 4.3 considers binary classifiers, i.e., $\mathcal{Y} = \{0, 1\}$. The argument, however, can be trivially extended to generic C-classifiers.

Proof. By assumption, for any given sample $\boldsymbol{x}$, all teachers agree in their predictions, so w.l.o.g., assume k teachers output label 0, while none of them outputs label 1. Next, let $\psi, \psi' \sim \mathcal{N}(0, \sigma^2)$ be two independent Gaussian random variables which are added to true voting counts, k and 0, respectively. The associated flipping probability is:

$$p_{\boldsymbol{x}}^{\leftrightarrow} = \Pr\left[\tilde{\mathrm{v}}(\boldsymbol{T}(\boldsymbol{x})) \neq \mathrm{v}(\boldsymbol{T}(\boldsymbol{x}))\right] = \Pr(k + \psi \leq 0 + \psi') = \Pr(\psi' - \psi \geq k) \tag{4.35}$$

$$= 1 - \Pr(\psi - \psi' \leq k), \tag{4.36}$$

since ψ, ψ' are two independent Gaussian random variable with zero mean and standard deviation of σ. Therefore, $\psi' - \psi \sim \mathcal{N}(0, 2\sigma^2)$. Thus:

$$\Pr(\psi - \psi' \leq k) = \Pr(\mathcal{N}(0, 2\sigma^2) \leq k) = \Phi(\frac{k}{\sqrt{2}\sigma}).$$

Hence, the flipping probability will be: $p_{\boldsymbol{x}}^{\leftrightarrow} = 1 - \Phi(\frac{k}{\sqrt{2}\sigma})$. □

Corollary 4.1 (Theorem 4.2)**.** *Let $\bar{f}_{\boldsymbol{\theta}}$ be a* logistic regression *classifier. Its expected model deviation is upper bounded as:*

$$\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] \leq \frac{1}{m\lambda}\left[\sum_{\boldsymbol{x} \in \bar{D}} p_{\boldsymbol{x}}^{\leftrightarrow} \|\boldsymbol{x}\|\right]. \tag{4.37}$$

Proof. The loss function $\ell(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), y)$ of a logistic regression classifier with binary cross entropy

loss can be rewritten as follows:

$$\ell(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = -y \log(\frac{1}{1+\exp(-\boldsymbol{\theta}^T \boldsymbol{x})}) - (1-y)\log(\frac{\exp(-\boldsymbol{\theta}^T \boldsymbol{x})}{1+\exp(-\boldsymbol{\theta}^T \boldsymbol{x})}) \tag{4.38}$$

$$= y \log(\exp(-\boldsymbol{\theta}^T \boldsymbol{x})) - \log\big(\frac{\exp(-\boldsymbol{\theta}^T \boldsymbol{x})}{1+\exp(-\boldsymbol{\theta}^T \boldsymbol{x})}\big) \tag{4.39}$$

$$= y(-\boldsymbol{\theta}^T \boldsymbol{x}) - \log\big(\frac{\exp(-\boldsymbol{\theta}^T \boldsymbol{x})}{1+\exp(-\boldsymbol{\theta}^T \boldsymbol{x}}\big). \tag{4.40}$$

Hence, $\ell(\cdot)$ is decomposable by Definition 4.4 with $h_{\boldsymbol{\theta}}(\boldsymbol{x}) = -\boldsymbol{\theta}^T \boldsymbol{x}$, $c = 1$ and $z(h) = -\log(\frac{\exp(h)}{1+\exp(h)})$.

Applying Theorem 4.2 with $G_{\boldsymbol{x}}^{\max} = \max_{\boldsymbol{\theta}} \|\nabla_{\boldsymbol{\theta}} h_{\boldsymbol{\theta}}(\boldsymbol{x})\| = \max_{\boldsymbol{\theta}} \|\nabla_{\boldsymbol{\theta}} - \boldsymbol{\theta}^T \boldsymbol{x}\| = \|\boldsymbol{x}\|$, and $c = 1$, gives the intended result.

□

Corollary 4.2 (Theorem 4.2)**.** *Given the same settings and assumption of Theorem 4.2, it follows:*

$$\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}^2\right] \leq \frac{|c|^2}{m\lambda^2}\left[\sum_{\boldsymbol{x}\in\bar{D}} p_{\boldsymbol{x}}^{\leftrightarrow 2} \|G_{\boldsymbol{x}}^{\max}\|^2\right]. \tag{4.41}$$

Proof. First, by Theorem 4.2 we obtain an upper bound for $\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}^2\right]$ as follows:

$$\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}^2\right] \leq \frac{c^2}{\lambda^2}\left[\frac{1}{m}\sum_{\boldsymbol{x}\in\bar{D}} p_{\boldsymbol{x}}^{\leftrightarrow} \|G_{\boldsymbol{x}}^{\max}\|\right]^2. \tag{4.42}$$

Applying the sum of squares inequality on the R.H.S. of eq:second we obtain:

$$\frac{c^2}{\lambda^2}\left[\frac{1}{m}\sum_{\boldsymbol{x}\in\bar{D}} p_{\boldsymbol{x}}^{\leftrightarrow} \|G_{\boldsymbol{x}}^{\max}\|\right]^2 \leq \frac{c^2}{\lambda^2}\left[\frac{1}{m} p_{\boldsymbol{x}}^{\leftrightarrow 2} \|G_{\boldsymbol{x}}^{\max}\|^2\right], \tag{4.43}$$

which concludes the proof.

□

4.10.3 Privacy analysis

This section provides the privacy analysis for the original PATE model and the proposed mitigation solution. In PATE with the noisy-max scheme presented in (4.3) of the main chapter (also called GNMAX), the privacy budget is used for releasing the voting labels $\tilde{v}(\boldsymbol{T}(\boldsymbol{x}_i))$ (a.k.a. hard labels) for each of the m public data samples $\boldsymbol{x}_i \in \bar{D}$ according to:

$$\tilde{v}(\boldsymbol{T}(\boldsymbol{x}_i)) = \underset{c}{\operatorname{argmax}}\{\#_c(\boldsymbol{T}(\boldsymbol{x}_i)) + \mathcal{N}(0, \sigma^2)\} \tag{4.44}$$

The proposed mitigation solution, instead, releases privately the voting counts $(\#_c(\boldsymbol{T}(\boldsymbol{x}_i)) + \mathcal{N}(0, \sigma^2))_{c=1}^{C}$ and uses these noisy counts to construct the *soft-labels*, see Equation (11).

Using an analogous analysis as that provided in [97], adding or removing one individual sample $\boldsymbol{x}$ from any disjoint partition D_i of D can change the voting count vector by at most two. This value of the query deviation is obtained by GNMAX [97]. Therefore the privacy cost for releasing hard labels or soft-labels is equivalent.

Next, this section provides the privacy computation ϵ given by Gaussian mechanism which adds Gaussian noise with standard deviation σ to the voting counts.

The privacy analysis of PATE with hard or soft-labels is based on the concept of Renyi differential privacy (RDP) [85]. In either implementations, the process uses the Gaussian mechanism to add independent Gaussian noise to the voting counts. The following Proposition 7.4 from [97] derives the privacy guarantee for GNMAX.

Proposition 4.1. *The GNMAX aggregator with private Gaussian noise $\mathcal{N}(0, \sigma^2)$ satisfies $(\gamma, \gamma/\sigma^2)$-RDP for all $\gamma \geq 1$.*

Since the GNMAX mechanism is applied on m public data samples from $\bar{D}$, the total privacy loss spent to provide the private labels is derived by the following composition theorem.

Theorem 4.4 (Composition for RDP)**.** *If a mechanism $\mathcal{M}$ consists of a sequence of adaptive mechanisms $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m$ such that for any $i \in [m]$, $\mathcal{M}_i$ guarantees (γ, ϵ_i)-RDP, then $\mathcal{M}$ guarantees $(\gamma, \sum_{i=1}^{m} \epsilon_i)$-RDP.*

Based on Theorem 4.4 and Proposition 7.4, PATE satisfies $(\gamma, {}^{m\gamma}/_{\sigma^2})$-RDP. PATE also satisfies (ϵ, δ)-DP by the following theorem.

Theorem 4.5 (From RDP to DP)**.** *If a mechanism $\mathcal{M}$ guarantees (γ, ϵ)-RDP, then $\mathcal{M}$ guarantees $(\epsilon + \frac{\log 1/\delta}{\gamma - 1}, \delta)$-DP for any $\delta \in (0, 1)$.*

As a result of Theorem 4.5, PATE (with either hard or soft labels) satisfies $({}^{m\gamma}/_{\sigma^2} + \frac{\log 1/\delta}{\gamma - 1}, \delta)$-DP.

4.10.4 Experimental analysis (Ext)

This section reports detailed information about the experimental setting as well as additional results conducted on the Income, Bank, Parkinsons, Credit Card and UTKFace datasets.

4.10.5 Setting and datasets

Computing Infrastructure All of our experiments are performed on a distributed cluster equipped with Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz and 8GB of RAM.

Software and Libraries All models and experiments were written in Python 3.7. All neural network classifier models in our chapter were implemented in Pytorch 1.5.0.

The Tensorflow Privacy package was also employed for computing the privacy loss.

Datasets This chapter evaluates the fairness analysis of PATE on the following four UCI datasets: *Bank*, *Income*, *Parkinsons*, *Credit card* and UTKFace dataset. A descriptions of each dataset is reported as follows:

1. **Income** (Adult) dataset, where the task is to predict if an individual has low or high income, and the group labels are defined by race: *White* vs *Non-White* [18].

2. **Bank** dataset, where the task is to predict if a user subscribes a term deposit or not and the group labels are defined by age: *people whose age is less than vs greater than 60 years old* [18].

3. **Parkinsons** dataset, where the task is to predict if a patient has total UPDRS score that exceeds the median value, and the group labels are defined by gender: *female vs male* [77].

4. **Credit Card** dataset, where the task is to predict if a customer defaults a loan or not. The group labels are defined by gender: *female vs male* [23].

5. **UTKFace** dataset, where the task is to predict the gender of a given facial image. The group labels are defined based on the following 9 age ranges: 0-10, 10-15, 15-20, 20-25, 25-30, 30-40, 40-50, 50-60, 60-120. [61]

On each dataset we perform standardization to render all input features with zero mean and unit standard deviation. Each dataset was partitioned into three disjoint subsets: private set, public train, and test set, as follows. We randomly select 75% of the dataset to use as private data and the rest for public data. For the public data, $m = 200$ samples are randomly selected to train the student model, and the rest of the data is used as a test set to evaluate that model.

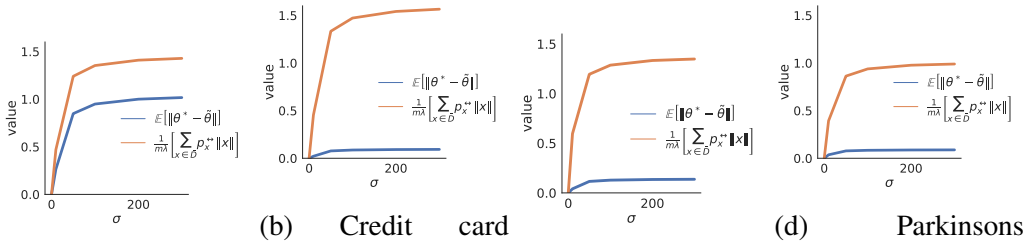
Models' Setting

To visually show how tight the upper bound from Corollary 4.1 is, the chapter uses a logistic regression model with 1000 runs to estimate the expected model deviation $\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right] = \mathbb{E}\left[\|\tilde{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|\right]$.

For other experiments, the chapter uses a neural network with with two hidden layers and nonlinear ReLU activations for both the ensemble and student models. All reported metrics are an average of 100 repetitions, used to compute the empirical expectations. The batch size for stochastic gradient descent is fixed to 32 and the learning rate is $\eta = 1e-4$.

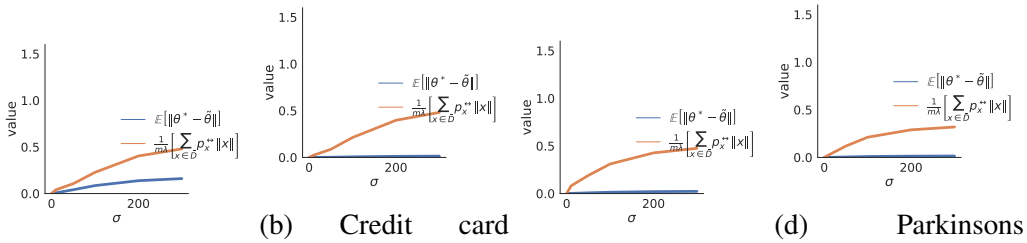
4.10.6 Upper bound of the expected model deviation

The following provides empirical results on Corollary 5.1 on four benchmark datasets. As indicated in this corollary, the expected model deviation is bounded by $\frac{1}{m\lambda}\left[\sum_{\boldsymbol{x}\in\bar{D}} p^{\leftrightarrow}_{\boldsymbol{x}}\|\boldsymbol{x}\|\right]$. To visualize how tight the bounds are we report the RHS and LHS values of eq:8 on different datasets. We run with two settings: $k = 20, \lambda = 20$ in Figure 4.8 and $k = 200, \lambda = 100$ in Figure 4.9.



(a) Bank dataset (b) Credit card dataset (c) Income dataset (d) Parkinsons dataset

Fig. 4.8: Upper bound of the expected model deviation on 4 datasets with $\lambda = 20, k = 20$.



(a) Bank dataset (b) Credit card dataset (c) Income dataset (d) Parkinsons dataset

Fig. 4.9: Upper bound of the expected model deviation on 4 datasets with $\lambda = 100, k = 200$.

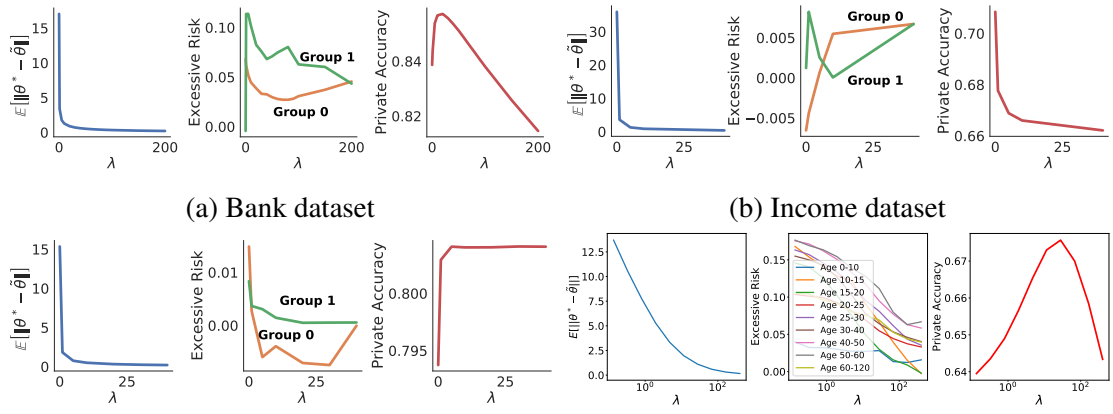
4.10.7 The impact of regularization parameter

This section provides further empirical supports regarding impact of the regularization parameter λ to the accuracy and fairness trade-off. As seen from Theorem 4.2, increasing λ reduces the model deviation which in turns decreases the group excessive risk $R(\bar{D}_{\leftarrow a})$ by Theorem 4.2 from the main text. On the other hand, large regularization can intuitively impacts negatively to the model accuracy. This was verified empirically in Figure 4.10 which shows how model deviation(left), excessive risk difference between two groups (middle) and utility(right) vary according to λ.

4.10.8 The impact of teachers ensemble size k

This section illustrates the effect of teacher ensemble sizes k to: 1) flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$, and 2) the trade-offs among model deviation $\mathbb{E}\left[\Delta_{\tilde{\boldsymbol{\theta}}}\right]$, model's fairness and utilities.

First, Theorem 4.3 from the main text shows that larger k values correspond to smaller flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$. We provide more empirical evidence on other datasets and report the dependency between flipping probability with number of teachers k in Figure 4.11. It can be observed consis-

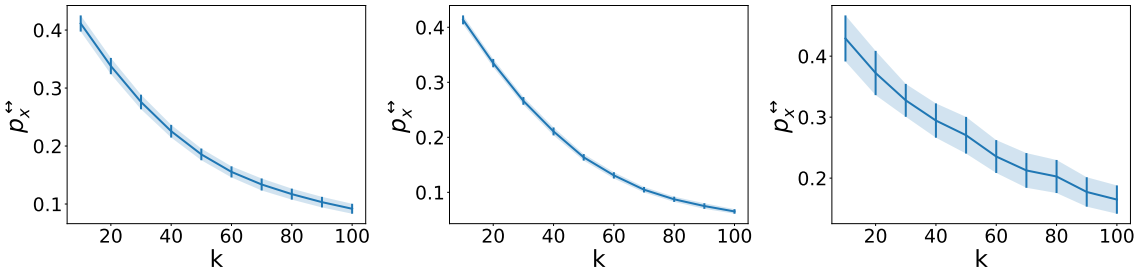


(a) Bank dataset (b) Income dataset

(c) Parkinsons dataset (d) UTKFace dataset

Fig. 4.10: Expected model deviation (left), empirical risk (middle), and model accuracy (right) as a function of the regularization. The experiments are performed with the following settings: $k = 150, \sigma = 50$.

.

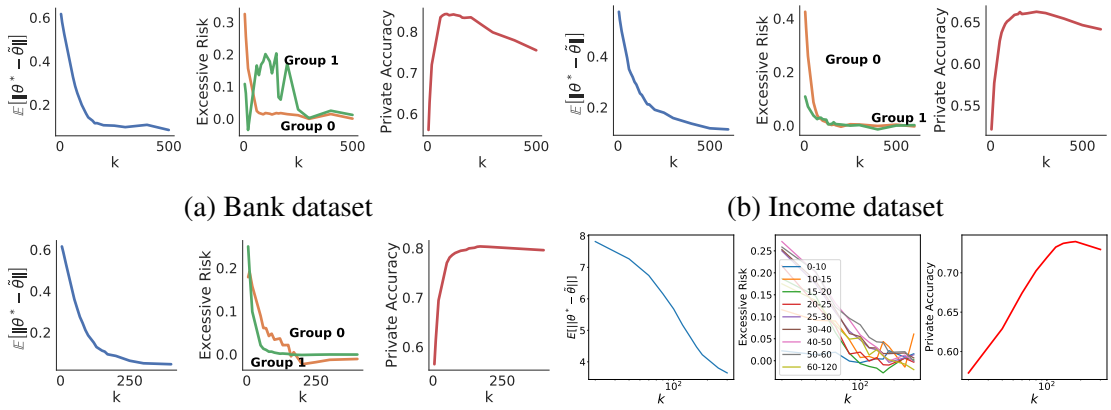


(a) Bank dataset (b) Income dataset (c) Parkinsons dataset

Fig. 4.11: Average flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ for samples $\boldsymbol{x} \in \bar{D}$ as a function of the ensemble size k.

tently on all datasets, the more number of teachers k, the smaller the flipping probability $p_{\boldsymbol{x}}^{\leftrightarrow}$ over all samples $\boldsymbol{x}$ is.

Second, regarding to the fairness analysis, similar to the previous subsection, we provide additional empirical supports on the effects of k on the model deviation, the difference between the group excessive risk, and the utility of the PATE models. We report these metrics on the other three benchmarks datasets in Figure 4.12. A similar trend with the regularization parameter λ also holds for the parameter k here. When the parameter k is increased to a large enough value, both model deviation and accuracy decreases, but the unfairness measured by the excessive risk difference between two groups reduces. This can be explained by looking again Figure 4.11 and Theorem 4.3



(a) Bank dataset (b) Income dataset

(c) Parkinsons dataset (d) UTKFace dataset

Fig. 4.12: Expected model deviation (left), empirical risk (middle), and model accuracy (right) as a function of the ensemble size. The experiments are performed with the following settings: $\lambda = 100$, $\sigma = 50$.

.

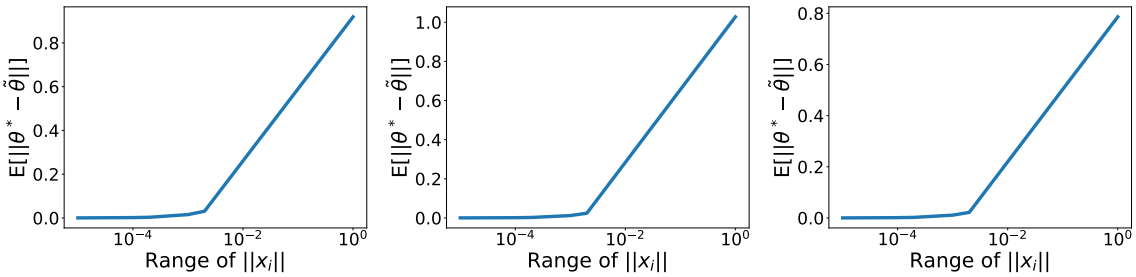
from the main text. A large number of teachers k results to a smaller flipping probability which in turns reduces the model deviation. By Theorem 4.2 a small model deviation can reduce the level of unfairness.

4.10.9 The impact of the data input norm

This section provides further experimental results regarding relation between (1) the input norm with the private model deviation and (2) the input norm with its excessive risk.

Regarding the first relation, Corollary 5.1 from the main text implies that the smaller the input norm $\|\boldsymbol{x}\|$ is the smaller the model deviation is. For each dataset, we then vary the range of the input norm $\|\boldsymbol{x}\|$ and report the associated values of the expected model deviation in Figure 4.13. It can be seen clearly from the Figure 4.13, a monotone connection between input norm and the model deviation which verifies the statement from Corollary 5.1.

On the other hand, the input norm can affect the excessive risk by Lemma 4.1 of the Appendix, the individuals or group of individuals of large gradient norm can suffer from large excessive risk. In other words, the individuals of large data norm which are often observed at the tail of data can loose more accuracy. To confirm such claims, we report in Figure 4.14 the Spearman



(a) Income dataset (b) Bank dataset (c) Parkinsons dataset

Fig. 4.13: Relation between input norm and model deviation.

correlation between input norm and the excessive risk at individual levels. On all datasets, we can see obviously a positive relationship between data input norm and the excessive risk.

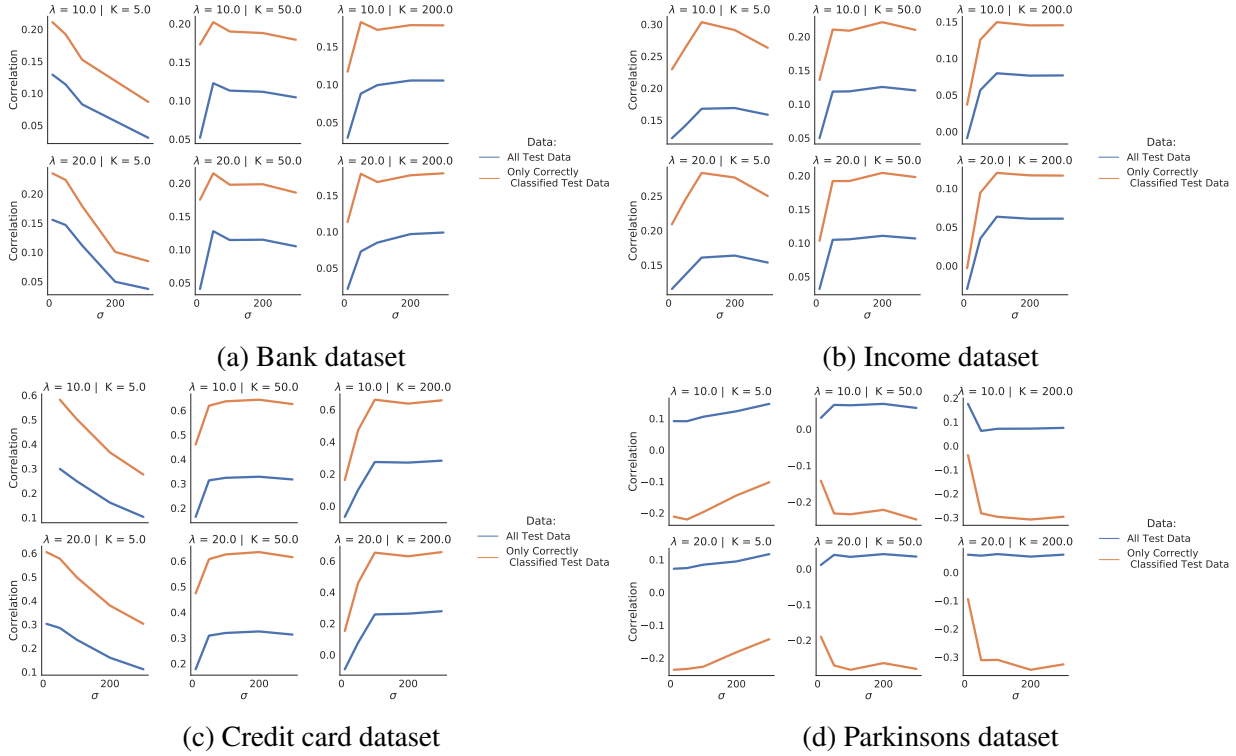
4.10.10 Connection between input norm and smoothness parameter β_a

It is noted that the smoothness parameter $\beta_{\boldsymbol{a}}$ captures the local flatness of the loss function of a particular group a. Consider the logistic regression classifier, then the smoothness parameter $\ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y)$ for one particular data point is given by $\beta_x = 0.25\|\boldsymbol{x}\|$ [114]. Recall the following important property of the smooth function: If $L = \sum_i \ell_i$ and each ℓ_i is β_i-smooth then L is $\max_i \beta_i$-smooth. Because of that, the smoothness parameter β_a for one particular group a is given by: $\beta_a = 0.25 \max_{\boldsymbol{x} \in D_a} \|\boldsymbol{x}\|$

The above clearly illustrates the relationship between input norms $\|\boldsymbol{x}\|$ and the smoothness parameters $\beta_{\boldsymbol{a}}$.

4.10.11 Connection between input norm and gradient norm

In the main text, we have described, for logistic regression classifiers, there is a strong relation between the individual input norm $\|\boldsymbol{x}\|$ and their gradient norm at optimal parameter $\|\nabla_{\boldsymbol{\theta}^\star} \ell(\bar{f}_{\boldsymbol{\theta}^\star}(\boldsymbol{x}), y)\|$. In this subsection, we extend the analysis for non-linear model. In particular, we show a similar connection between the gradient norm and the input norm for a neural network with a single hidden layer. We start by considering the following settings:



(a) Bank dataset

(b) Income dataset

(c) Credit card dataset

(d) Parkinsons dataset

Fig. 4.14: Correlation between the excessive risk and input norm on 5 datasets. The experiments are performed with the following settings: $\lambda = 100, \sigma = 50, k = 150$.

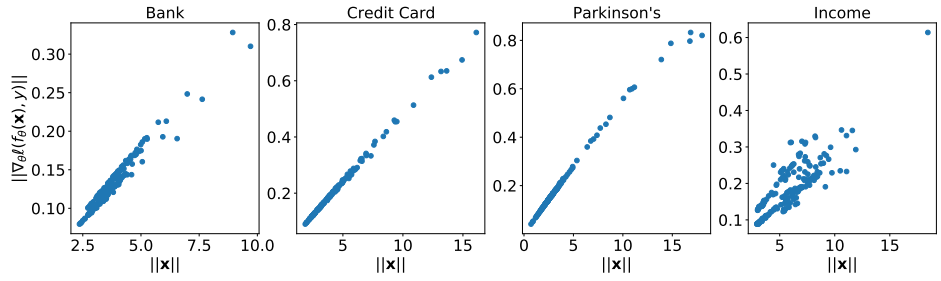


Fig. 4.15: Relation Between Gradient Norm and Input Norm on all datasets.

Settings Consider a neural network model $\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}) \overset{\text{def}}{=} \textit{softmax}\left(\overset{\star}{\boldsymbol{\theta}}_1^T \tau(\overset{\star}{\boldsymbol{\theta}}_2^T \boldsymbol{x})\right)$ where $\boldsymbol{x} = (\boldsymbol{x}^i)_{i=1}^d$ is a d dimensional input vector, the parameters $\overset{\star}{\boldsymbol{\theta}}_2 \in \mathbb{R}^{d\times H}, \overset{\star}{\boldsymbol{\theta}}_1 \in \mathbb{R}^{H\times C}$ and the cross entropy loss $\ell(\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y) = -\sum_{c=1}^C y_c \log \bar{f}_{\overset{\star}{\boldsymbol{\theta}},c}(\boldsymbol{x})$ where $\tau(\cdot)$ is a proper activation function, e.g., a sigmoid function. Let $\boldsymbol{O} = \tau(\overset{\star}{\boldsymbol{\theta}}_2^T \boldsymbol{x}) \in \mathbb{R}^H$ be the vector $(O_1, \ldots, O_H)$ of H hidden nodes of the network. Denote the variables $h_j = \sum_{i=1}^d \overset{\star}{\boldsymbol{\theta}}_{2,j,i} \boldsymbol{x}^i$ as the j-th hidden unit before the activation function. Next, denote $\overset{\star}{\boldsymbol{\theta}}_{1,j,k} \in \mathbb{R}$ as the weight parameter that connects the j-th hidden unit h_j with the c-th output unit $\bar{f}_c$ and $\overset{\star}{\boldsymbol{\theta}}_{2,i,j} \in \mathbb{R}$ as the weight parameter that connects the i-th input unit $\boldsymbol{x}^i$ with the j-th hidden unit h_j.

Given the settings above, we now show the dependency between gradient norm and input norm. First notice that we can decompose the gradients norm of this neural network into two layers as follows:

$$\|\nabla_{\overset{\star}{\boldsymbol{\theta}}} \ell(\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y)\|^2 = \|\nabla_{\overset{\star}{\boldsymbol{\theta}}_1} \ell(\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y)\|^2 + \|\nabla_{\overset{\star}{\boldsymbol{\theta}}_2} \ell(\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y)\|^2. \tag{4.45}$$

We will show that $\nabla_{\overset{\star}{\boldsymbol{\theta}}_2} \ell(\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y)\| \propto \|\boldsymbol{x}\|$.

Notice that:

$$\|\nabla_{\overset{\star}{\boldsymbol{\theta}}_2} \ell(\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y)\|^2 = \sum_{i,j} \|\nabla_{\overset{\star}{\boldsymbol{\theta}}_{2,i,j}} \ell(\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y)\|^2.$$

Applying, Equation (14) from [108], it follows that:

$$\nabla_{\overset{\star}{\boldsymbol{\theta}}_{2,i,j}} \ell(\bar{f}_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y) = \sum_{c=1}^C \left(y_c - \bar{f}_{\overset{\star}{\boldsymbol{\theta}},c}(\boldsymbol{x})\right) \overset{\star}{\boldsymbol{\theta}}_{1,j,c} \left(O_j(1 - O_j)\right) \boldsymbol{x}^i, \tag{4.46}$$

which highlights the dependency of the gradient norm $\|\nabla_{\overset{\star}{\boldsymbol{\theta}}_2} \ell(\bar{f}_{\boldsymbol{\theta}}(\boldsymbol{x}), y)\|$ and the input norm $\|\boldsymbol{x}\|$. Figure 4.15 provides an empirical evidence for this dependency on all four datasets used in our analysis. It can be seen clearly a strong positive correlation between input norm and the gradient norm at individual levels on all datasets from Figure 4.15.

4.10.12 Effectiveness of mitigation solution

This subsection provides extended empirical results regarding the effectiveness of our proposed mitigation solution which was presented in Section 4.8.

We report the comparison between training PATE with hard and soft labels when $k = 20$ in Figure 4.16 and when $k = 150$ in Figure 4.17. These figures again illustrate the effects of the proposed mitigating solution in terms of utility/fairness tradeoff on the private student model. The top subplots of each figure show the group excessive risks $R(\bar{D}_{\leftarrow 0})$ and $R(\bar{D}_{\leftarrow 1})$ associated with two groups while the bottom subplots illustrate the accuracy of the model, at increasing of the privacy loss ϵ. Recall that our mitigation solution does not require the availability of group labels during training. This challenging settings are of importance under the scenario when it is not feasible to collect or use protected features (e.g., under GDPR).

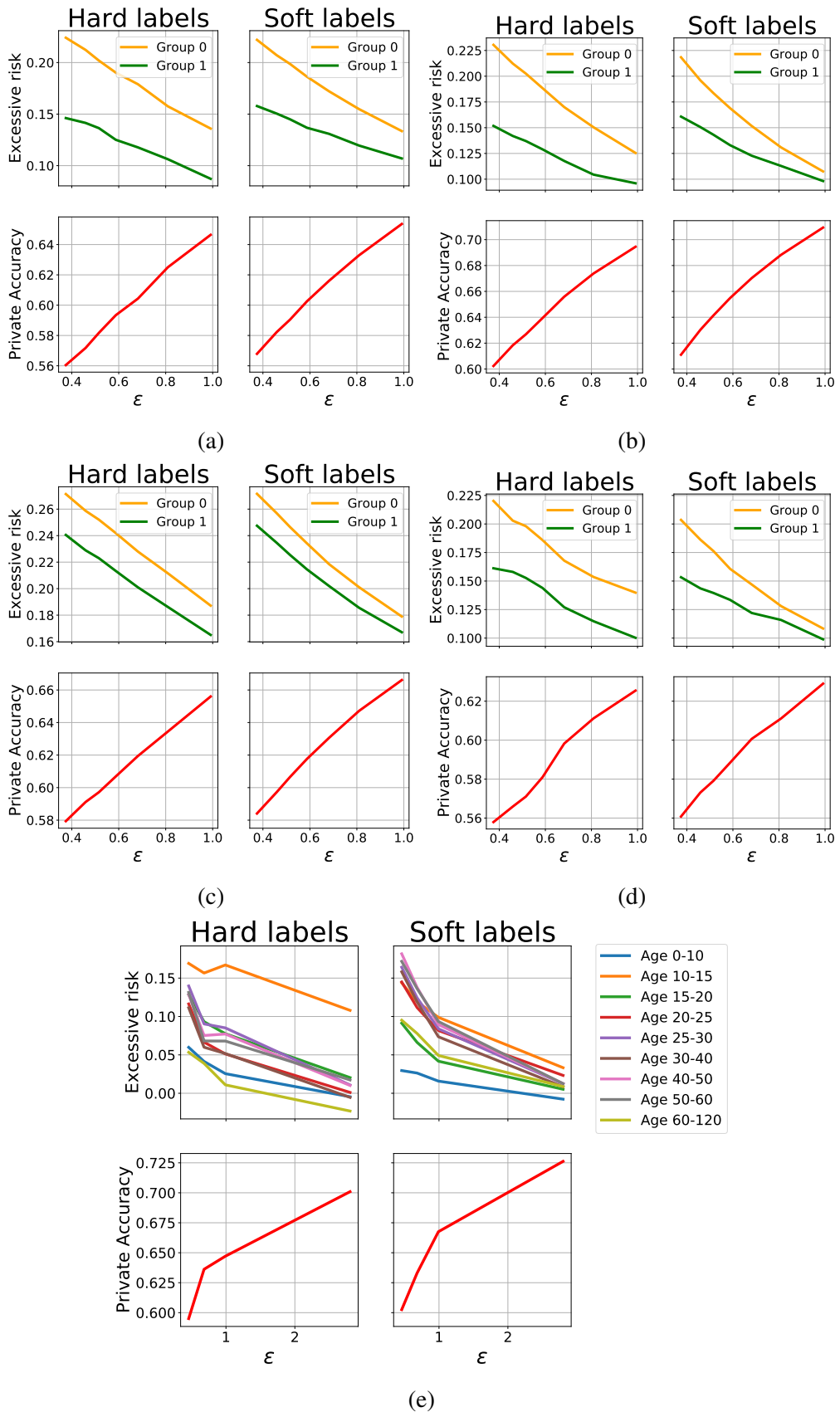


Fig. 4.16: Comparison between training privately PATE with hard labels and soft labels in term of fairness (top subfigures) and utility(bottom subfigures) on (a) Bank, (b) Credit card, (c) Income (d) Parkinsons, (e) UTKFace dataset. Here for each dataset, the number of teachers $k = 20$.

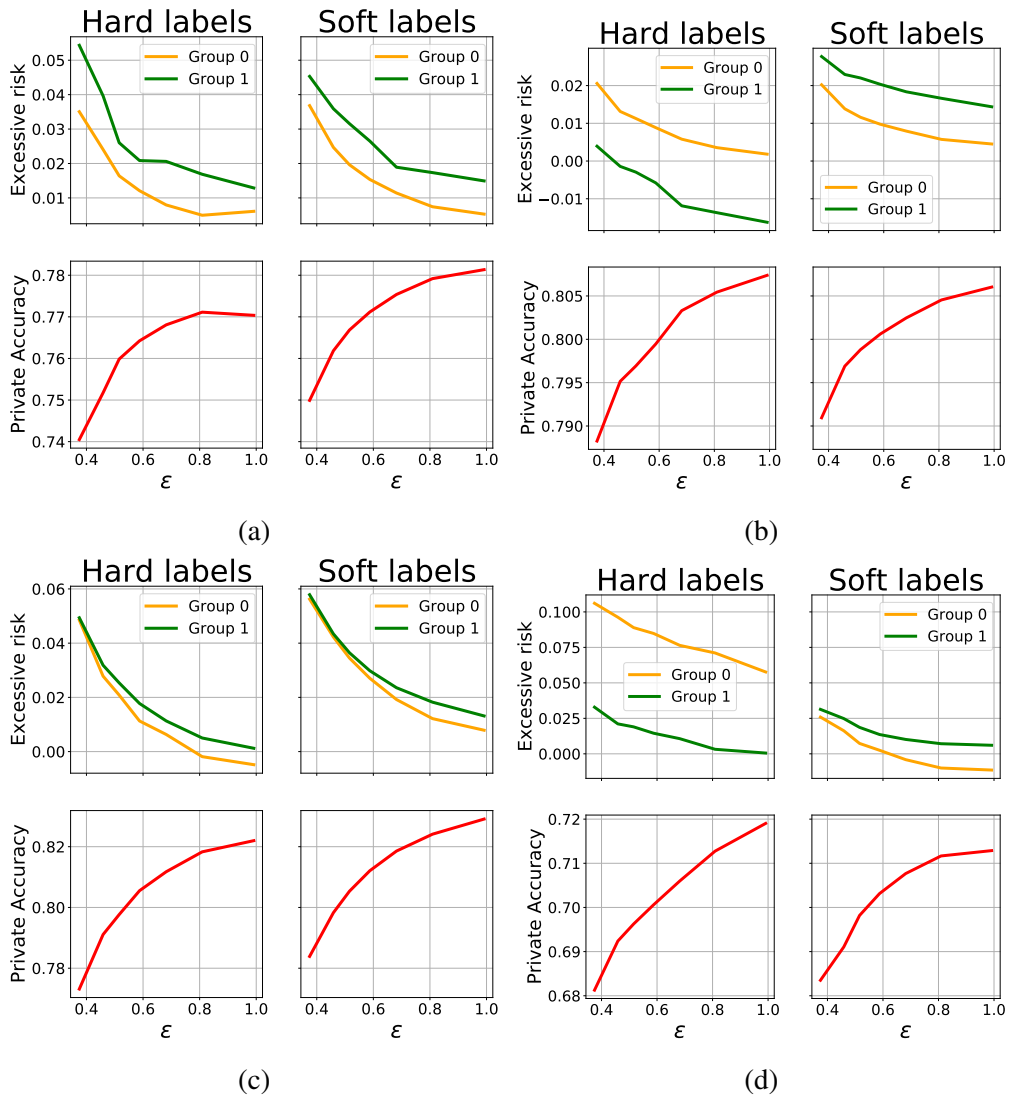


Fig. 4.17: Comparison between training privately PATE with hard labels and soft labels in term of fairness (top subfigures) and utility(bottom subfigures) on (a) Bank, (b) Credit card, (c) Income, and (d) Parkinsons Here for each dataset, the number of teachers $k = 150$.

CHAPTER 5
PRUNING HAS A DISPARATE IMPACT ON MODEL ACCURACY

Network pruning is a widely-used compression technique that is able to significantly scale down overparameterized models with minimal loss of accuracy. This chapter shows that pruning may create or exacerbate disparate impacts. The chapter sheds light on the factors to cause such disparities, suggesting differences in gradient norms and distance to decision boundary across groups to be responsible for this critical issue. It analyzes these factors in detail, providing both theoretical and empirical support, and proposes a simple, yet effective, solution that mitigates the disparate impacts caused by pruning.

5.1 Introduction

As deep learning models evolve and become more powerful, they also become larger and more costly to store and execute. The trend hinders their deployment in resource-constrained platforms, such as embedded systems or edge devices, which require efficient models in time and space. To address this challenge, studies have developed a variety of techniques to prune the relatively insignificant or insensitive parameters from a neural network while ensuring competitive accu-

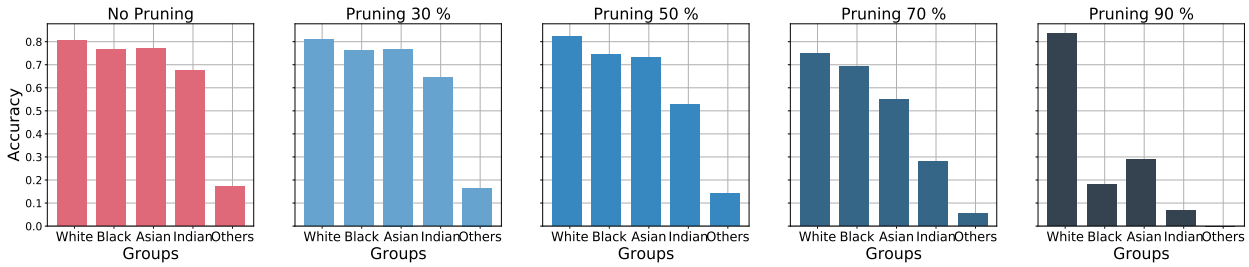


Fig. 5.1: Accuracy of each demographic group in the UTK-Face dataset using Resnet18 [56], at the increasing of the pruning rate.

racy [8, 15, 20, 104, 107, 111, 143]. When a model needs to be developed to fit given and certain requirements in size and resource consumption, a pruned model which is derived from a large, rigorously-trained, and (often) over-parameterized model, is regarded as a de-facto standard. That is because it performs incomparably better than a same-size dense model which is trained from scratch, when the same amount of effort and resources are invested.

In spite of its strengths, pruning has been showed to induce or exacerbate disparate effects in the accuracy of the resulting reduced models [58, 59]. Intuitively, the removal of model weights affects the process in which the network separates different classes, which can have contrasting consequences for different groups of individuals. This chapter further shows that the accuracy of the pruned models tends to increase (decrease) more in classes that had already high (low) accuracy in the original model, leading to a "the rich get richer" and "the poor get poorer" effect. This *Matthew* effect is illustrated in Figure 7.1. The figure shows the accuracy of a facial recognition task on different demographic groups for several pruning rates (indicating the percentage of parameters removed from the original models). Notice how the accuracy of the majority group (White) tends to increase while that of the minority groups tends to decrease as the pruning ratio increases.

Following these observations, we shed light on the factors to cause such disparities. The theoretical findings suggest the presence of two key factors responsible for why accuracy disparities arise in pruned models: **(1)** disparity in *gradient norms* across groups, and **(2)** disparity in *Hessian matrices* associated with the loss function computed using a group's data. Informally, the former carries information about the groups' local optimality, while the latter relates to model separability.

We analyze these factors in detail, providing both theoretical and empirical support on a variety of settings, networks, and datasets. By recognizing these factors, we also develop a simple yet effective training technique that largely mitigates the disparate impacts caused by pruning. The method is based on an alteration of the loss function to include components that penalize disparity of the average gradient norms and distance to decision boundary across groups.

These findings are significant: *Pruning is a key enabler for neural network models in embedded systems with deployments in security cameras and sensors for autonomous devices for applications where fairness is an essential need. Without careful consideration of the fairness impact of these techniques, the resulting models can have profound effects on our society and economy*.

Related work

Fairness and network pruning have been long studied in isolation. The reader is referred to the related chapters and surveys on fairness [14, 24, 36, 54, 84] and pruning [8, 15, 20, 104, 107, 111, 121, 143] for a review on these areas.

The recent interest in assessing societal values of machine learning models has seen an increase of studies at the intersection of different properties of a learning model and their effects on fairness. For example, [139] studies the setting of adversarial robustness and show that adversarial training introduces unfair outcomes in term of accuracy parity [145]. [148] show that semisupervised settings can introduce unfair outcomes in the resulting accuracy of the learned models. Finally, several authors have also shown that private training can have unintended disparate impacts to the resulting models' outputs [12, 46, 123, 131, 147] and downstream decisions [71, 125].

Network compression has also been shown to have a profound impact towards the model fairness. For example, several works observed empirically that network compression may amplify unfairness in different learning tasks [58,59,65,92]. Most of the focus has been on vision tasks and in identifying the set of *Pruning Identified Exemplars* (PIEs), the samples that are impacted most under the compression scheme and conclude that PIEs belongs to low frequency groups (those observed at the tail of the data distribution). [19] further investigate how bias could be evaluated and

mitigated in pruned neural networks using knowledge distillation while [60] observed empirically that knowledge distillation processes may produce unfair student models. The impact of network compression towards fairness has also been assessed in natural language tasks. For example, [35] and [136] empirically measure the robustness of compressed large language models, while [9] look into how compression schemes affects data-limited regimes. Finally, [138] investigate ways to improve fairness in generative language models by compressing them. We also note that, concurrently to this work, [51] studied the relative distortion in recall for various classes. They show that pruning has a Matthews effect on the recall for various classes and propose an algorithm to attenuate such an effect.

This chapter builds on this body of work and their important empirical observations and provides a step towards a deeper theoretical understanding of the fairness issues arising as a result of pruning. It derives conditions and studies the causes of unfairness in the context of pruning as well as it introduces mitigating guidelines.

5.2 Problem settings and goals

The chapter considers datasets D consisting of n datapoints $(\boldsymbol{x}_i, a_i, y_i)$, with $i \in [n]$, drawn i.i.d. from an unknown distribution Π. Therein, $\boldsymbol{x}_i \in \mathcal{X}$ is a feature vector, $a_i \in \mathcal{A}$ with $\mathcal{A} = [m]$ (for some finite m) is a demographic group attribute, and $y_i \in \mathcal{Y}$ is a class label. For example, consider the case of a face recognition task. The training example feature $\boldsymbol{x}_i$ may describe a headshot of an individual, the protected attribute a_i may describe the individual's gender or ethnicity, and y_i represents the identity of the individual. The goal is to learn a predictor $f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$, where $\boldsymbol{\theta}$ is a k-dimensional real-valued vector of parameters that minimizes the empirical risk function:

$$\overset{\star}{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, J(\boldsymbol{\theta}; D) = \frac{1}{n} \sum_{i=1}^{n} \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i), \tag{5.1}$$

where $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ is a non-negative *loss function* that measures the model quality.

We focus on analyzing properties arising when extracting a small model $f_{\bar{\boldsymbol{\theta}}}$ with $\bar{\boldsymbol{\theta}} \subset \overset{\star}{\boldsymbol{\theta}}$ of size

$|\bar{\boldsymbol{\theta}}| = \bar{k} \ll k$. Model $f_{\bar{\boldsymbol{\theta}}}$ is constructed by pruning the least important values or filters from vector $\overset{\star}{\boldsymbol{\theta}}$ (i.e., those with smaller values in magnitude) according to a prescribed criterion, such as an ℓ_p norm [88,107]. The chapter focuses on understanding the fairness impacts (as defined next) arising when pruning general classifiers, such as neural networks.

Fairness The fairness analysis focuses on the notion of *excessive loss*, defined as the difference between the original and the pruned risk functions over some group $a \in \mathcal{A}$:

$$R(a) = J(\bar{\boldsymbol{\theta}}; D_a) - J(\overset{\star}{\boldsymbol{\theta}}; D_a), \tag{5.2}$$

where D_a denotes the subset of the dataset D containing samples $(\boldsymbol{x}_i, a_i, y_i)$ whose group membership $a_i = a$. Intuitively, the excessive loss represents the change in loss (and thus, in accuracy) that a given group experiences as a result of pruning. Fairness is measured in terms of the maximal *excessive loss difference*, also referred to as *fairness violation*:

$$\xi(D) = \max_{a,a' \in \mathcal{A}} |R(a) - R(a')|, \tag{5.3}$$

defining the largest excessive loss difference across all protected groups. (Pure) fairness is achieved when $\xi(D) = 0$, and thus a fair pruning method aims at minimizing the excessive loss difference.

The goal of this chapter is to shed light on why fairness issues arise (i.e., $R(a) > 0$) as a result of pruning, why some groups suffer more than others (i.e., $R(a) > R(a')$), and what mitigation measures could be taken to minimize unfairness due to pruning.

We use the following notation: variables are denoted by calligraph symbols, vectors or matrices by bold symbols, and sets by uppercase symbols. Finally, $\|\cdot\|$ denotes the Euclidean norm and we use $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ to refer to the model' *soft* outputs. All proofs are reported in Section 5.8.1.

Network pruning In the scope of this chapter, we focus on single shot network pruning.[CITE] This network pruning scheme consists of the following three steps:

1. **Standard training** In the first step, we perform standard training as in Equation 6.1 to obtain optimal parameters θ^*.

2. **Pruning** In the second step, we prune/remove the least important parameters/filters based on some criteria (e.g., l_1/l_2 norm based) from θ_* to obtain $\bar{\theta}$.

3. **Fine-tuning** In the last step, we fine-tune the pruned parameter $\bar{\theta}$ to obtain parameter $\tilde{\theta}$

We are interested in quantifying the disparate impact on accuracy of using the $\tilde{\theta}$ compared to the before pruning parameter θ_* to different classes.

5.3 Fairness analysis in pruning: Roadmap

To gain insights on how pruning may introduce unfairness, we start with providing a useful upper bound for a group's excessive loss. Its goal is to isolate key aspects of model pruning that are responsible for the observed unfairness. The following discussion assumes the loss function $\ell(\cdot)$ to be at least twice differentiable, which is the case for common ML loss functions, such as mean squared error or cross entropy loss.

Theorem 5.1. *The* excessive loss *of a group $a \in \mathcal{A}$ is upper bounded by*[1]*:*

$$R(a) \leq \left\|\boldsymbol{g}_a^{\ell}\right\| \times \left\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right\| + \frac{1}{2}\lambda\left(\boldsymbol{H}_a^{\ell}\right) \times \left\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right\|^2 + \mathcal{O}\left(\left\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right\|^3\right), \tag{5.4}$$

where $\boldsymbol{g}_a^{\ell} = \nabla_{\boldsymbol{\theta}} J(\overset{\star}{\boldsymbol{\theta}}; D_a)$ is the vector of gradients associated with the loss function ℓ evaluated at $\overset{\star}{\boldsymbol{\theta}}$ and computed using group data D_a, $\boldsymbol{H}_a^{\ell} = \nabla^2_{\boldsymbol{\theta}} J(\overset{\star}{\boldsymbol{\theta}}; D_a)$ is the Hessian matrix of the loss function ℓ, at the optimal parameters vector $\overset{\star}{\boldsymbol{\theta}}$, computed using the group data D_a (henceforth simply referred to as group hessian*), and $\lambda(\Sigma)$ is the maximum eigenvalue of a matrix Σ.*

The bound above follows from a second order Taylor expansion of the loss function, Cauchy-Schwarz inequality, and properties of the Rayleigh quotient.

[1]With a slight abuse of notation, the results refer to $\bar{\boldsymbol{\theta}}$ as the homonymous vector which is extended with $k - \bar{k}$ zeros.

Notice that, in addition to the difference in the original and pruned parameters vectors, two key terms appear in Equation (5.4): **(1)** The norms of the gradients $\boldsymbol{g}_a^\ell$ and **(2)** the maximum eigenvalue of the Hessian matrix $\boldsymbol{H}_a^\ell$ for a group a. Informally, the former is associated with the groups' local optimality while the latter relates to the ability of the model to separate the groups data. As we will show next these components represent the main sources of unfairness due to model pruning.

The following is an important corollary of Theorem 5.1. It shows that the larger the pruning, the larger will be the excessive loss for a given group.

Corollary 5.1. *Let $\bar{k}$ and $\bar{k}'$ be the size of parameter vectors $\bar{\boldsymbol{\theta}}$ and $\bar{\boldsymbol{\theta}}'$, respectively, resulting from pruning model $f_{\hat{\boldsymbol{\theta}}}$, where $\bar{k} < \bar{k}'$ (i.e., the former model prunes more weight than the latter one). Then, for any group $a \in \mathcal{A}$,*

$$\tilde{R}(a, \bar{\boldsymbol{\theta}}) \geq \tilde{R}(a, \bar{\boldsymbol{\theta}}'), \tag{5.5}$$

where $\tilde{R}(a, \boldsymbol{\omega})$ is the excessive loss upper bound computed using pruned model parameters $\boldsymbol{\omega}$ (Eq. (5.4)*).*

The corollary above indicates that the excess risk for a group increases as the pruning regime increase. Building on this result, the chapter illustrates next why unfairness can become more significant as the pruning regime increases.

The next sections analyze the effect of gradient norms and the Hessian to unfairness in the pruned models. The theoretical claims are supported and complemented by analytical results. These results use the UTKFace dataset [144] for a vision task whose goal is to classify ethnicity. The experiments use a ResNet-18 architecture and the pruning counterparts remove the $P\%$ parameters with the smallest absolute values for various P. All reported metrics are normalized and an average of 10 repetitions. While the theoretical analysis focuses on the notion of disparate impacts under the lens of excessive loss, the empirical results report differences in accuracy of the resulting models. The empirical results thus reflect the setting commonly adopted when measuring accuracy parity [145].

We report a glimpse of the empirical results, with the purpose of supporting the theoretical

claims, and extended experiments, as well as additional descriptions of the datasets and settings, are reported in Section 5.8.2.

5.4 Why disparity in groups' gradients causes unfairness?

This section analyzes the effect of gradients norms on the unfairness observed in the pruned models. In more detail, it shows that unbalanced datasets result in a model with large differences in gradient norms between groups (Proposition 5.1), it connects gradients norms for a group with the resulting model errors in such a group (Proposition 5.2), and connects these concepts with the excessive loss (Theorem 5.1) to show that unfairness in model pruning is largely controlled by the difference in gradient norms among groups.

Gradient norms and group sizes. The section first shows that imbalanced datasets lead a model to have imbalanced gradient norms across groups. The following result assumes that the training converges to a local minima.

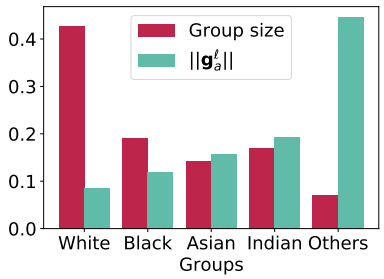


Fig. 5.2: Group size vs. gradient norms.

Proposition 5.1. *Consider a dataset with two groups, a and b with $|D_a| \geq |D_b|$. Then $\left\|\boldsymbol{g}_a^\ell\right\| \leq \left\|\boldsymbol{g}_b^\ell\right\|$.*

That is, groups with more data samples will result in smaller gradients norms than groups with fewer data samples and vice-versa. Figure 5.2 illustrates Proposition 5.1. The plot shows the relation between groups sizes $|D_a|$ and their associated gradient norms $\|\boldsymbol{g}_a^\ell\|$ on the UTK dataset and settings described above. Notice the strong trend between decreasing group sizes and increasing gradient norms for such groups. These theoretical considerations can be used to explain why underrepresented groups are often subject to larger performance impacts after network pruning [59]. These groups tend to exhibit large gradient norms at convergence, relative to other groups, thus, by Theorem 5.1, they are also subject to larger excessive losses due to pruning.

Gradient norms and accuracy. Next, the section shows a strong connection between the gradient norms of a group and its associated accuracy. The following assumes the models adopt a cross entropy loss (or mean squared error for regression tasks, as shown Section 5.8.1).

Proposition 5.2. *For a given group $a \in \mathcal{A}$, gradient norms can be upper bounded as:*

$$\|\boldsymbol{g}_a^\ell\| \in \mathcal{O}\left(\sum_{(\boldsymbol{x},y)\in D_a} \underbrace{\|f_{\boldsymbol{\theta}}^\star(\boldsymbol{x}) - y\|}_{Error} \times \left\|\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}^\star(\boldsymbol{x})\right\|\right).$$

The above relates gradient norms with an error measure of the classifier to a target label multiplied by the gradient of the predictions. For example, in a classification task with cross entropy loss, $\ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y) = -\sum_{z\in\mathcal{Y}} f_{\boldsymbol{\theta}}^z(\boldsymbol{x})\boldsymbol{y}^z$, where $f_{\boldsymbol{\theta}}^z(\boldsymbol{x})$ represents the z-th element of the output associated with the soft-max layer of model $f_{\boldsymbol{\theta}}$, and $\boldsymbol{y}$ is a one-hot encoding of the true label y, with $\boldsymbol{y}^z$ representing its z-th element, then,

$$\begin{aligned}
\|\boldsymbol{g}_a\| = \|\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}; D_a,)\| &= \left\| {}^1\!/_{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} \nabla_f \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y) \times \nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}(\boldsymbol{x}) \right\| \\
&= \left\| {}^1\!/_{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} (f_{\boldsymbol{\theta}}(\boldsymbol{x}) - \boldsymbol{y}) \times \nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}(\boldsymbol{x}) \right\| \\
&\leq {}^1\!/_{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} \|f_{\boldsymbol{\theta}}(\boldsymbol{x}) - \boldsymbol{y}\| \times \|\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}(\boldsymbol{x})\| .
\end{aligned}$$

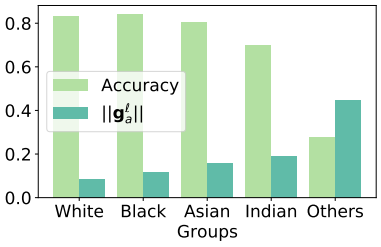
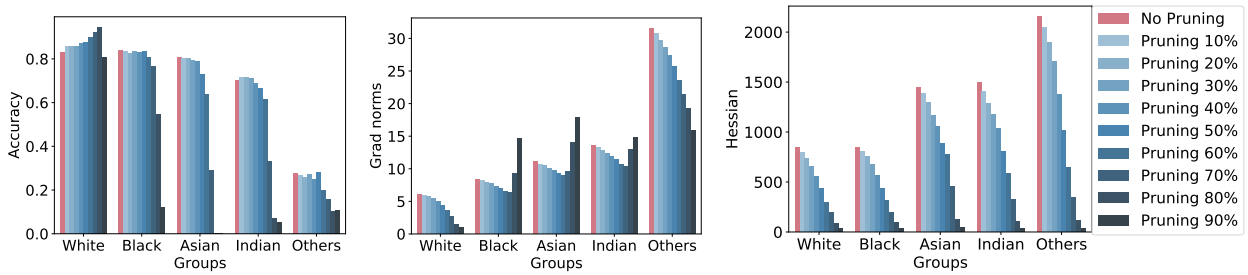


Fig. 5.3: Accuracy vs. gradient norms.

A similar observation holds for mean square error loss, as illustrated in Section 5.8.1. The observation above sheds light on the correlation between the prediction error of a group and its model gradients. This relation is emphasized in Figure 5.3, which illustrates that the gradient norm for a given group increases as its prediction accuracy decreases.

Proposition 5.2 allows us to link the gradient norms with the group accuracy of the resulting model, which, together with the result above will be useful to reason about the impact of gradient norms on the disparities in the group excessive losses.



(a) Accuracy (b) Gradient Norm: $\|\boldsymbol{g}_a^\ell\|$ (c) Group Hessian: $\lambda(\boldsymbol{H}_a^\ell)$

Fig. 5.4: Accuracy, gradient norm, and group Hessian max eigenvalues of each ethnicity group, before and after increasing pruning ratios for UTK-Face dataset. The percentage of data samples across groups *White, Black, Asian, Indian, and Others* is $\sim 0.42, 0.19, 0.15, 0.15, 0.07$, respectively.

The role of gradient norms in pruning. Having highlighted the connection between gradients norms of a group with the accuracy of the pruned model on such a group, this section provides theoretical intuitions on the role of gradient norms in the disparate group losses during pruning.

From Theorem 5.1, notice that the excessive loss is controlled by term $\|\boldsymbol{g}_a^\ell\| \times \|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|$. As already noted in Corollary 5.1, the term $\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|$ regulates the impact of pruning on the excessive loss, as the difference between the pruned and non-pruned parameters vectors directly depends on the pruning rate. For a fixed pruning rate, however, notice that groups with different gradient norms will have a disparate effect on the resulting term. In particular, groups with very small gradients norms (those generally associated with highly accurate predictions) will be less sensitive to the effects of the pruning rate. Conversely, groups with large gradient norms will be affected by the pruning rate to a greater extent, with larger pruning rates, *typically* reflecting in larger excessive losses.

These observations of the factors of disparity, accuracy, and group size, can also be appreciated empirically in Figures 5.4a and 5.4b. The plots report accuracy (a) and gradient norms (b) on the UTKFace datasets for a variety of pruning rates. Consider group *White* (containing 42% of the total samples) and *Others* (containing 7% of the total samples). The unpruned model has high accuracy on the former group and small gradient norms. The accuracy of this group is insensitive to various pruning rates and even increases at large pruning regimes. In contrast, group *Others*

has much lower accuracy and larger gradient norms in the unpruned model. As the pruning rate increase, their accuracies drastically drop. As a result, in high pruning regimes, this minority group exhibits poor accuracy and very high gradient norms.

Notice that the empirical results apply to much more complex settings than those which can be analyzed formally, thus they complement the theoretical observations.

5.5 Why disparity in groups' Hessians causes unfairness?

Having examined the properties of the groups gradients and their relation to unfairness in pruning, this section turns on analyzing how the Hessian associated with the loss function for a group is linked to the unfairness observed during pruning. In more detail, it connects the groups' Hessian to the distance to the decision boundary for the samples in that group and their resulting model errors (Theorem 5.2), it illustrates a strong positive correlation between groups' Hessian and gradient norms, and links these concepts with the excessive loss (Theorem 5.1) to show that unfairness in model pruning is controlled by the difference in maximum eigenvalues of the Hessians among groups.

Group Hessians and accuracy. The section first shows that groups presenting large Hessian values may suffer larger disparate impacts due to pruning, when compared with groups that have smaller Hessians. It does so by connecting the maximum eigenvalues of the groups Hessians with their distance to decision boundary and the group accuracy. The following result sheds light on these observations. It restricts its attention to models trained under binary cross entropy losses, for clarity of explanation, although an extension to a multi-class case is directly attainable.

Theorem 5.2. *Let f_θ be a binary classifier trained using a binary cross entropy loss. For any*

group $a \in \mathcal{A}$, the maximum eigenvalue of the group Hessian $\lambda(\boldsymbol{H}_a^\ell)$ can be upper bounded by:

$$\lambda(\boldsymbol{H}_a^\ell) \leq \frac{1}{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} \underbrace{\left(f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x})\right)\left(1-f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x})\right)}_{\textit{Closeness to decision boundary}} \times \left\|\nabla_{\boldsymbol{\theta}} f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x})\right\|^2 + \underbrace{\left|f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x})-y\right|}_{\textit{Error}} \times \lambda\left(\nabla_{\boldsymbol{\theta}}^2 f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x})\right). \tag{5.6}$$

The proof relies on derivations of the Hessian associated with model loss function and Weyl inequality. In other words, Theorem 5.2 highlights a direct connection between the maximum eigenvalue of the group Hessian and **(1)** the closeness to the decision boundary of the group samples, and **(2)** the accuracy of the group. The distance to the decision boundary is derived from [29]. Intuitively this term is maximized when the classifier is highly uncertain about the prediction: $f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x}) \to 0.5$, and minimized when it is highly certain $f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x}) \to 0$ or 1, as showed in the following proposition.

Proposition 5.3. *Consider a binary classifier $f_{\boldsymbol{\theta}}(\boldsymbol{x})$. For a given sample $\boldsymbol{x} \in D$, the term $f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x})(1-f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x}))$ is maximized when $f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x}) = 0.5$ and minimized when $f_{\hat{\boldsymbol{\theta}}}^\star(\boldsymbol{x}) \in \{0,1\}$.*

Observe that a group consisting of samples that are far from the decision boundary will have smaller Hessians and, thus, be less subject to a drop in accuracy due to model pruning. These results can be appreciated in Figure 5.5. Notice the inverse relationship between maximum eigenvalues of the groups' Hessians and their average distance to the decision boundary. The same relation also holds for accuracy: the higher the Hessians maximum eigenvalues, the smaller the accuracy. This is intuitive as samples which are close to the decision boundary will be more prone to errors due to small changes in the model due to pruning, when compared with samples lying far from the decision boundary.

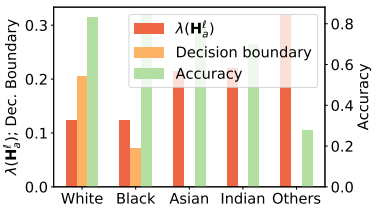


Fig. 5.5: Group Hessians, distance to decision boundary, and accuracy.

Correlation between group Hessians and gradient norms. This section observes a positive correlation between maximum eigenvalues of the Hessian of a group and their gradient norms.

This relation can be appreciated in Figure 5.6. While mainly empirical, this observation is important as it illustrates that both the Hessian $\lambda(\boldsymbol{H}_a^\ell)$ and the gradient $\|\boldsymbol{g}_a^\ell\|$ terms appearing in the upper bound of the excessive loss $R(a)$ reported in Theorem 5.1 are in agreement. This relation was observed in all our experiments and settings. Such observation allows us to infer that it is the combined effect of gradient norms and group Hessians that is responsible for the excessive loss of a group and, in turn, for the exacerbation of unfairness in the pruned models.

The role of the group Hessian in pruning. Having highlighted the connection between Hessian for a group with the resulting accuracy of the model on such a group, this section provides theoretical intuitions on the role of the Hessians in the disparate group losses during pruning.

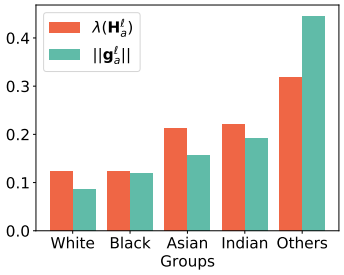


Fig. 5.6: Group Hessians and gradient norms.

In Theorem 5.1, notice that the excessive loss is controlled by term $\|\boldsymbol{H}_a^\ell\| \times \|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|^2$. As also noted in the previous section, the term $\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\|$ regulates the impact of pruning on the excessive loss as the difference between the pruned and non-pruned parameter vectors directly depends on the pruning rate. Similar to the observation for gradient norms, with a fixed pruning rate, groups with different Hessians will have a disparate effect on the resulting term. In particular, groups with small Hessians eigenvalues (those generally distant from the decision boundary and highly accurate) will be less sensitive to the effects of the pruning rate. Conversely, groups with large Hessians eigenvalues will be affected by the pruning rate to a greater extent, *typically* resulting in larger excessive losses. These observations can further be appreciated empirically in Figures 5.4a (for accuracy) and 5.4c (for maximum group Hessian eigenvalues) on the UTKFace datasets for a variety of pruning rates.

5.6 Mitigation solution and evaluation

The previous sections highlighted the presence of two key factors playing a role in the observed model accuracy disparities due to pruning: the difference in gradient norms, and the difference in Hessians losses across groups. This section first shows how to leverage these findings to provide a simple, yet effective solution to reduce the disparate impacts of pruning. Then, the section illustrates the benefits of this mitigating solution on a variety of tasks, datasets, and network architectures.

5.6.1 Mitigation solution

To achieve fairness, the aforementioned findings suggest to equalize the disparity associated with gradient norms $\|\boldsymbol{g}_a^\ell\|$ and Hessians $\lambda(\boldsymbol{H}_a^\ell)$ across different groups $a \in \mathcal{A}$. For this goal, we adopt a constrained empirical risk minimization approach:

$$\underset{\boldsymbol{\theta}}{\text{minimize}} \;\; J(\boldsymbol{\theta}; D) \quad \text{such that:} \;\; \|\boldsymbol{g}_a^\ell\| = \|\boldsymbol{g}^\ell\|, \;\; \lambda(\boldsymbol{H}_a^\ell) = \lambda(\boldsymbol{H}^\ell) \;\; \forall a \in \mathcal{A}, \tag{5.7}$$

where $\boldsymbol{g}^\ell = \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}; D)$ and $\boldsymbol{H}^\ell = \nabla^2_{\boldsymbol{\theta}} J(\boldsymbol{\theta}; D)$ refer to the gradients and Hessian associated with loss function ℓ, respectively, and are computed using the whole dataset D. The approach (5.7) is a common strategy adopted in fair learning tasks, and the chapter uses the Lagrangian Dual method of [44] which exploits Lagrangian duality to extend the loss function with trainable and weighted regularization terms that encapsulate constraints violations (see Section 5.8.2 for additional details).

A shortcoming of this approach is, however, that requires computing the gradient norms and Hessian matrices of the group losses in each and every training iteration, rendering the process computationally unviable, especially for deep, overparametrized networks.

To overcome this computational burden, we will use two observations made earlier in the chapter. First, recall the strong relation between gradient norms for a group and their associated losses. This aspect was noted in Proposition 5.2. That is, when the losses across the groups are similar,

the gradient norms across such groups will also tend to be similar. Next, Theorem 5.2 noted a positive correlation between model errors (and thus loss values) for a group and its associated Hessian eigenvalues. Thus, when the losses across the groups are similar, the group Hessians will also tend to be similar. This intuition is also complemented by the strong correlation between group Hessians and gradient norms reported in Section 5.5. Based on the above observations, we propose a simpler version of the constrained minimizer (5.7) defined as

$$\underset{\boldsymbol{\theta}}{\text{minimize}} \;\; J(\boldsymbol{\theta}; D) \quad \text{such that:} \;\; J(\boldsymbol{\theta}; D_a) = J(\boldsymbol{\theta}; D) \;\; \forall a \in \mathcal{A}, \tag{5.8}$$

that substitutes the gradient norms and max eigenvalues of group Hessians equality constraints with proxy terms capturing the group $J(\boldsymbol{\theta}; D_a)$ and population $J(\boldsymbol{\theta}; D)$ losses.

The impact of such proxy terms in the fairness-constrained problem above can be appreciated, empirically, in Figure 5.7. The plots, that use the UTK-Face dataset, with Ethnicity as protected group, show an original unfair model (top) and a fair counterpart obtained through problem (5.8) (bottom). Both top and bottom sub-figures use an unpruned model. The top sub-figure shows the performance of an original unpruned model trained by minimizing the empirical risk function while the bottom one shows the effect of solving Problem (5.8), i.e., it constrains the empirical risk function with the various group loss terms. Notice how enforcing balance in the group losses also helps reducing and balancing the gradient

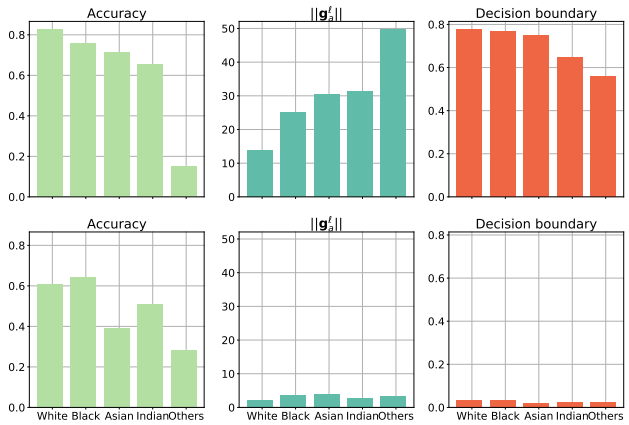


Fig. 5.7: Effects of fairness constraints in balancing not only group accuracy (left) but also gradient norms (middle) and group average distance to the decision boundary (right).

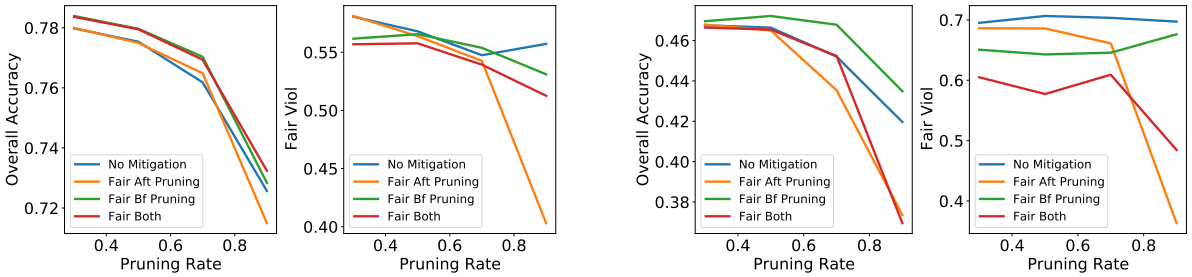


Fig. 5.8: Accuracy and Fairness violations attained by all models on ResNet50, UTK-Face dataset with *ethnicity* (5 classes) as group attribute (and labels) [left] and *age* (9 classes) [right].

norms and group's average distance to the de-

cision boundary. As a consequence, the resulting model fairness is dramatically enhanced (bottom-left subplot).

5.6.2 Assessment of the mitigation solution

Datasets, models, and settings. This section analyzes the results obtained using the proposed mitigation solution with ResNet50 and VGG19 on the UTKFace dataset [144], CIFAR-10 [70], and SVHN [90] for various protected attributes. The experiments compare the following four models:

• *No Mitigation*: it refers to the standard pruning approach which uses no fairness mitigation strategy.

• *Fair Bf Pruning*: it applies the fairness mitigation process (Problem (5.8)) exclusively to the original large network, thus *before* pruning.

• *Fair Aft Pruning*: it applies the mitigation exclusively to the pruned network, thus *after* pruning.

• *Fair Both*: it applies the mitigation both to the original large network and to the pruned network.

The experiments report the overall accuracy of resulting models as well as their fairness violations, defined here as the difference between the maximal and minimal group accuracy. The reported metrics are the average of 10 repetitions. Additional details on datasets, architectures, and hyper-parameters adopted, as well as additional and extended results are reported in Section 5.8.2.

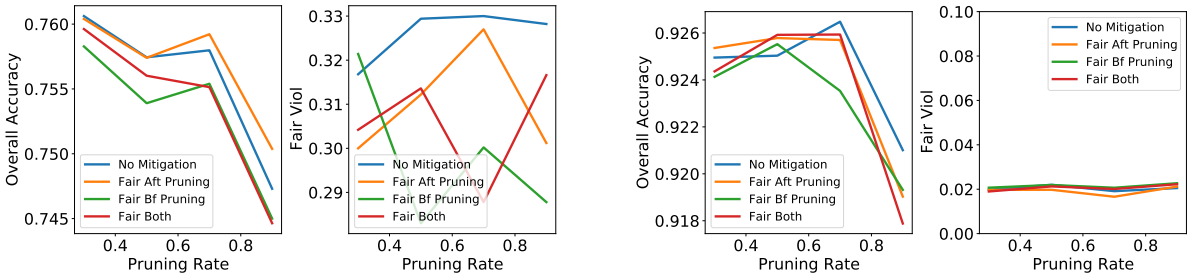


Fig. 5.9: Accuracy and Fairness violations attained by all models on VGG-19, CIFAR-10 dataset (left) and SVHN (right) with 10 class labels also used as group attribute.

Effects on accuracy. The section first focuses on analyzing the effects of accuracy drop due to applying the proposed mitigation solution for fair pruning. Figure 5.8 compares the four models on the UTK-Face dataset using a ResNet50 architecture. The left subplots use *ethnicity* as protected group and class label, with $|\mathcal{Y}| = 5$, while the right subplots use *age* as protected group and class label, with $|\mathcal{Y}| = 9$. Notice that, as expected, all compared models present some accuracy deterioration as the pruning rate increases. However, notably, the deterioration of the models that apply the fair mitigation steps are comparable to (or even improved) those of the "*No mitigation*" model, which applies standard pruning.

A similar trend can be seen in Figure 5.9 that reports results on CIFAR (left) and SVHN (right). Both use the ten class labels as protected attributes. These results clearly illustrate the ability of the mitigating solution to preserve highly accurate models.

A comparison of the “full” (Equation 5.7) and “relaxed” (Equation 5.8) versions of the proposed mitigation solutions is provided in Table 5.1. We note that while the "full" version leads to fairer results, the reduction in the various groups accuracy is often insubstantial. We also note that the running time of the "full" version is largely (over an order magnitude) longer than the relaxed counterpart. This is due to the calculation of gradient norms and the Hessian terms associated with each group.

Effects on fairness. The section next illustrates the ability of the proposed solution to achieve fair pruned models. Table 5.2 illustrates the results for the UTKFaces dataset with ethnicity as class labels and age as protected attributes for a CNN with two convolutional layers and three

Dataset	version	Class-wise accuracy	Overall accuracy
UTK-age bins	**full**	0.856, 0.128, 0.145, 0.319, 0.331, 0.342, 0.181, 0.334, 0.512	0.395
	relaxed	0.810, 0.096, 0.141, 0.284, 0.385, 0.324, 0.227, 0.257, 0.533	0.390
UTK-gender	**full**	0.830, 0.876	0.857
	relaxed	0.868, 0.845	0.852
SVHN	**full**	0.864, 0.911, 0.869, 0.819, 0.887, 0.784, 0.840, 0.877, 0.805, 0.856	0.857
	relaxed	0.824, 0.910, 0.775, 0.726, 0.827, 0.752, 0.747, 0.789, 0.713, 0.755	0.795
MNIST	**full**	0.998, 0.996, 0.993, 0.998, 0.994, 0.991, 0.991, 0.993, 0.992, 0.985	0.993
	relaxed	0.994, 0.988, 0.989, 0.986, 0.987, 0.979, 0.981, 0.988, 0.969, 0.994	0.986

Table 5.1: Full (Equation 5.7) vs relaxed (Equation 5.8) versions of the proposed mitigation solutions.

Methods	Overall accuracy				Fairness violations			
	30%	50%	70%	90 %	30%	50%	70%	90%
No mitigation	0.546	0.545	0.529	0.559	0.179	0.186	0.152	0.134
Fair bf Pruning	0.539	0.557	0.529	0.540	0.189	0.190	0.174	0.238
Fair Aft Pruning	0.538	0.532	0.497	0.472	0.172	0.161	0.163	0.05
Fair both	0.525	0.541	0.508	0.484	0.170	0.144	0.156	0.073

Table 5.2: Accuracy and fairness violations for the UTKFaces dataset with *ethnicity* as class labels and *age* as protected attributes and prune amounts of 30%, 50%, 70%, and 90%.

linear layers and prune amounts: 30%, 50%, 70%, and 90%. Notice how Fair Aft pruning and Fair both achieve relatively lesser fairness violations compared to the No mitigation and the Fair bf Pruning methods in most cases.

Next, the second and fourth subplots presented in Figures 5.8 and 5.9 illustrate the fairness violations obtained by the four models analyzed on different datasets and settings. We make the following observations: First, all the plots exhibit a consistent trend in that the mitigation solution produces models which improve the fairness of the baseline, "*No mitigation*" model. Observe that, as already illustrated in Figure 5.7, the fair models tend to equalize the gradient norms and group Hessians components (and thus the distance to the decision boundary across groups). Thus, the resulting pruned models also attain better fairness, when compared to their standard counterparts.

Next, notice that "*Fair Aft Pruning*" often achieves better fairness violations than "*Fair Bf Pruning*", especially at high pruning regimes. This is because the former has the advantage to apply the mitigation solution directly to the pruned model to ensure that the resulting model has low differences in gradient norms and group Hessians. The presentation also illustrates the application of the mitigation strategies both before and after pruning (*Fair Both*) which shows once again the

significance of applying the mitigation solution over the pruned network.

Finally, it is notable that "*Fair Aft Pruning*" achieves good reductions in fairness violation. Indeed, pre-trained large (non-pruned) fair models may not be available and the ability to retrain these large models prior to pruning may be hindered by their size and complexity.

5.7 Discussion and limitations

This section discusses three key messages found in this study. First, we notice that pruning affecting model separability and distance to the decision boundary is related to concepts also explored in robust machine learning [52, 94]. Not surprisingly, some recent literature in network pruning has empirically observed that pruning may have a negative impact on adversarial robustness [53]. These observations raise questions about the connection between pruning, robustness, and fairness, which we believe is an important direction to further investigate.

Next, although the solution proposed in Problem (5.8) allows it to be adopted in large models, the size of modern ML models (together with the amount of hyperparameters searches) may hinder retraining such original massive models from incorporating fairness constraints. Notably, however, the proposed mitigation solution can be used as a post-processing step to be applied during the pruning operation directly. The previous section shows that the proposed method delivers desirable performance in terms of both accuracy and fairness.

Finally, we notice that the results analyzed in this chapter pertain to losses that are twice differentiable. Lifting such an assumption will be an interesting and challenging future research avenue.

5.8 Appendix

5.8.1 Missing proofs

Theorem 5.1. *The* excessive loss *of a group $a \in \mathcal{A}$ is upper bounded by*[2]*:*

$$R(a) \leq \left\|\boldsymbol{g}_a^\ell\right\| \times \left\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right\| + \frac{1}{2}\lambda\left(\boldsymbol{H}_a^\ell\right) \times \left\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right\|^2 + \mathcal{O}\left(\left\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right\|^3\right), \tag{5.9}$$

where $\boldsymbol{g}_a^\ell = \nabla_{\overset{\star}{\boldsymbol{\theta}}} J(\overset{\star}{\boldsymbol{\theta}}; D_a)$ is the vector of gradients associated with the loss function ℓ evaluated at $\overset{\star}{\boldsymbol{\theta}}$ and computed using group data D_a, $\boldsymbol{H}_a^\ell = \nabla^2_{\overset{\star}{\boldsymbol{\theta}}} J(\overset{\star}{\boldsymbol{\theta}}; D_a)$ is the Hessian matrix of the loss function ℓ, at the optimal parameters vector $\overset{\star}{\boldsymbol{\theta}}$, computed using the group data D_a (henceforth simply referred to as group hessian*), and $\lambda(\Sigma)$ is the maximum eigenvalue of a matrix Σ.*

Proof. Using a second order Taylor expansion around $\overset{\star}{\boldsymbol{\theta}}$, the excessive loss $R(a)$ for a group $a \in \mathcal{A}$ can be stated as:

$$\begin{aligned}
R(a) &= J(\bar{\boldsymbol{\theta}}; D_a) - J(\overset{\star}{\boldsymbol{\theta}}; D_a) \\
&= \left[J\left(\overset{\star}{\boldsymbol{\theta}}; D_a\right) + \left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right)^\top \nabla_\theta J\left(\overset{\star}{\boldsymbol{\theta}}; D_a\right) + \frac{1}{2}\left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right)^\top \boldsymbol{H}_a^\ell \left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right) + \mathcal{O}\left(\left\|\overset{\star}{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}\right\|^3\right)\right] - J\left(\overset{\star}{\boldsymbol{\theta}}; D_a\right) \\
&= \left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right)^\top \boldsymbol{g}_a^\ell + \frac{1}{2}\left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right)^\top \boldsymbol{H}_a^\ell \left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right) + \mathcal{O}\left(\| \overset{\star}{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}\|^3\right)
\end{aligned}$$

The above, follows from the loss $\ell(\cdot)$ being at least twice differentiable, by assumption.

By Cauchy-Schwarz inequality, it follows that

$$\left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right)^\top \boldsymbol{g}_a^\ell \leq \left\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right\| \times \left\|\boldsymbol{g}_a^\ell\right\|.$$

In addition, due to the property of Rayleigh quotient we have:

$$\frac{1}{2}\left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right)^\top \boldsymbol{H}_a^\ell \left(\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right) \leq \frac{1}{2}\lambda\left(\boldsymbol{H}_a^\ell\right) \times \left\|\bar{\boldsymbol{\theta}} - \overset{\star}{\boldsymbol{\theta}}\right\|^2.$$

The upper bound for the excessive loss $R(a)$ is thus obtained by combining these two inequalities.

[2]With a slight abuse of notation, the results refer to $\bar{\boldsymbol{\theta}}$ as the homonymous vector which is extended with $k - \bar{k}$ zeros.

□

Proposition 5.1. *Consider two groups a and b in $\mathcal{A}$ with $|D_a| \geq |D_b|$. Then $\left\|\boldsymbol{g}_a^\ell\right\| \leq \left\|\boldsymbol{g}_b^\ell\right\|$.*

Proof. By the assumption that the model converges to a local minima, it follows that:

$$\begin{aligned}\nabla_\theta \mathcal{L}(\overset{\star}{\boldsymbol{\theta}}; D) &= \sum_{a \in \mathcal{A}} \frac{|D_a|}{|D|} \nabla_\theta J(\overset{\star}{\boldsymbol{\theta}}; D_a) \\ &= \frac{|D_a|}{|D|} \boldsymbol{g}_a^\ell + \frac{|D_b|}{|D|} \boldsymbol{g}_b^\ell = \mathbf{0}\end{aligned}$$

Thus, $\boldsymbol{g}_a^\ell = -\frac{|D_b|}{|D_a|} \boldsymbol{g}_b$. Hence $\|\boldsymbol{g}_a^\ell\| = \frac{|D_b|}{|D_a|} \|\boldsymbol{g}_b^\ell\| \leq \|\boldsymbol{g}_b^\ell\|$, because $|D_a| \geq |D_b|$. □

Proposition 5.2. *For a given group $a \in \mathcal{A}$, gradient norms can be upper bounded as:*

$$\|\boldsymbol{g}_a^\ell\| \in \mathcal{O}\left(\sum_{(\boldsymbol{x}, y) \in D_a} \underbrace{\|f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}) - y\|}_{\textit{Accuracy}} \times \left\|\nabla_{\overset{\star}{\boldsymbol{\theta}}} f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x})\right\| \right).$$

The above proposition is presented in the context of cross entropy loss or mean squared error loss functions. These two cases are reviewed as follows

Cross Entropy Loss. Consider a classification task with cross entropy loss: $\ell(f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y) = -\sum_{z \in \mathcal{Y}} f_{\overset{\star}{\boldsymbol{\theta}}}^z(\boldsymbol{x}) \boldsymbol{y}^z$, where $f_{\overset{\star}{\boldsymbol{\theta}}}^z(\boldsymbol{x})$ represents the z-th element of the output associated with the softmax layer of model $f_{\overset{\star}{\boldsymbol{\theta}}}$, and $\boldsymbol{y}$ is a one-hot encoding of the true label y, with $\boldsymbol{y}^z$ representing its z-th element, then,

$$\begin{aligned}\|\boldsymbol{g}_a\| = \left\|\nabla_{\boldsymbol{\theta}} J(\overset{\star}{\boldsymbol{\theta}}; D_a,)\right\| &= \left\| {}^1\!/_{|D_a|} \sum_{(\boldsymbol{x}, y) \in D_a} \nabla_f \ell(f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}), y) \times \nabla_{\boldsymbol{\theta}} f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}) \right\| \\ &= \left\| {}^1\!/_{|D_a|} \sum_{(\boldsymbol{x}, y) \in D_a} (f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}) - \boldsymbol{y}) \times \nabla_{\boldsymbol{\theta}} f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}) \right\| \\ &\leq {}^1\!/_{|D_a|} \sum_{(\boldsymbol{x}, y) \in D_a} \left\|f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x}) - \boldsymbol{y}\right\| \times \left\|\nabla_{\boldsymbol{\theta}} f_{\overset{\star}{\boldsymbol{\theta}}}(\boldsymbol{x})\right\|,\end{aligned}$$

where the third equality is due to that the gradient of the cross entropy loss reduces to $f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}) - \boldsymbol{y}$.

Mean Squared Error. Next, consider a regression task with mean squared error loss $\ell(f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}), y) = (f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}) - y)^2$. Using the same notation as that made above, if follows:

$$\begin{aligned}
\|\boldsymbol{g}_a\| = \left\|\nabla_{\boldsymbol{\theta}} J(\overset{\star}{\boldsymbol{\theta}}; D_a,)\right\| &= \left\| {}^{1}\!/_{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} \nabla_f \ell(f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}), y) \times \nabla_{\boldsymbol{\theta}} f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}) \right\| \\
&= \left\| {}^{2}\!/_{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} (f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}) - \boldsymbol{y}) \times \nabla_{\boldsymbol{\theta}} f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}) \right\| \\
&\leq {}^{2}\!/_{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} \left\| f_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x}) - \boldsymbol{y} \right\| \times \left\| \nabla_{\boldsymbol{\theta}} f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}) \right\|,
\end{aligned}$$

where the third equality is due to that the gradient of the mean squared error loss w.r.t. $f_{\hat{\boldsymbol{\theta}}}^{\star}(\cdot)$ reduces to $2(f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}) - \boldsymbol{y})$.

Theorem 5.2. *Let f_θ be a binary classifier trained using a binary cross entropy loss. For any group $a \in \mathcal{A}$, the maximum eigenvalue of the group Hessian $\lambda(\boldsymbol{H}_a^{\ell})$ can be upper bounded by:*

$$\lambda(\boldsymbol{H}_a^{\ell}) \leq \frac{1}{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} \underbrace{\left(f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x})\right)\left(1 - f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x})\right)}_{\textit{Distance to decision boundary}} \times \left\|\nabla_{\boldsymbol{\theta}} f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x})\right\|^2 + \underbrace{\left|f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x}) - y\right|}_{\textit{Accuracy}} \times \lambda\left(\nabla_{\boldsymbol{\theta}}^2 f_{\hat{\boldsymbol{\theta}}}^{\star}(\boldsymbol{x})\right). \tag{5.10}$$

Proof. First notice that an upper bound for the Hessian loss computed on a group $a \in \mathcal{A}$ can be derived as:

$$\lambda(\boldsymbol{H}_a^{\ell}) = \lambda\left(\frac{1}{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} \boldsymbol{H}_{\boldsymbol{x}}^{\ell}\right) \leq \frac{1}{|D_a|} \sum_{(\boldsymbol{x},y)\in D_a} \lambda\left(\boldsymbol{H}_{\boldsymbol{x}}^{\ell}\right) \tag{5.11}$$

where $\boldsymbol{H}_{\boldsymbol{x}}^{\ell}$ represents the Hessian loss associated with a sample $\boldsymbol{x} \in D_a$ from group a. The above follows Weily's inequality which states that for any two symmetric matrices A and B, $\lambda(A+B) \leq \lambda(A) + \lambda(B)$.

Next, we will derive an upper bound on the Hessian loss associated to a sample $\boldsymbol{x}$. First, based

on the chain rule a closed form expression for the Hessian loss associated to a sample $\boldsymbol{x}$ can be written as follows:

$$\boldsymbol{H}_{\boldsymbol{x}}^{\ell} = \nabla_f^2 \ell\left(f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}), y\right)\left[\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\left(\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right)^{\top}\right] + \nabla_f \ell\left(f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}), y\right)\nabla_{\boldsymbol{\theta}}^2 f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}). \tag{5.12}$$

The next follows from that

$$\begin{aligned}\nabla_f \ell\left(f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}), y\right) &= \left(f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) - y\right),\\ \nabla_f^2 \ell\left(f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}), y\right) &= f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\left(1 - f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right).\end{aligned}$$

Applying the Weily inequality again on the R.H.S. of Equation 5.12, we obtain:

$$\begin{aligned}\lambda(\boldsymbol{H}_{\boldsymbol{x}}^{\ell}) &\leq f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\left(1 - f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right) \times \left\|\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right\|^2 + \lambda\left(f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) - y\right) \times \nabla_{\boldsymbol{\theta}}^2 f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\\ &\leq f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\left(1 - f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right) \times \left\|\nabla_{\boldsymbol{\theta}} f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right\|^2 + \left|f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) - y\right| \lambda\left(\nabla_{\boldsymbol{\theta}}^2 f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right)\end{aligned} \tag{5.13}$$

The statement of Theorem 5.2 is obtained combining Equations 5.13 with 5.11. □

Proposition 5.3. *Consider a binary classifier $f_{\boldsymbol{\theta}}(\boldsymbol{x})$. For a given sample $\boldsymbol{x} \in D$, the term $f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})(1 - f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}))$ is maximized when $f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) = 0.5$ and minimized when $f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) \in \{0, 1\}$.*

Proof. First, notice that $f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) \in [0, 1]$, as it represents the soft prediction (that returned by the last layer of the network), thus $f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) \geq f_{\boldsymbol{\theta}}^{\star 2}(\boldsymbol{x})$. It follows that:

$$f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\left(1 - f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right) = f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) - f_{\boldsymbol{\theta}}^{\star 2}(\boldsymbol{x}) \geq 0. \tag{5.14}$$

In the above, it is easy to observe that the equality holds when either $f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) = 0$ or $f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) = 1$.

Next, by the Jensen inequality, it follows that:

$$f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\left(1 - f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right) \leq \frac{\left(f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x}) + 1 - f_{\boldsymbol{\theta}}^{\star}(\boldsymbol{x})\right)^2}{4} = \frac{1}{4}. \tag{5.15}$$

The above holds when $f^{\star}_{\boldsymbol{\theta}}(\boldsymbol{x}) = 1 - f^{\star}_{\boldsymbol{\theta}}(\boldsymbol{x})$, in other words, when $f^{\star}_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0.5$. Notice that, in the case of binary classifier, this refers to the case when the sample $\boldsymbol{x}$ lies on the decision boundary. □

5.8.2 Dataset and experimental settings

5.8.3 Datasets

The chapter uses the following datasets to validate the findings discussed in the main chapter:

- **UTK-Face** [144]. A large-scale face dataset with a long age span (range from 0 to 116 years old). The dataset consists of over 20,000 face images with annotations of age, gender, and ethnicity. The images cover large variations in pose, facial expression, illumination, occlusion, resolution, etc. The experiments adopt the following attributes for classification (e.g., $\mathcal{Y}$) and as protected group ($\mathcal{A}$): *ethnicity*, *age bins*, *gender*.

- **CIFAR-10** [70]. This dataset consists of 60,000 32×32 RGB images in 10 classes, with 6,000 images per class. The 10 different classes represent airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks.

- **SVHN** [90] Street View House Numbers (SVHN) is a digit classification dataset that contains 600,000 32×32 RGB images of printed digits (from 0 to 9) cropped from pictures of house number plates.

5.8.4 Architectures, hyper-parameters, and settings

The study adopts the following architectures to validate the results of the main chapter:

- **ResNet18**: An 18-layer architecture, with 8 residual blocks. Each residual block consists of two convolution layers. The model has ~ 11 million trainable parameters.

- **ResNet50** This model contains 48 convolution layers, 1 MaxPool layer and a AvgPool layer. ResNet50 has ~ 25 million trainable parameters.

- **VGG-19** This model consists of 19 layers (16 convolution layers, 3 fully connected layers, 5 MaxPool layers and 1 SoftMax layer). The model has ~ 143 million parameters.

Hyperparameters for each of the above models was performed over a grid search (for different learning rates = $[0.0001, 0.001, 0.01, 0.1, 0.5, 0.05, 0.005, 0.0005]$) over a cluster of NVIDIA RTX A6000 with the above networks using the UTKFace dataset. The models with the highest accuracy were chosen and employed for the assessment of the mitigation solution in Sec. 5.6.2. The running time required for all sets of experiments which include mitigation solutions was about ~3 *days*.

The training uses the SGD optimizer with a momentum of 0.9 and weight_decay of $1e^{-4}$. Finally, the Lagrangian step size adopted in the Lagrangian dual learning framework [44] is set to 0.001.

All the models developed were implemented using Pytorch 3.0. The training was performed using NVidia Tesla P100-PCIE-16GB GPUs and 2GHz Intel Cores. The model was run for 100 epochs for the CIFAR-10 and SVHN and 40 epochs for UTK-Face dataset. Each reported experiment is an average of 10 repetitions. In all experiments, the protected group set coincides with the target label set: i.e., $\mathcal{A} = \mathcal{Y}$.

5.8.5 Impact of pruning on fairness

This section shows and affirms the impact of pruning towards accuracy disparity through VGG-19 network. The same training procedures as employed with ResNet18 in Fig 7.1 were followed. Each demographic group's accuracy is shown before and after pruning on the UTK-Face dataset in two cases: when *ethnicity* is a group attribute as in Figure 5.10, and when *gender* is a group attribute as in Figure 5.11. A consistent message is that under a higher pruning rate, the accuracies are more imbalanced across different groups, emphasizing the negative impact of pruning on fairness.

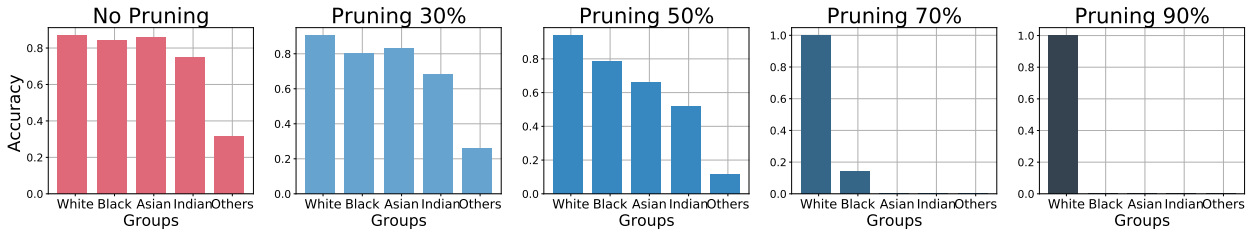


Fig. 5.10: Accuracy of each demographic group in the UTK-Face dataset with ethnicity (5 classes) as group attribute using VGG19 over increasing pruning rates.

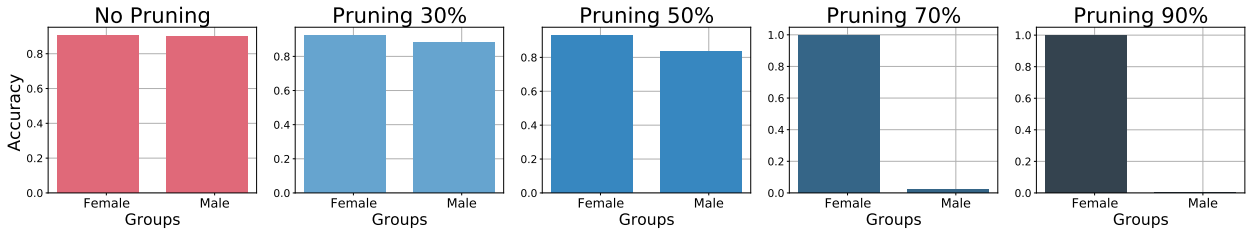


Fig. 5.11: Accuracy of each demographic group in the UTK-Face dataset with gender (2 classes) as group attribute using VGG19 over increasing pruning rates.

5.8.6 Correlation of gradient/hessian norm and average distance to the decision boundary

This subsection elaborates the impact of gradient norms and group Hessians towards the fairness issues shown in Figures 5.10 and 5.11. In Section 5.4, it has been shown that the group with a larger gradient norm before pruning will be penalized more than the groups with a smaller gradient norm. Figures 5.13 and 5.12 show the gradient norm of each demographic group for UTK-Face dataset under two choices of protected attributes for VGG 19 networks. The results indicate that a group penalized less will have a smaller gradient norm with respect to those of the other groups.

In addition, Section 5.5 supports that Hessian norm is another factor. More precisely, the groups with a larger Hessian norm will be penalized more (drop much more in accuracy) than groups with a smaller Hessian norm. Evidence is provided for the claim on VGG19 in Figures 5.12 and 5.13. These results on VGG19 again confirm the theoretical findings.

Finally, in Section 5.5, a positive correlation between gradient norms and Hessian groups is shown in Theorem 5.2, and a negative correlation between Hessian groups and distance to the decision boundary is shown in Proposition 5.3. These important results again are supported by the

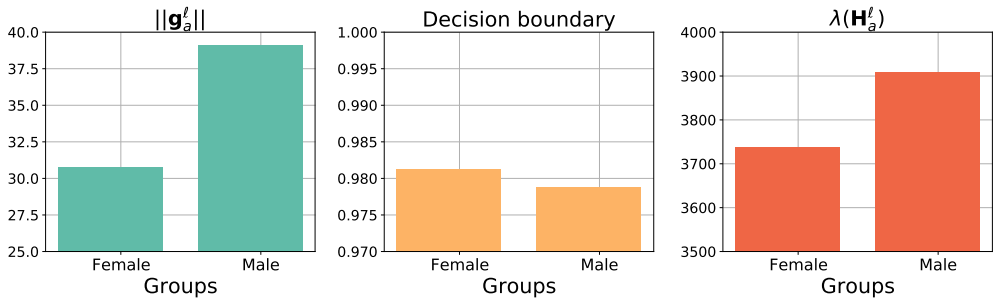


Fig. 5.12: Gradient/Hessian norm and average distance to the decision boundary of each demographic group in the UTK-Face dataset with gender (2 classes) as group attribute using VGG19 with no pruning.

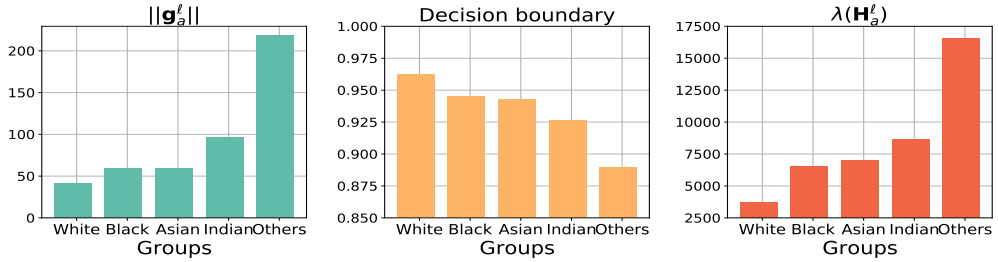


Fig. 5.13: Gradient/Hessian norm and average distance to the decision boundary of each demographic group in the UTK-Face dataset with ethnicity (5 classes) as group attribute using VGG19 with no pruning.

results in Figures 5.12 and 5.13.

5.8.7 Impact of group sizes to gradient norm

This section presents additional empirical results to support Theorem 5.1, stating that the group with more samples will tend to have a smaller gradient norm. In these experiments, run on a ResNet50 network, one group is chosen and upsampled $1\times$, $5\times$, $10\times$, and $20\times$ times. Note that by increasingly upsampling it, the group becomes the majority group in that dataset. A group with *more samples* is expected to end up with a *smaller gradient norm* when the training convergences.

UTK-Face with gender Since the UTK-Face is balanced with regard to gender (Female/Male), the number of samples in Female, and Male groups is upsampled in turn. Figure 5.14 reports the respective gradient norms at the last training iteration when upsampling Females (left) and Males

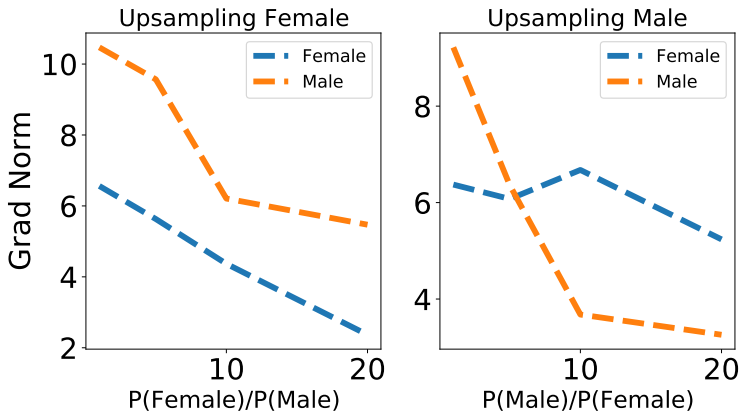


Fig. 5.14: Impact of group sizes to the gradient norm per group in UTK-Face datase where groups are Male and Female.

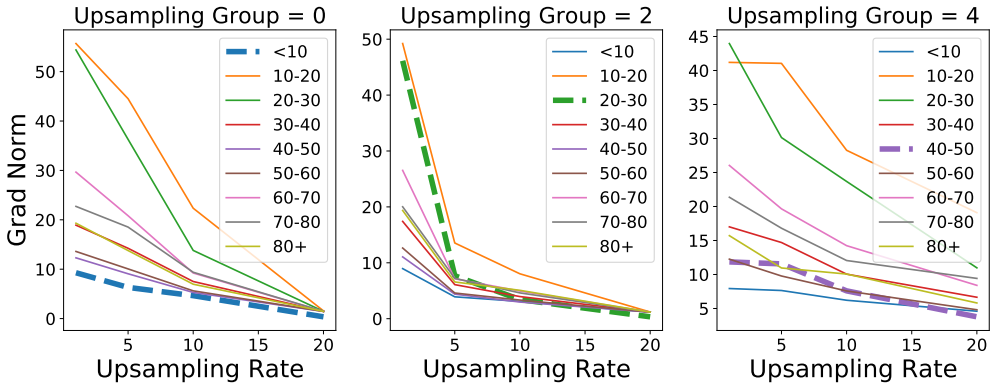


Fig. 5.15: Impact of group sizes to the gradient norm per group in UTK-Face dataset where groups are nine age bins. The group with dotted thick line is a *majority group* in each chart.

(right.) Note how the Male group, initially with no upsampling, has a larger gradient norm than the Female group (right sub-plot). However, if the number of Male samples is increased enough, its gradient norm becomes smaller than that of the Female group.

UTK-Face with age bins Similar experiments are performed with UTK-Face on nine age bin groups. Three age bins are randomly chosen, 0, 2, 4, and the number of samples for each group is upsampled in turn. The gradient norms of nine age bin groups are shown in Figure 5.15, where the upsampled groups are highlighted with dotted thick lines. The results echo that if a group's number of samples is increased enough, its gradient norm at convergence will be smaller than the other 8 age bin groups.

CHAPTER 6
FAIRNESS INCREASES ADVERSARIAL VULNERABILITY

The remarkable performance of deep learning and its applications in consequential domains (e.g., facial recognition) introduces important challenges at the intersection of equity and security. Fairness and robustness are two desired notions often required in learning models. Fairness ensures that models do not disproportionately harm (or benefit) some groups over others, while robustness measures the models' resilience against small input perturbations.

This chapter shows the existence of a dichotomy between fairness and robustness, and analyzes when striving for fairness decreases the model robustness to adversarial samples. The reported analysis sheds light on the factors causing such contrasting behavior, suggesting that distance to the decision boundary across groups as a key factor. Extensive experiments on non-linear models and different architectures validate the theoretical findings in multiple vision domains. Finally, the chapter proposes a simple, yet effective, solution to construct models achieving good tradeoffs between fairness and robustness.

6.1 Introduction

Data-driven learning systems have become instrumental for decision-making in a variety of consequential contests, including assisting in legal decisions [64], lending [118], hiring [110], and providing personalized recommendations [22]. Consequentially, fairness has emerged as a critical requirement for successful adoption and usage of these systems. Various notions of fairness drawing from legal and philosophical doctrine have been proposed to ensure that the models' errors do not affect specific groups [84]. In general, fair models attempt at constraining their hypochapter space so that the errors in reported outcomes are uniformly distributed across different protected groups.

When these fairness constraints are enforced in learning systems, a commonly observed behavior is an overall degradation of the model accuracy. Thus, a growing body of research has been focusing on striking the right balance between fairness and accuracy [105]. This chapter shows that fairness may have another important consequence on the deployed models: *a reduction of the model robustness*. This aspect is important as the vulnerability of deep learning models to adversarial examples hinders their application in many security-sensitive domains. However, these behaviors are currently not fully understood and have not received the attention they deserve given the significant equity and security consequences they have on the final decisions.

This chapter addresses this important gap and shows that enforcing fairness may negatively affect the robustness of a model. Specifically, the chapter (1) it analyzes when and why fairness and robustness may be misaligned in their objectives, (2) it provides an analysis on the relationship between fair, robust, and "natural" (e.g., non-fair non-robust) models, and (3) it identifies *the distance to the decision boundary* as a key aspect linking fairness and robustness. Moreover, (4) the chapter shows how the distance to the decision boundary can explain the increase of adversarial vulnerability of fair models, providing extensive experiments and validation over a variety of vision tasks and architectures, and verifying the presence of the fairness/robustness dichotomy for multiple techniques aimed at achieving fairness and measuring robustness. Finally, (5) building from the reported theoretical observations, the chapter also proposes a simple, yet effective,

strategy to find a good tradeoff between accuracy, fairness, and robustness.

To the best of the authors' knowledge, this is is the first work showing that enforcing fairness may negatively affect robustness. The results show that, without careful considerations, inducing a desired equity property on a learning task may create significant security challenges. *These results should not be read as an endorsement to avoid constructing fairer or safer models; rather it should be understood as a call for additional research at the intersection of fairness and robustness to achieve appropriate tradeoffs.*

6.2 Difference with previous work

The intersection of fairness and robustness has received limited attention thus far, with only a handful of studies examining this area. For instance, Xu et al. [139] recently showed that adversarially robust models can exhibit significant accuracy disparity among different classes, as opposed to their standard counterparts. To address this issue, they proposed a Fair-Robust-Learning (FRL) framework for adversarial defense. Meanwhile, Khani and Liang [66] analyzed the impact of noise in features on disparities in error rates when learning regression models.

In contrast to previous studies such as [89,139], which highlighted how adversarial training can disproportionately harm certain protected groups, our work takes a different approach. Specifically, it demonstrates that enforcing fairness comes at the expense of reduced robustness. As a result, our analysis requires a theoretical approach distinct from those proposed in earlier studies.

6.3 Problem settings

The chapter considers a typical multi-class classification problem, whose input is a dataset D consisting of n data points (X_i, A_i, Y_i), each of which drawn i.i.d. from an unknown distribution Π and where $X_i \in \mathcal{X}$ is a feature vector, $A_i \in \mathcal{A}$ is a protected attribute, and $Y_i \in \mathcal{Y} = [C]$ is a label, with C being the number of possible class labels. For example, consider the case of a classifier to predict the age range of an individual. The features X_i may describe the pixels associated with the

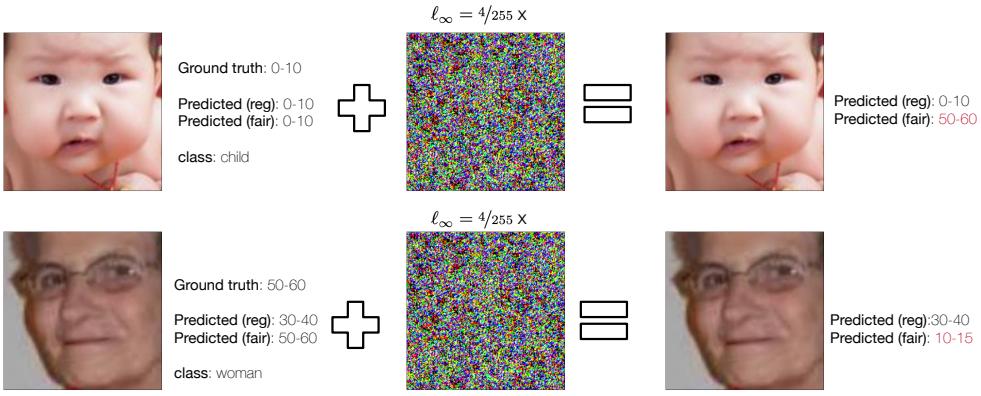


Fig. 6.1: An example of robustness loss in the UTKFace dataset. A regular (reg) and a fair models are trained to predict age group from faces and exposed to adversarial examples generated under an RFGSM [122] attack. The predictions of the regular model do not change under adversarial examples (regardless of their original correctness), while the fair models decision change in the presence of adversarial noise.

individual headshot and their demographics, the protected attribute A_i may describe the individual gender or ethnicity, and Y_i represents the age range. The goal is to learn a classifier $f_\theta : \mathcal{X} \to \mathcal{Y}$, where θ is a vector of real-valued parameters. The model quality is assessed in terms of a non-negative loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$, and the training aims at minimizing the empirical risk function:

$$\overset{\star}{\theta} = \underset{\theta}{\operatorname{argmin}}\ \mathcal{L}_\theta(D) \left(= \frac{1}{n} \sum_{i=1}^{n} \ell(f_\theta(X_i), Y_i) \right) \tag{6.1}$$

For a group $a \in \mathcal{A}$, notation D_a is used to denote the subset of D containing exclusively samples i with $A_i = a$. Importantly, the chapter assumes that the attribute A is not part of the model input during inference. The chapter focuses on learning classifiers that satisfy group fairness (to be defined shortly) and on analyzing the robustness impact of fairness.

6.4 Preliminaries

6.4.1 Fairness and fair learning

This chapter considers a classifier f satisfying accuracy parity [145], a group fairness notion commonly adopted in machine learning requiring model misclassification rates to be conditionally

independent of the protected attribute. That is, $\forall(X, A, Y) \sim \Pi$ and $\forall a \in \mathcal{A}$,

$$|\Pr\left(f_\theta(X) \neq Y \mid A = a\right) - \Pr\left(f_\theta(X) \neq Y\right)| \leq \alpha, \tag{6.2}$$

where α denotes the allowed *fairness violation*. In practice, the above is expressed as a difference of empirical expectations of the group and population misclassification rates. That is, $\forall a \in \mathcal{A}$:

$$\left| {}^{1}\!/_{|D_a|} \sum_{(X,A,Y)\in D_a} \mathbb{1}\{f_\theta(X) \neq Y\} - {}^{1}\!/_{n} \sum_{(X,A,Y)\in D} \mathbb{1}\{f_\theta(X) \neq Y\} \right| \leq \alpha.$$

Several approaches have been proposed in the literature to encourage the satisfaction of accuracy parity. They can be summarized in methods that use penalty terms into the empirical risk loss function to capture the fairness violations, and those which minimize the maximum group loss. The core of the chapter focuses on the first set of methods; the analysis for the second set is presented in Section 6.10.1.

Penalty-based methods. In this category, the model loss function (Equation (6.1)) is augmented with penalty fairness constraint terms [7, 129] as follows:

$$\theta_{\mathrm{f}}(\lambda) = \underset{\theta}{\operatorname{argmin}}\ \mathcal{L}_\theta(D) + \lambda \left(\sum_{a\in\mathcal{A}} |\mathcal{L}_\theta(D_a) - \mathcal{L}_\theta(D)| \right) \tag{6.3}$$

where $\mathcal{L}_\theta(D_a) = {}^{1}\!/_{|D_a|} \sum_{(X,A,Y)\in D_a} \ell(f_\theta(X), Y)$ is the empirical risk loss associated with protected group $a \in \mathcal{A}$. In addition, $\lambda > 0$ is the fairness penalty parameter that enforces a tradeoff between fairness and accuracy.

6.4.2 Robustness and robust learning

This chapter analyzes the effect of enforcing fairness on adversarial robustness, a key property of trustworthy machine-learning systems. In this work, and following robust learning conventions,

the robustness of a model f is measured in terms of the *robust error*:

$$\mathcal{L}_\theta^{\text{rob}}(\epsilon) = \Pr\left(\exists \tau,\ \|\tau\|_p \leq \epsilon, f_\theta(X+\tau) \neq Y\right), \tag{6.4}$$

which measures the sensitivity of the model errors to small input perturbations $\|\tau\|_p \leq \epsilon$ in ℓ_p norms, with p often considered in $\{0, 1, 2, \infty\}$. The robust error can be decomposed into two components [141]:

$$\mathcal{L}_\theta^{\text{rob}}(\epsilon) = \mathcal{L}_\theta^{\text{nat}} + \mathcal{L}_\theta^{\text{bdy}}(\epsilon), \tag{6.5}$$

where the first denotes the *natural error* and the second the *boundary error*. The natural error measures the standard model performance when exposed to *unperturbed* samples (X, A, Y):

$$\mathcal{L}_\theta^{\text{nat}} = \Pr\left(f_\theta(X) \neq Y\right), \tag{6.6}$$

whose empirical version is defined in Equation (6.1) with a 0/1 loss function. The boundary error measures the probability that the model predictions change on *perturbed* samples $(X+\|\tau\|_p, A, Y)$:

$$\begin{aligned}\mathcal{L}_\theta^{\text{bdy}}(\epsilon) = \Pr\big(&\exists \|\tau\|_p \leq \epsilon,\ f_\theta(X+\tau) \neq f_\theta(X), \\ &f_\theta(X) = Y\big).\end{aligned} \tag{6.7}$$

The boundary error implicitly introduces a notion of *decision boundary* and a *distance between an input sample and this decision boundary*. For instance, in linear classifiers, the decision boundary is represented by a hyperplane. The distance of a sample X to the decision boundary for a classifier f_θ can be formalized as

$$\Delta(X, f_\theta) = \max \epsilon \text{ s.t. } f_\theta(X+\tau) = f_\theta(X),\ \forall \|\tau\| \leq \epsilon.$$

Samples close to the decision boundary will be less tolerant to noise than those lying far from it. The analysis in this chapter regarding the impact of fairness on robustness is based on this concept.

In particular, the results show that imposing fairness constraints may reduce the distance to the decision boundary of the samples $(X, A, Y) \sim \Pi$.

6.5 Real-world implications

Prior diving into the analysis, the chapter provides an example showing how robustness errors can be exacerbated when a image classifier is trained to satisfy fairness. Deep neural networks have been used in many real-world applications, including image facial recognition and object detection. When perturbations (either due to noise or by malicious adversaries) are introduced in the model inputs, they may cause harmful effects as they lead the classifier to misclassify targeted inputs.

Figure 6.1 shows an example of inputs from the UTKFace dataset where a classifier is trained to minimize the regular empirical risk loss of equation (6.1) (top) or the fair empirical risk loss of equation (6.3) (bottom). Both inputs are perturbed with the same amount of ℓ_∞ noise, but the fair network is much more brittle than its regular counterpart, inducing errors in the classifier outputs. It is important to note that this chapter uses datasets such as UTKFace (described in detail in Section 6.10.2) only to demonstrate the effects of fairness to robustness. As noted in previous works, the very task of predicting gender, race, or other characteristics from a person face is flawed and raises deep ethical concerns [100].

6.6 Why fairness weakens robustness?

This section presents the main results of the chapter. It will show that fairness affects model robustness because the learned decision boundary is *pulled in opposite directions* by fair and robust models. To render the analysis tractable, the theoretical discussion focuses on linear classifiers, and more specifically on learning a mixture of Gaussians with a linear classifiers. In addition, Section 6.7 will show that a similar phenomenon occurs in large non-linear models. This section assumes that $\mathcal{A} = \mathcal{Y}$, i.e., the protected attribute is also the output of the classifier, again to simplify exposition. All proofs are given in Section 6.10.1.

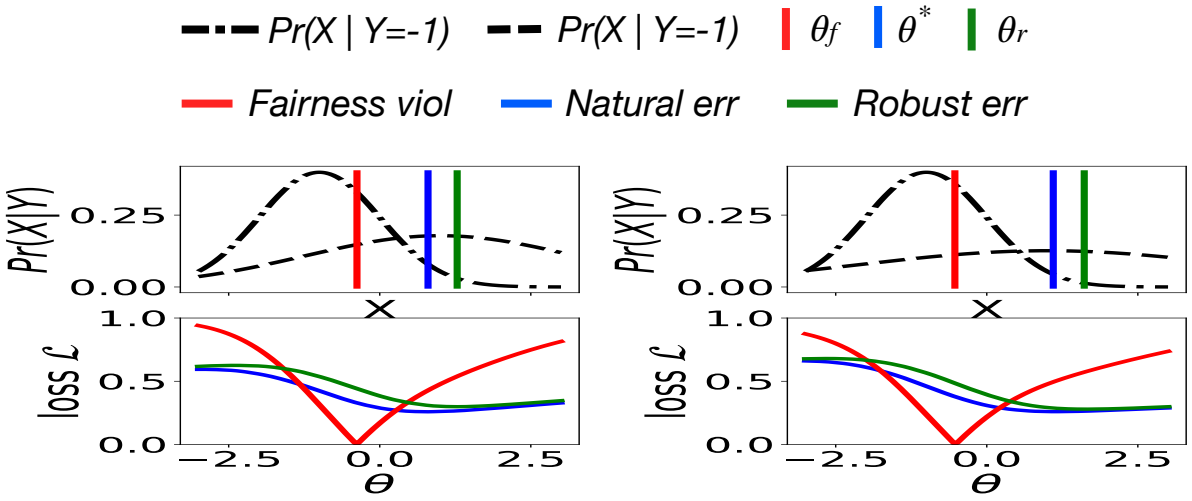


Fig. 6.2: Illustration of optimal natural θ^*, fair θ_f, and robust θ_r classifiers for $K = 5$ (left) and $K = 10$ (right) with $\mu_- = -1$ and $\mu_+ = 1$.

Optimal Models for Mixtures of Gaussians

Consider a binary classification setting (i.e., $\mathcal{Y} = \{-1, 1\}$) with data drawn from a mixture of Gaussian distributions, so that $\Pr(X \mid Y = -1) \propto \mathcal{N}(\mu_-, 1)$ and $\Pr(X \mid Y = 1) \propto \mathcal{N}(\mu_+, K^2)$, with $\mu_- < \mu_+$ and different variances ($K > 1$). The analysis can be easily extended to higher-dimensional cases, but these non-restrictive assumptions help simplifying and clarifying exposition. An illustration of this setting is reported in Figure 6.2 (top) where the data distributions are highlighted with black dashed curves.

The following analysis poses no restrictions on the relative subgroup sizes $|D_1|$ and $|D_{-1}|$ and focuses on the less-restrictive *balanced* data setting, in which data samples from different protected groups are equally likely.

The chapter studies a family of parametric classifiers $\{f_\theta\}_\theta$ with $\theta \in [\mu_-, \mu_+] \subseteq \mathbb{R}$, where $f_\theta(X) = \mathbb{1}\{X > \theta\}$ denotes the classification output of the classifier. The optimal models with respect to the natural, fair, and robust losses can be specified as follows:

• **Optimal natural model** $\left(f_{\theta}^{\star}\right)$**.** It is the Bayes classifier which minimizes the natural classification error as defined in Equation (6.1). In Figure 6.2 (top), this classifier is represented by vertical blue lines.

• **Optimal fair model** $\left(f_{\theta_{\mathrm{f}}}\right)$**.** Intuitively, this classifier is $\theta_{\mathrm{f}}(\infty)$ as defined in Equation (6.3).

Formally speaking, this classifier minimizes a lexicographic function whose first component is $\sum_{a\in\mathcal{A}}|\mathcal{L}_\theta(D_a)-\mathcal{L}_\theta(D))|$ and second component is $\mathcal{L}_\theta(D)$. In Figure 6.2 (top), this classifier is represented by vertical red lines.

• **Optimal robust model** $(f_{\theta_{\mathrm{r}}^{(\epsilon)}})$**.** This classifier minimizes the robust classification error in Equation (6.5), for a given ϵ. In Figure 6.2 (top), it is depicted by vertical green lines.

Relationships Between the Optimal Models

The next result characterizes the positional relationship among the three optimal models mentioned above, which can be observed in Figure 6.2.

Theorem 6.1. *For any* $\epsilon \in \left[0, \frac{\mu_+-\mu_-}{2}\right]$ *and* $K \in (1, B_K]$*, where* $B_K = \min\left\{\exp\left(\frac{(\mu_+-\mu_--2\epsilon)^2}{2}\right), \frac{\mu_+-\mu_-}{\epsilon}-1\right\}$,

$$\mu_- + \epsilon \leq \theta_{\mathrm{f}} \leq \overset{\star}{\theta} \leq \theta_{\mathrm{r}}^{(\epsilon)} \leq \mu_+ - \epsilon\,. \tag{6.8}$$

Besides, $\theta_{\mathrm{r}}^{(\epsilon)}$ *is an increasing function of* ϵ *over* $[0, \frac{\mu_+-\mu_-}{2}]$.

The result follows from the observation that the optimal natural model $f_{\overset{\star}{\theta}}$ can be expressed as

$$\overset{\star}{\theta} = \mu_- - \frac{\mu_+-\mu_-}{K^2-1} + \frac{K}{K^2-1}\sqrt{2(K^2-1)\ln(K)+(\mu_+-\mu_-)^2}\,; \tag{6.9}$$

the fair classifier $f_{\theta_{\mathrm{f}}}$ as:

$$\theta_{\mathrm{f}} = \mu_- + \frac{\mu_+-\mu_-}{K+1}$$

and the robust classifier $f_{\theta_{\mathrm{r}}^{(\epsilon)}}$ as

$$\theta_{\mathrm{r}}^{(\epsilon)} = \mu_- - \frac{\mu_+-\mu_--(K^2+1)\epsilon}{K^2-1} + \frac{K}{K^2-1}\sqrt{2(K^2-1)\ln(K)+(\mu_+-\mu_--2\epsilon)^2}\,. \tag{6.10}$$

We note that [139] derives similar expressions for the optimal natural and robust models, which are used to investigate the natural error gap between the two classes. Despite the similarity in the adopted notation, importantly, Theorem 6.1 provides a unique comparison among the three classi-

fiers analyzed, *highlighting the difficulty in achieving both robustness and fairness simultaneously, as fairness and robustness pull the optimal classifier in opposing directions.*

While [139] mainly focuses on the unfairness resulting from robust training, the remainder of this section examines the cost of adversarial robustness in fair training. Specifically, we measure this reduced robustness cost analyzing robust and boundary errors.

From the relations highlighted in Theorem 6.1, it follows that **(1)** *the fair classifier achieves the largest robust error while the robust classifier results in the least error*, and **(2)** *the fair classifier achieves the largest boundary error while the robust classifier results in the smallest boundary error*, as expressed by the following Corollaries.

Corollary 6.1. *For any* $\epsilon \in \left[0, \frac{\mu_+ - \mu_-}{2}\right]$ *and* $K \in (1, B_K]$,

$$\mathcal{L}^{\text{rob}}_{\theta_{\text{f}}}(\epsilon) \geq \mathcal{L}^{\text{rob}}_{\theta^\star}(\epsilon) \geq \mathcal{L}^{\text{rob}}_{\theta_{\text{r}}^{(\epsilon)}}(\epsilon)\,.$$

Corollary 6.2. *For any* $\epsilon \in \left[0, \frac{\mu_+ - \mu_-}{4}\right]$ *and* $K \in \left(1, \bar{B}_K\right]$,

$$\mathcal{L}^{\text{bdy}}_{\theta_{\text{f}}}(\epsilon) \geq \mathcal{L}^{\text{bdy}}_{\theta^\star}(\epsilon) \geq \mathcal{L}^{\text{bdy}}_{\theta_{\text{r}}^{(\epsilon)}}(\epsilon)\ ,$$

where $\bar{B}_K = \min\left\{\exp\left(\frac{(\mu_+ - \mu_- - 2\epsilon)^2}{2}\right), \phi^{-1}\left(\frac{\mu_+ - \mu_-}{\epsilon} - 2\right)\right\}$ *and* ϕ^{-1} *is the inverse function associated with* $\phi : [1, +\infty) \mapsto [2, +\infty)$ *such that* $\phi(x) = x + 1/x$.

These results further highlight the impossibility of achieving fairness and robustness simultaneously in this classification task. Fairness and robustness are pulling the classifier in opposite directions.

The Role of the Decision Boundary

Building on the previous results, this section provides the key theoretical intuitions to explain why fairness increases adversarial vulnerability. It identifies the average distance to the decision boundary as the central aspect linking fairness and robustness, which is formalized in Theorem

6.2.

Theorem 6.2. *For any* $\epsilon \in \left[0, \frac{\mu_+ - \mu_-}{2}\right]$ *and* $K \in (1, B_K]$*,*

$$\mathbb{E}\left[\Delta\left(X, f_{\theta_{\mathrm{r}}^{(\epsilon)}}\right)\right] \geq \mathbb{E}\left[\Delta\left(X, f_{\theta}^{\star}\right)\right] \geq \mathbb{E}\left[\Delta\left(X, f_{\theta_{\mathrm{f}}}\right)\right].$$

In addition, the fair model minimizes the average distance to its decision boundary over all valid classifiers, i.e.,

$$\theta_{\mathrm{f}} = \underset{\theta \in [\mu_-, \mu_+]}{\operatorname{argmin}} \ \mathbb{E}\left[\Delta\left(X, f_{\theta}\right)\right].$$

Theorem 6.2 indicates that, among the three considered optimal models, the fair model has the smallest average distance to the decision boundary while the robust model has the largest distance. The result above is exemplified in Figure 6.2. The bottom plots show the losses associated with the optimal natural, fair, and robust models for two choices of K (left and right) while the top plots show the optimal decision boundaries associated with each of the three models – notice they correspond to the minima of their relative losses.

Observe that class $Y = 1$ has a higher classification error than class $Y = -1$ under the natural (and thus unfair) classifier $f_{\theta}^{\star}$. This is intuitive since the conditional distribution $\Pr\left(X \mid Y = 1\right)$ has much higher variance than $\Pr\left(X \mid Y = -1\right)$. Hence, to balance the classification errors, the fair classifier pushes the decision boundary towards the mean of class $Y = -1$. This increases the error of class $Y = -1$ while decreasing the error of class $Y = 1$. In contrast, the robust classifier pushes the decision boundary far away from the dense input region, i.e., the mean of the data associated with class $Y = -1$.

There are a few points worth emphasizing. First, *robustness and fairness pull the decision boundary into two opposite directions*. Second, the fair model $f_{\theta_{\mathrm{f}}}$ results in predictions with higher robust errors, when compared to the optimal natural model $f_{\theta}^{\star}$, and it also increases adversarial vulnerability as the variance K increases. The variance K regulates the difference in the standard deviation of the underlying distributions associated with the protected groups and thus controls the overall distance to the decision boundary. *In summary, fairness can reduce the average distance*

of the training samples to the decision boundary which, in turn, makes the model less tolerant to adversarial noise.

This section concludes with another important result. The previous relationships continue to hold even when the optimality conditions of the fair classifier are relaxed, i.e., when λ is taking values different from ∞. Moreover, the fairness constraints always reduce the distance to the decision boundary among protected groups and this reduction is proportional to the strength of the fairness constraints (or the tightness of the required fairness bound α).

Theorem 6.3. *Consider the fair classifier $f_{\theta_{\mathrm{f}}(\lambda)}$ that optimizes Eq.* (6.3)*. It follows that, for any $\lambda \in \left(\frac{K-1}{K+1}, +\infty\right)$,*

$$\theta_{\mathrm{f}}(\lambda) = \theta_{\mathrm{f}}\,,$$

which means that the fair classifier $f_{\theta_{\mathrm{f}}(\lambda)}$ coincides with the optimal fair classifier $f_{\theta_{\mathrm{f}}}$, when the fairness penalty λ is large. while for any $\lambda \in \left[0, \frac{K-1}{K+1}\right]$, $\theta_{\mathrm{f}}(\lambda) = \mu_- - \frac{\mu_+ - \mu_-}{K^2-1} + \frac{K}{K^2-1}\sqrt{2(K^2-1)\ln\left(\frac{1-\lambda}{1+\lambda}\cdot K\right) + (\mu_+ - \mu_-)^2}$. Moreover, the parameter $\theta_{\mathrm{f}}(\lambda)$ associated with the fair classifier and the average distance to its decision boundary $\mathbb{E}\left[\Delta\left(X, f_{\theta_{\mathrm{f}}(\lambda)}\right)\right]$ are both decreasing as λ increases.

Informally speaking, Theorem 6.3 states that applying fairness constraint with large enough penalty λ will push the decision boundary towards the negative class (group with smallest variance). As a result, the average distance to the decision boundary of all samples will be reduced.

While the analysis above applies to the linear setting considered in this section, the results were empirically validated on large non-linear models. For example, Figure 6.3 compares the performance of a penalty based fair CNN model (bottom plots) with $\lambda = 1.0$ against a natural (non-fair) CNN classifier (top plots). The left plots report the task accuracy by each subgroup (denoting races) and average distance to decision boundary (right) of each subgroup. Note how the fair classifier reduces the disparities in task accuracy experienced by the various subgroups. This effect, however, also reduces the *overall* average distance to the decision boundary. As a consequence, fair models will be more vulnerable to adversarial perturbations.

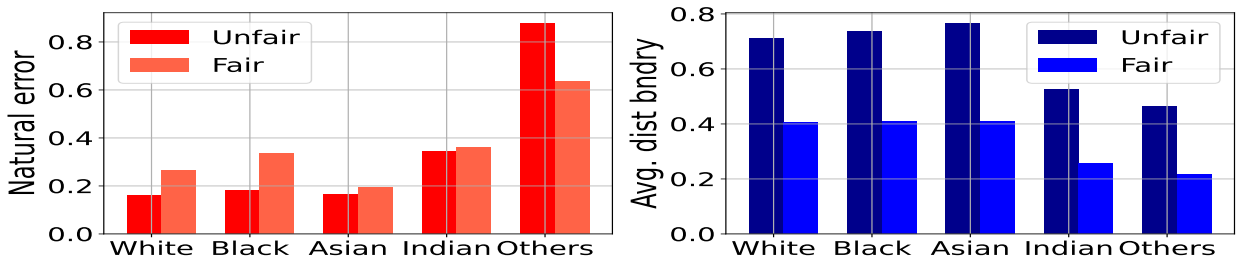


Fig. 6.3: Comparison between group's natural accuracy (left) and its average distance to the decision boundary (right) between unfair and fair models (UTK-Face dataset).

The next sections focus on assessing these theoretical intuitions onto general non-linear classifiers in a variety of settings and on devising a possible mitigation strategy to balance a good tradeoff between fairness and robustness.

6.7 Beyond the linear case

This section validates the theoretical intuitions presented above on much more complex architectures, datasets, and loss functions. The experiments focus on highlighting fairness, robustness, errors, and their relation to the distance to the decision boundary. When f_θ is a non-linear model, computing the distance to the decision boundary becomes a computational challenge. Thus, this section uses a commonly adopted proxy metric that measures the difference between the first two order statistics of the softmax outputs in the model [74, 133].

Datasets. The experiments of this section focus on three vision datasets: *UTK-Face* [144], *FMNIST* [134] and *CIFAR-10* [69]. The adopted protected groups and labels in the UTK-Face datasets are *ethnicity* (White/Black/Indian/Asian/Others) or *age* (nine age bins), resulting in two distinct tasks. For FMNIST and CIFAR, the experiments use their standard labels and assume that labels are also protected groups, mirroring the setting of previous work [79, 127, 139]. A complete description of the dataset and settings is found in Section 6.10.3.

Settings. The experiments consider several deep neural network architectures, including CNN [91], ResNet 50 [56] and VGG-13 [115]. The former uses 3 convolutional layers followed by 3 fully connected layers. Models trained on the UTK-Face data use a learning rate of $1e^{-3}$ and 70

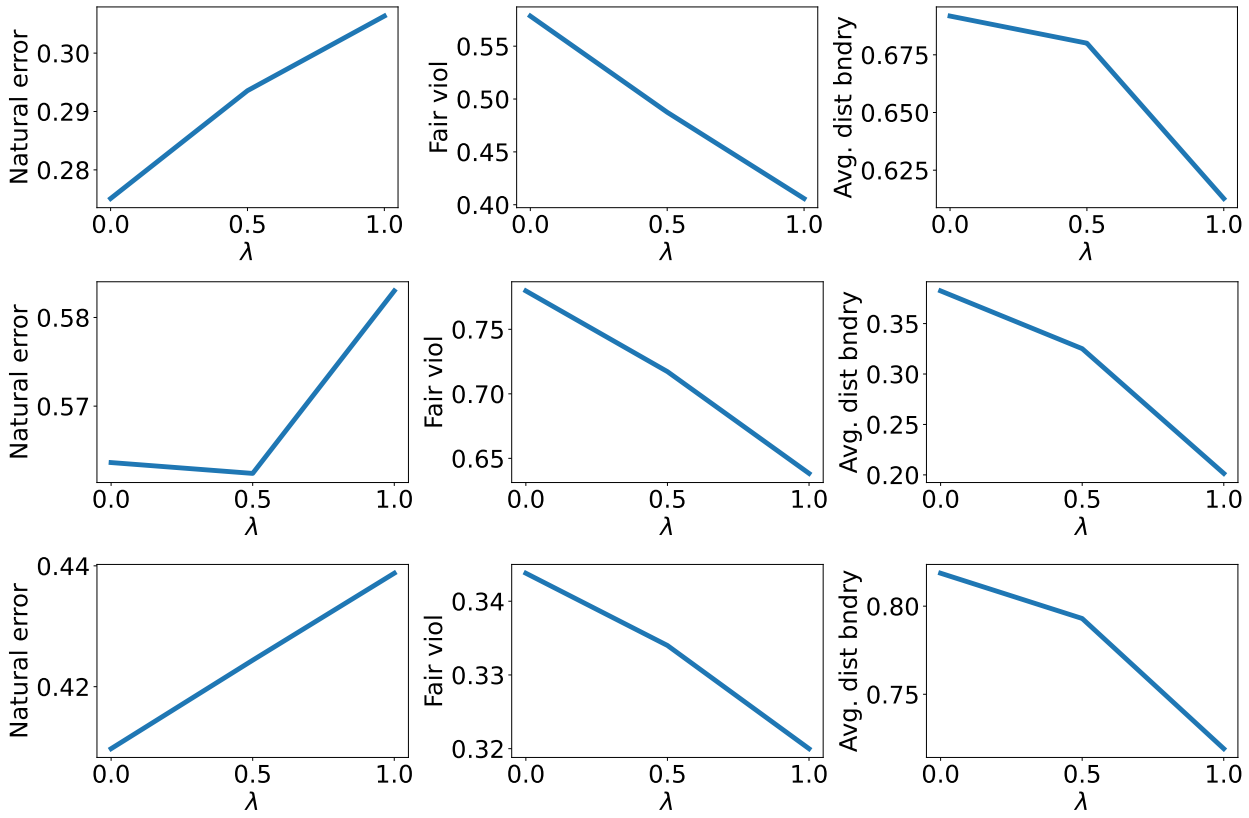


Fig. 6.4: Natural errors, fairness violations, and average distance to the decision boundary for the UTK-Face *ethnicity* (top) and *age bins* (middle) and CIFAR (bottom) datasets when varying the fairness parameter λ on a CNN model.

epochs. Those trained on FMNIST and CIFAR, use a learning rate of $1e^{-1}$ and 200 epochs, as suggested in previous work [139]. For all datasets and models, unless otherwise specified, a batch size of 32 is used. The experiments analyze penalty-based fairness method, RFGSM attacks [122], and the VGG-13 network, unless specified otherwise. Additional experiments using group-loss focused method (see Section 6.10.1), additional network architectures, and adversarial attacks are reported in Section 6.10.3.

Fairness impacts on the decision boundary

As shown by Theorem 6.3, fairness reduces the average distance of the testing samples to the decision boundary. This section illustrates how this result carries over to larger non-linear models. Figure 6.4 reports results obtained by executing the penalty-based fair models on the UTK-Face

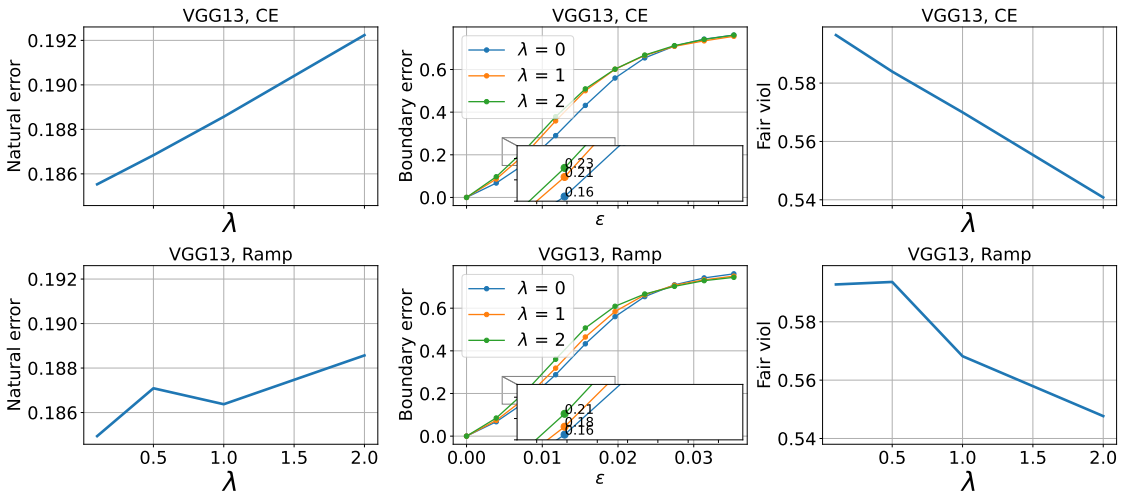


Fig. 6.5: **Top**: Natural errors (left) and fairness violations (right) on the UTKFace *ethnicity* task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different RFGSM attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss.

datasets for ethnicity (top) and age (middle) classification and on CIFAR (bottom). A clear trend emerges: As more fairness is enforced (larger λ values), the natural errors (left plots) increase, while the fairness violations (center plots) decrease. Importantly, and in agreement with the theoretical results, the experiments report a sharp reduction to the average distance to the decision boundary (right plots). This behavior renders fair models more vulnerable to adversarial attacks, as will be highlighted shortly. Similar results are also observed for the group-loss based models and other architectures (see Section 6.10.3).

Boundary errors increase as fairness decreases

This section highlights the key consequence of the sharp reduction to the average distance to the decision boundary: *the increase of the vulnerability to adversarial attacks*. Figure 6.5 (top) reports the natural errors (left), boundary errors (middle), and fairness violations (right) for a VGG-13 model trained on UTKFace dataset on the *ethnicity* task using a standard cross-entropy (CE) loss. Once again, other architectures[1] and datasets are reported in the Section and the results follow the

[1]With the caveat that VGG-13 could not be used for FMNIST since the 28x28 pixel resolution of FMNIST is smaller than that required by some VGG filters.

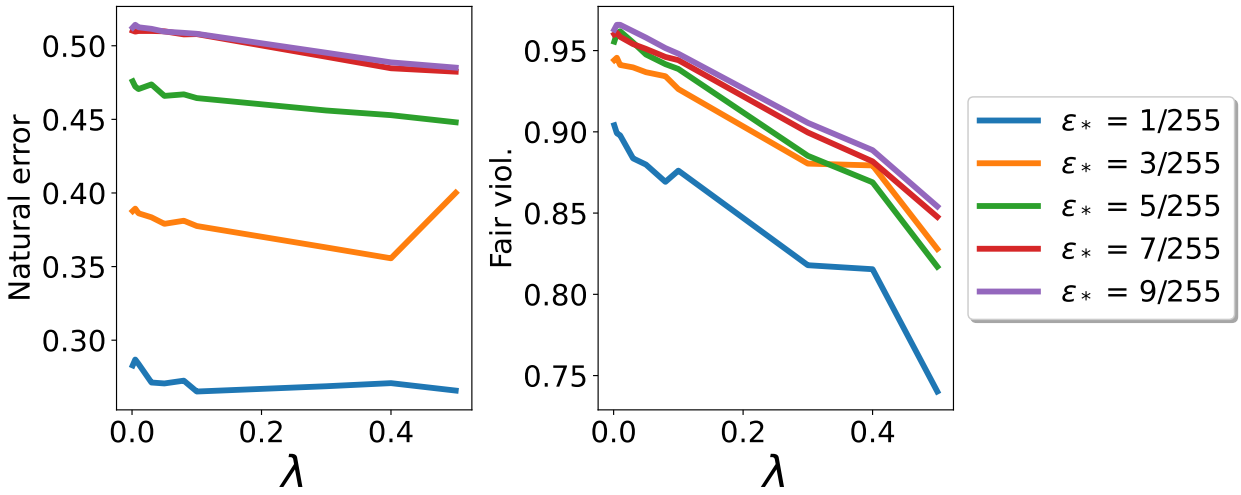


Fig. 6.6: Natural error (left) and fairness violation (right) at varying of the margin perturbation ϵ_* and fairness parameters λ.

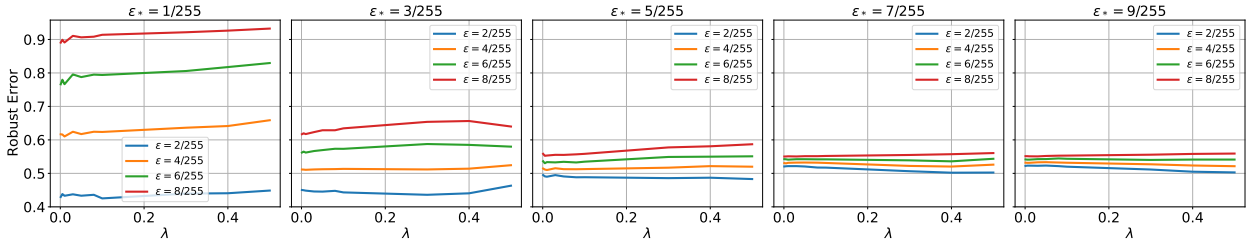


Fig. 6.7: Robust errors for different attack levels ϵ of a robust and fair classifier at varying of the margin perturbation ϵ_* and fairness parameters λ.

same trends as those reported here.

The natural errors and fairness violations are reported for *fair* classifiers, at varying of the fairness violation parameter λ. The boundary errors (middle) are reported for classifiers satisfying various fairness levels (i.e., using different λ values) and at varying of the strength ϵ of the desired robustness level (see Equation (6.4)).

Notice how, compared to the natural models, the fair models incur much higher natural and boundary errors. In particular, the relative increase in boundary errors are significant: The fairness models have boundary errors that are up to 9% larger than their natural counterparts. These observations match the theoretical analysis and highlight a significant increase in vulnerability to adversarial examples by the fair models, even for moderate selections of the fairness violation parameters λ.

Enforcing Fairness and Robustness Simultaneously

This section considers an additional experiment that highlights the potential negative impact of fairness on robustness. The experiment involves a classifier attempting to achieve both fairness and robustness, similar to the approach proposed by Xu et al. [139]. Their method involved incorporating two fairness components to align per-class natural/robust accuracy per class with overall natural/robust accuracy (refer to Equation (9) in [139] for more details). Our approach is similar, in spirit, as it adds both robustness and fairness regularization terms to the standard classification objective function.

The resulting model aims at solving the following regularized ERM problem:

$$\min_{\theta} \mathcal{L}_\theta(D) + \frac{1}{n}\sum_{i=1}^{n} \max_{\|\tau\|_p \leq \epsilon_*} \ell(f_\theta(X_i + \tau), Y_i)$$
$$+ \lambda \sum_{a \in \mathcal{A}} \left| {}^{1}\!/_{|D_a|} \sum_{(X,A,Y)\in D_a} \ell(f_\theta(X), Y) - \frac{1}{n}\sum_{i=1}^{n} \ell(f_\theta(X_i), Y_i) \right|$$

using stochastic gradient descent. The second component aims at increasing the robustness of the classifier under a margin perturbation ϵ_*, following the PGD training [80] with perturbation norm $p = \infty$. It works by first generating adversarial samples $X_i + \tau$, where $\|\tau\|_\infty \leq \epsilon_*$, and then the learning progress aims at minimizing the loss between the model prediction for that adversarial samples and the ground-truth $\ell(f_\theta(X_i + \tau), Y_i)$. The larger the margin perturbation ϵ_*, the more robust the resulting classifier. The third component implements a penalty-based fairness strategy [7], which promotes fairness by penalizing the difference among each groups' average loss and the overall's average loss.

The experiments vary the margin perturbation ϵ_* (robustness) and the penalty value λ (fairness). Figure 6.6 reports the (natural) error (left) and fairness violations (right) for different levels of the margin perturbation ϵ_* on the UTK-Face (ethnicity) dataset. As expected, enforcing larger margin perturbations ϵ_* increases model robustness, but at the cost of significantly increasing the natural errors. Increasing the fairness parameter λ decreases the fairness violation.

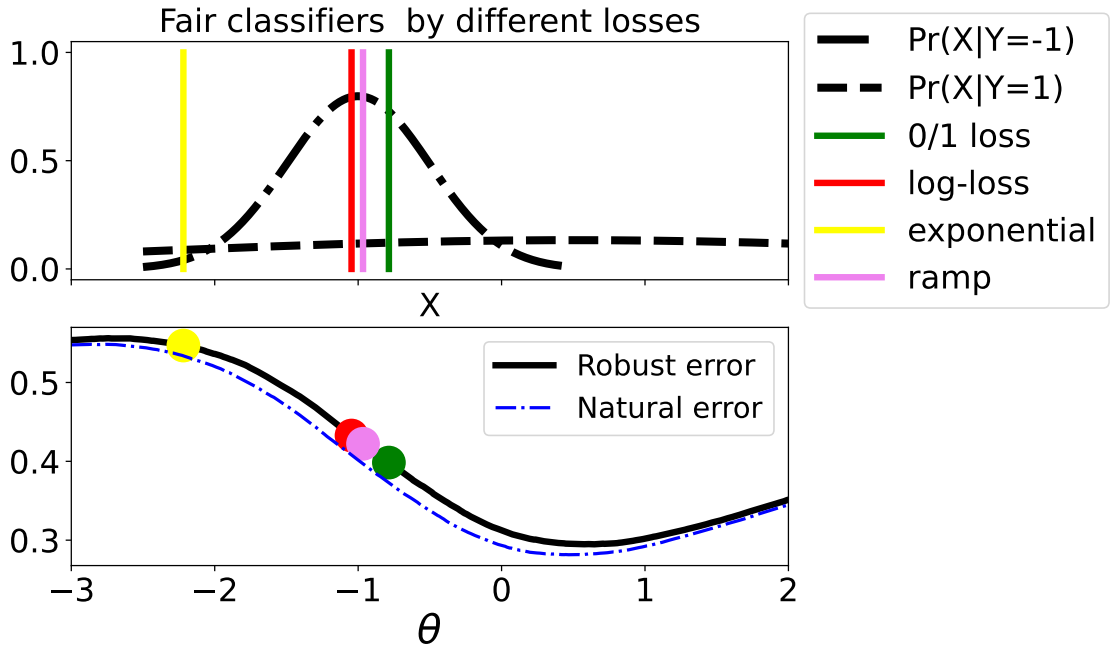


Fig. 6.8: Classifiers using different losses (top) and the associated natural and robust errors (bottom).

Figure 6.7 reports the robust errors under different levels of adversarial attacks, which are specified by the level of perturbation ϵ. Notice how the level of defense ϵ_* correlates with higher robustness (smaller robust errors) for all fairness parameters λ tested. These results show the challenge to achieving simultaneously robustness, fairness, and accuracy. The results also suggest that the incorporation of both robustness and fairness, as proposed in [139], may not effectively reduce the trade-off between accuracy, robustness, and fairness.

Overall, the results show that, without a careful consideration, inducing a desired equity property on a learning task may create significant security challenges. This should not be read as an endorsement to satisfy a single property, but as a call for additional research at the intersection of fairness and robustness in order to design appropriate tradeoffs.

6.8 A mitigating solution with bounded losses

While the previous sections have shown that the conflict between fairness and robustness is unavoidable, this section proposes a theoretically motivated solution attempting to attenuate this tension. The proposed solution relies on the observation that, using standard (unbounded) loss functions, misclassified samples lying far away from the decision boundary are associated to much larger losses than those which are closer to it. Recall also that the decision boundary was found as the predominant factor linking fairness and robustness. This key observation suggests the use of a bounded loss function, defined as [30, 50]:

$$\ell_{Ramp}(f_\theta(X), Y) = \min(1, \max(0, 1 - Y f_\theta(X))).$$

and referred to as *Ramp loss*, with domain $(0, 1]$. The proposed strategy simply applies this loss function to a fair classifier (Equation 6.3). Its benefits can be appreciated in Figure 6.8, which reports the results for the same setting used in the previous section and compares a fair classifier trained using the ramp loss with one trained using a 0/1-loss (which is also bounded but not differentiable), a log-loss, and an exponential loss (both unbounded) (top). The results show that the fair classifier trained using a ramp-loss is the least impacted by misclassified samples, resulting in lower robust errors compared to unbounded losses. It can be observed in the bottom subplot, where its associated loss is the closest, among all differentiable losses, to the local minima. The observed benefits of the ramp loss also carry over high-dimensional data and non-linear models, as shown in Figure 6.5 bottom, and further reported in Section 6.10.3.

6.9 Conclusions

This chapter was motivated by two key challenges brought by the the adoption of modern machine learning systems in consequential domains: *fairness* and *robustness*. The chapter observed and analyzed the relationship between these two important machine-learning properties and showed

that fairness increases vulnerability to adversarial examples. Through a theoretical analysis on linear models, this work provided a new understanding of why such tension arises and identified the distance to the decision boundary as a key explanation factor linking fairness and robustness. These theoretical findings were validated on non-linear models through extensive experiments on a variety of vision tasks. Finally, building from this new understanding, the chapter proposed a simple, yet effective, strategy to find a better balance between accuracy, fairness and robustness. We hope these results could stimulate a needed discussion and research at the intersection of fairness and robustness to achieve appropriate tradeoffs.

6.10 Section

6.10.1 Missing proofs

Proof of Theorem 6.1

Proof. i) Notice that the natural classification error and its derivative can be expressed as

$$\begin{aligned}
\mathcal{L}_\theta^{\text{nat}} &= \Pr\left(f_\theta(X) \neq Y\right) \\
&= \frac{1}{2}\Pr\left(f_\theta(X) \neq 1 \mid Y = 1\right) + \\
&\quad \frac{1}{2}\Pr\left(f_\theta(X) \neq -1 \mid Y = -1\right) \\
&= \frac{1}{2}\int_{-\infty}^{\theta} \frac{1}{\sqrt{2\pi}K} \exp\left(-\frac{(x-\mu_+)^2}{2K^2}\right) dx + \\
&\quad \frac{1}{2}\int_{\theta}^{+\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-\mu_-)^2}{2}\right) dx\,.
\end{aligned}$$

and

$$\left(\mathcal{L}_\theta^{\text{nat}}\right)' = \frac{\left[\exp\left(-\frac{(\theta-\mu_+)^2}{2K^2}\right) - K\exp\left(-\frac{(\theta-\mu_-)^2}{2}\right)\right]}{2K\sqrt{2\pi}}\,.$$

The derivative $\left(\mathcal{L}_\theta^{\text{nat}}\right)'$ turns out to be an increasing function over the interval $[\mu_-, \mu_+]$ with $\left(\mathcal{L}_{\mu_-}^{\text{nat}}\right)' < 0$ and $\left(\mathcal{L}_{\mu_+}^{\text{nat}}\right)' > 0$ due to the assumption that $K < B_K < \exp\left(-\frac{(\mu_+-\mu_-)^2}{2}\right)$.

Since the Bayes classifier is to minimize the natural classification error, $\overset{\star}{\theta}$ is supposed to be the unique root of $(\mathcal{L}_\theta^{\mathrm{nat}})'$, i.e.,

$$\frac{\left[\exp\left(-\frac{(\overset{\star}{\theta}-\mu_+)^2}{2K^2}\right) - K\exp\left(-\frac{(\overset{\star}{\theta}-\mu_-)^2}{2}\right)\right]}{2K\sqrt{2\pi}} = 0\,.$$

By solving the equation above, we end up with the following.

$$\overset{\star}{\theta} = \mu_- \frac{\mu_+ - \mu_-}{K^2-1} + \frac{K}{K^2-1}\sqrt{2(K^2-1)\ln(K) + (\mu_+ - \mu_-)^2}\,, \tag{6.11}$$

which belongs to the open interval (μ_-, μ_+).

ii) Equalized classification errors require the following equations hold.

$$\begin{aligned}
\Pr\left(f_{\theta_{\mathrm{f}}}(X) \neq 1 \mid Y = 1\right) &= \int_{-\infty}^{\theta_{\mathrm{f}}} \frac{1}{\sqrt{2\pi}K}\exp\left(-\frac{(x-\mu_+)^2}{2K^2}\right)dx \\
&= \int_{\theta_{\mathrm{f}}}^{+\infty} \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{(x-\mu_-)^2}{2}\right)dx \\
&= \Pr\left(f_{\theta_{\mathrm{f}}}(X) \neq -1 \mid Y = -1\right)\,,
\end{aligned}$$

which leads to the result that

$$\theta_{\mathrm{f}} = \mu_- + \frac{\mu_+ - \mu_-}{K+1} > \mu_- + \epsilon\,,$$

where the inequality is due to the assumption that $K < (\mu_+ - \mu_-)/\epsilon - 1 \leq B_K$.

iii) The robust classification error and its partial derivative can then be given by the following:

$$\begin{aligned}
\mathcal{L}_\theta^{\mathrm{rob}}(\epsilon) = &\ \Pr\left(\exists\, |\tau| \le \epsilon,\ f_\theta(X+\tau) \neq Y\right) \\
= &\ \frac{1}{2}\Pr\left(\exists\, |\tau| \le \epsilon,\ f_\theta(X+\tau) \neq 1 \mid Y = 1\right) + \\
&\ \frac{1}{2}\Pr\left(\exists\, |\tau| \le \epsilon,\ f_\theta(X+\tau) \neq -1 \mid Y = -1\right) \\
= &\ \frac{1}{2}\left(\Pr\left(X \le \theta + \epsilon \mid Y = 1\right) + \Pr\left(X > \theta - \epsilon \mid Y = -1\right)\right) \\
= &\ \frac{1}{2}\int_{-\infty}^{\theta+\epsilon} \frac{1}{\sqrt{2\pi}K} \exp\left(-\frac{(x-\mu_+)^2}{2K^2}\right) dx + \\
&\ \frac{1}{2}\int_{\theta-\epsilon}^{+\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-\mu_-)^2}{2}\right) dx\,,
\end{aligned}$$

and

$$\frac{\partial}{\partial\theta}\mathcal{L}_\theta^{\mathrm{rob}}(\epsilon) = \frac{\left[\exp\left(-\frac{(\theta+\epsilon-\mu_+)^2}{2K^2}\right) - K\exp\left(-\frac{(\theta-\epsilon-\mu_-)^2}{2}\right)\right]}{2K\sqrt{2\pi}}\,. \tag{6.12}$$

By the assumptions made about K and ϵ, there exists a unique root $\theta_{\mathrm{r}}^{(\epsilon)} \in (\mu_-, \mu_+ - \epsilon)$ of $\frac{\partial}{\partial\theta}\mathcal{L}_\theta^{\mathrm{rob}}(\epsilon) = 0$ such that the robust error $\mathcal{L}_\theta^{\mathrm{rob}}(\epsilon)$ is decreasing over $(\mu_-, \theta_{\mathrm{r}}^{(\epsilon)})$ while increasing over $(\theta_{\mathrm{r}}^{(\epsilon)}, \mu_+)$, i.e.,

$$\frac{\partial}{\partial\theta}\mathcal{L}_\theta^{\mathrm{rob}}(\epsilon) \begin{cases} < 0\,, & \theta \in (\mu_-, \theta_{\mathrm{r}}^{(\epsilon)})\,, \\ > 0\,, & \theta \in (\theta_{\mathrm{r}}^{(\epsilon)}, \mu_+)\,, \end{cases} \tag{6.13}$$

which indicates that $\theta_{\mathrm{r}}^{(\epsilon)}$ essentially minimizes the robust classification error.

Therefore, by solving $\frac{\partial}{\partial\theta}\mathcal{L}_\theta^{\mathrm{rob}}(\epsilon)\Big|_{\theta=\theta_{\mathrm{r}}^{(\epsilon)}} = 0$, we are able to derive the robust classifier as follows.

$$\theta_{\mathrm{r}}^{(\epsilon)} = \mu_- - \frac{\mu_+ - \mu_- - (K^2+1)\epsilon}{K^2-1} + \frac{K}{K^2-1}\sqrt{2(K^2-1)\ln(K) + (\mu_+ - \mu_- - 2\epsilon)^2}\,,$$

which satisfies the following

$$\theta_{\mathrm{r}}^{(\epsilon)} \le \mu_+ - \epsilon \le \mu_+\,.$$

iv) The next step is to compare the three different classifiers we just obtained. We start with the Bayes and fair classifiers.

$$\begin{aligned}\overset{\star}{\theta} - \theta_{\mathrm{f}} =& \frac{K\left(\sqrt{2(K^2-1)\ln(K) + (\mu_+ - \mu_-)^2} - \mu_+ - \mu_-\right)}{K^2-1} \\ >& \frac{K\left(\sqrt{(\mu_+ - \mu_-)^2} - \mu_+ - \mu_-\right)}{K^2-1} = 0\,,\end{aligned}$$

where the inequality comes from the assumption that K is strictly larger than 1. It indicates that the threshold of the Bayes classifier is greater than that of the fair classifier. Then, we move on to the comparison between the Bayes and robust classifier. Note that the robust classifier is identical to the Bayes classifier when ϵ is 0, i.e., $\theta_{\mathrm{r}}^{(0)} = \overset{\star}{\theta}$. Besides, we have that

$$\begin{aligned}\frac{\partial}{\partial\epsilon}\theta_{\mathrm{r}}^{(\epsilon)} =& \frac{K^2 + 1 - \frac{2(\mu_+ - \mu_- - 2\epsilon)K}{\sqrt{(\mu_+ - \mu_- - 2\epsilon)^2 + 2(K^2-1)\ln(K)}}}{K^2-1} \\ >& \frac{K^2 + 1 - \frac{2(\mu_+ - \mu_- - 2\epsilon)K}{\sqrt{(\mu_+ - \mu_- - 2\epsilon)^2}}}{K^2-1} \qquad (6.14) \\ =& \frac{K^2 - 2K + 1}{K^2-1} \qquad (6.15) \\ =& \frac{K-1}{K+1} > 0\,,\end{aligned}$$

where Equation (6.14) is due to the fact that $K > 1$ and Equation (6.15) comes from the assumption that $\epsilon \leq \frac{\mu_+ - \mu_-}{2}$, which ensures that $\mu_+ - \mu_- - 2\epsilon$ is strictly positive. Therefore, the partial derivative of $\theta_{\mathrm{r}}^{(\epsilon)}$ in ϵ is strictly positive over the interval $\left[0, \frac{\mu_+ - \mu_-}{2}\right]$. As a consequence, for any $\epsilon \in \left[0, \frac{\mu_+ - \mu_-}{2}\right]$, the following relation always holds that $\mu_+ \geq \theta_{\mathrm{r}}^{(\epsilon)} \geq \theta_{\mathrm{r}}^{(0)} = \overset{\star}{\theta}$. Putting things together, we end up with following relation: for any $\epsilon \in \left[0, \frac{\mu_+ - \mu_-}{2}\right]$ and $K \in (1, B_K)$,

$$\mu_- + \epsilon \leq \theta_{\mathrm{f}} \leq \overset{\star}{\theta} \leq \theta_{\mathrm{r}}^{(\epsilon)} \leq \mu_+ - \epsilon\,.$$

□

Proof of Corollary 6.2

Proof. Note that, by Equation (6.13), the robust classification error is strictly decreasing over $(\mu_-, \theta_{\mathrm{r}}^{(\epsilon)})$. Then, by Equation (6.8) in Proposition 6.1, the three classifiers satisfy the following relation

$$\mu_- \leq \theta_{\mathrm{f}} \leq \overset{\star}{\theta} \leq \theta_{\mathrm{r}}^{(\epsilon)} .$$

Due to contiguity of $\mathcal{L}_\theta^{\mathrm{rob}}(\epsilon)$, we can argue that, for any $\epsilon \in \left(0, \frac{\mu_+ - \mu_-}{2}\right)$ and $K \in (1, B_K)$,

$$\mathcal{L}_{\theta_{\mathrm{f}}}^{\mathrm{rob}}(\epsilon) \geq \mathcal{L}_{\overset{\star}{\theta}}^{\mathrm{rob}}(\epsilon) \geq \mathcal{L}_{\theta_{\mathrm{r}}^{(\epsilon)}}^{\mathrm{rob}}(\epsilon) .$$

□

Proof of Corollary 6.3

Proof. The boundary error and its partial derivative are presented in the following:

$$\begin{aligned}
&\mathcal{L}_\theta^{\mathrm{bdy}}(\epsilon) \\
&\quad = \Pr\left(\exists\, |\tau| \leq \epsilon,\ f_\theta(X+\tau) \neq Y,\ f_\theta(X) = Y\right) \\
&\quad = \frac{1}{2}\Pr\left(\exists\, |\tau| \leq \epsilon,\ f_\theta(X+\tau) = -1,\ f_\theta(X) = 1 \mid Y = 1\right) + \\
&\qquad \frac{1}{2}\Pr\left(\exists\, |\tau| \leq \epsilon,\ f_\theta(X+\tau) = 1,\ f_\theta(X) = -1 \mid Y = -1\right) \\
&\quad = \frac{1}{2}\Pr\left(\theta < X \leq \theta + \epsilon \mid Y = 1\right) + \\
&\qquad \frac{1}{2}\Pr\left(\theta \geq X > \theta - \epsilon \mid Y = -1\right) \\
&\quad = \frac{1}{2}\int_\theta^{\theta+\epsilon} \frac{1}{\sqrt{2\pi}K} \exp\left(-\frac{(x-\mu_+)^2}{2K^2}\right) dx + \\
&\qquad \frac{1}{2}\int_{\theta-\epsilon}^{\theta} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-\mu_-)^2}{2}\right) dx ,
\end{aligned}$$

and

$$\begin{aligned}\frac{\partial}{\partial\theta}\mathcal{L}_{\theta}^{\mathrm{bdy}}\left(\epsilon\right) = & \frac{\exp\left(-\frac{(\theta+\epsilon-\mu_{+})^{2}}{2K^{2}}\right)}{2\sqrt{2\pi}K} - \frac{\exp\left(-\frac{(\theta-\mu_{+})^{2}}{2K^{2}}\right)}{2\sqrt{2\pi}K} + \\ & \frac{\exp\left(-\frac{(\theta-\mu_{-})^{2}}{2}\right)}{2\sqrt{2\pi}} - \frac{\exp\left(-\frac{(\theta-\epsilon-\mu_{-})^{2}}{2}\right)}{2\sqrt{2\pi}} \\ = & \frac{g(\epsilon;\theta,K) - g(0;\theta,K)}{2\sqrt{2\pi}}\,,\end{aligned}$$

where $g : \mathbb{R} \mapsto \mathbb{R}$ is an auxiliary function shown as follows

$$g(\epsilon;\theta,K) = \frac{\exp\left(-\frac{(\theta+\epsilon-\mu_{+})^{2}}{2K^{2}}\right)}{K} - \exp\left(-\frac{(\theta-\epsilon-\mu_{-})^{2}}{2}\right)\,. \tag{6.16}$$

By Proposition 6.1, the partial derivative of the boundary error in θ is always negative for any $\theta \in \left[\theta_{\mathrm{f}}, \theta_{\mathrm{r}}^{(\epsilon)}\right]$ because

$$\frac{\partial}{\partial\theta}\mathcal{L}_{\theta}^{\mathrm{bdy}}\left(\epsilon\right) = \frac{g(\epsilon;\theta,K) - g(0;\theta,K)}{2\sqrt{2\pi}} < 0\,.$$

It leads to the following result that the boundary error $\mathcal{L}_{\theta}^{\mathrm{rob}}\left(\epsilon\right)$ decreases in θ over $\left[\theta_{\mathrm{f}}, \theta_{\mathrm{r}}^{(\epsilon)}\right]$, and, therefore,

$$\mathcal{L}_{\theta_{\mathrm{f}}}^{\mathrm{bdy}}\left(\epsilon\right) \geq \mathcal{L}_{\theta^{\star}}^{\mathrm{bdy}}\left(\epsilon\right) \geq \mathcal{L}_{\theta_{\mathrm{r}}^{(\epsilon)}}^{\mathrm{bdy}}\left(\epsilon\right)\,.$$

□

Proposition 6.1. *For any* $\epsilon \in \left[0, \frac{\mu_{+}-\mu_{-}}{4}\right]$, $K \in \left(1, \bar{B}_{K}\right)$, *and* $\theta \in \left[\theta_{\mathrm{f}}, \theta_{\mathrm{r}}^{(\epsilon)}\right]$, *the following relation holds*

$$g(\epsilon;\theta,K) \leq g(0;\theta,K)\,, \tag{6.17}$$

where the function g *is defined in Equation* (6.16).

Proof. Note that the partial derivative of the function g in ϵ can be given by

$$\frac{\partial}{\partial \epsilon} g(\epsilon; \theta, K) = \frac{\mu_+ - \theta - \epsilon}{K^3} \exp\left(-\frac{(\mu_+ - \theta - \epsilon)^2}{2K^2}\right) - (\theta - \epsilon - \mu_-) \exp\left(-\frac{(\theta - \epsilon - \mu_-)^2}{2}\right).$$

In order to establish the result in Equation (6.17), it suffices to demonstrate that the partial derivative $\frac{\partial}{\partial \epsilon} g(\epsilon; \theta, K)$ is negative, which implies that the function $g(\epsilon; \theta, K)$ is strictly decreasing over $\left[0, \frac{\mu_+ - \mu_-}{4}\right]$. Note that

$$\begin{aligned}
&\ln\left(\frac{\mu_+ - \theta - \epsilon}{K^3} \exp\left(-\frac{(\mu_+ - \theta - \epsilon)^2}{2K^2}\right)\right) - \ln\left((\theta - \epsilon - \mu_-) \exp\left(-\frac{(\theta - \epsilon - \mu_-)^2}{2}\right)\right) \\
= &\ -\left[\ln(\theta - \epsilon - \mu_-) - \ln(\mu_+ - \theta - \epsilon) -\right] + \left[\frac{(\theta - \epsilon - \mu_-)^2}{2} - \frac{(\mu_+ - \theta - \epsilon)^2}{2K^2}\right] - 3\ln(K) \\
= &\ \frac{1}{2}\left(q(\theta; \epsilon, K) - p(\theta; \epsilon, K) - 4\ln(K)\right) \leq 0\,,
\end{aligned}$$

where the last inequality comes from Proposition 6.2. It follows that, due to monotonicity of the function $x \mapsto \ln(x)$,

$$\begin{aligned}
\frac{\partial}{\partial \epsilon} g(\epsilon; \theta, K) = &\ \frac{\mu_+ - \theta - \epsilon}{K^3} \exp\left(-\frac{(\mu_+ - \theta - \epsilon)^2}{2K^2}\right) - \\
&\ (\theta - \epsilon - \mu_-) \exp\left(-\frac{(\theta - \epsilon - \mu_-)^2}{2}\right) \\
\leq &\ 0\,,
\end{aligned}$$

which completes our proof here. □

Proposition 6.2. *For any* $\epsilon \in \left[0, \frac{\mu_+ - \mu_-}{4}\right]$, $K \in \left(1, \bar{B}_K\right]$, *and* $\theta \in \left[\theta_{\mathrm{f}}, \theta_{\mathrm{r}}^{(\epsilon)}\right]$, *the following relation always holds:*

$$p(\theta; \epsilon, K) \geq q(\theta; \epsilon, K) - 4\ln(K)\,,$$

where

$$p(\theta;\epsilon,K) = \ \ln\left((\theta-\epsilon-\mu_-)^2\right) - \ln\left(\frac{(\mu_+-\theta-\epsilon)^2}{K^2}\right), \tag{6.18}$$

$$q(\theta;\epsilon,K) = \ (\theta-\epsilon-\mu_-)^2 - \frac{(\mu_+-\theta-\epsilon)^2}{K^2}. \tag{6.19}$$

Proof. First off, observe that the functions p and q are both increasing over $\left[\theta_{\mathrm{f}}, \theta_{\mathrm{r}}^{(\epsilon)}\right]$. It follows that, for any $\left[\theta_{\mathrm{f}}, \theta_{\mathrm{r}}^{(\epsilon)}\right]$,

$$p(\theta;\epsilon,K) \geq \ p(\theta_{\mathrm{f}};\epsilon,K) \tag{6.20}$$

$$\geq \ -2\ln(K) = 2\ln(K) - 4\ln(K) \tag{6.21}$$

$$= \ q\left(\theta_{\mathrm{r}}^{(\epsilon)};\epsilon,K\right) - 4\ln(K) \tag{6.22}$$

$$\geq \ q(\theta;\epsilon,K) - 4\ln(K), \tag{6.23}$$

where Equation (6.20) and (6.23) come from monotonicity of the functions p and q. Equation (6.21) and (6.22) are due to Proposition 6.4 and 6.3 respectively. □

Proposition 6.3. *For any* $\epsilon \in \left[0, \frac{\mu_+-\mu_-}{4}\right]$ *and* $K \in \left(1, \bar{B}_K\right]$,

$$q\left(\theta_{\mathrm{r}}^{(\epsilon)};\epsilon,K\right) = 2\ln(K),$$

where the definition of the function q *is given in Equation* (6.19).

Proof. Due to optimality of $\theta_{\mathrm{r}}^{(\epsilon)}$ in terms of robust error, $\theta_{\mathrm{r}}^{(\epsilon)}$ should be a solution to Equation (6.12), i.e.,

$$\exp\left(-\frac{(\theta_{\mathrm{r}}^{(\epsilon)}+\epsilon-\mu_+)^2}{2K^2}\right) - K\exp\left(-\frac{(\theta_{\mathrm{r}}^{(\epsilon)}-\epsilon-\mu_-)^2}{2}\right) = 0, \tag{6.24}$$

which leads to the following, by multiplying the both sides with the term $\exp\left(\frac{(\theta_{\mathrm{r}}^{(\epsilon)}-\epsilon-\mu_-)^2}{2}\right)$,

$$\exp\left(\frac{q(\theta_{\mathrm{r}}^{(\epsilon)};\epsilon,K)}{2}\right) = K,$$

and

$$q(\theta_{\mathrm{r}}^{(\epsilon)}; \epsilon, K) = 2\ln(K)\,.$$

□

Proposition 6.4. *For any* $\epsilon \in \left[0, \frac{\mu_+ - \mu_-}{4}\right]$ *and* $K \in \left(1, \bar{B}_K\right]$,

$$p\left(\theta_{\mathrm{f}}; \epsilon, K\right) \geq -2\ln(K)\,, \tag{6.25}$$

where the definition of the function p *is given in Equation* (6.18)*.*

Proof. Equation (6.25) can be rewritten into the following equivalent form

$$\exp\left(p\left(\theta_{\mathrm{f}}; \epsilon, K\right) + 2\ln(K)\right) = K - \frac{(K-1)\epsilon}{\frac{\mu_+ - \mu_-}{K+1} - \frac{\epsilon}{K}} > 1\,. \tag{6.26}$$

Note that

$$\begin{aligned} K \leq \bar{B}_K \implies & K + \frac{1}{K} \leq \frac{\mu_+ - \mu_-}{\epsilon} - 2 \\ \implies & \frac{\mu_+ - \mu_-}{K+1} \geq \frac{K+1}{K}\epsilon\,, \end{aligned}$$

which leads to the following result

$$\begin{aligned} & \exp\left(p\left(\theta_{\mathrm{f}}; \epsilon, K\right) + 2\ln(K)\right) \\ = \; & K - \frac{(K-1)\epsilon}{\frac{\mu_+ - \mu_-}{K+1} - \frac{\epsilon}{K}} \geq K - (K-1) \\ = \; & 1\,. \end{aligned}$$

It helps complete our proof here. □

Proof of Theorem 6.4

Proof. Since f_θ is essentially a linear classifier, the distance to the decision boundary of f_θ is

simply the absolute value between the feature X and the threshold θ, i.e., $\Delta\left(X, f_\theta\right) = |X - \theta|$. Notice that the average distance to the decision boundary of f_θ can then be given by

$$\begin{aligned}
\mathbb{E}\left[\Delta\left(X, f_\theta\right)\right] = & \ \mathbb{E}\left[|X-\theta|\right] \\
= & \ \mathbb{E}\left[|X-\theta| \mid Y=1\right] \cdot \Pr\left(Y=1\right) + \\
& \ \mathbb{E}\left[|X-\theta| \mid Y=-1\right] \cdot \Pr\left(Y=-1\right) \\
= & \ \frac{1}{2} \int_{-\infty}^{+\infty} \frac{|x-\theta|}{\sqrt{2\pi}K} \exp\left(-\frac{(x-\mu_+)^2}{2K^2}\right) dx + \\
& \ \frac{1}{2} \int_{-\infty}^{+\infty} \frac{|x-\theta|}{\sqrt{2\pi}} \exp\left(-\frac{(x-\mu_-)^2}{2}\right) dx \,,
\end{aligned}$$

whose derivative can be expressed as

$$\left(\mathbb{E}\left[\Delta\left(X, f_\theta\right)\right]\right)' = 2\left(\Phi(\theta-\mu_-) - \Phi\left(\frac{\mu_+ - \theta}{K}\right)\right) \,,$$

where Φ represents the cumulative distribution function associated with the standard normal distribution. Recall that

$$\theta_{\mathrm{f}} = \mu_- + \frac{\mu_+ - \mu_-}{K+1} \,.$$

It follows that

$$\left(\mathbb{E}\left[\Delta\left(X, f_\theta\right)\right]\right)' \begin{cases} < 0 \,, & \theta \in \left(\mu_-, \theta_{\mathrm{f}}\right) , \\ > 0 \,, & \theta \in \left(\theta_{\mathrm{f}}, \mu_+\right) , \end{cases}$$

which implies that the average distance strictly decreases over $(\mu_-, \theta_{\mathrm{f}})$ while increases over $(\theta_{\mathrm{f}}, \mu_+)$. By the relation shown in Equation (6.8), we figure out that, for any $\epsilon \in \left[0, \frac{\mu_+ - \mu_-}{2}\right]$ and $K \in (1, B_K]$,

$$\mathbb{E}\left[\Delta\left(X, f_{\theta_{\mathrm{r}}^{(\epsilon)}}\right)\right] \geq \mathbb{E}\left[\Delta\left(X, f_{\theta}^{\star}\right)\right] \geq \mathbb{E}\left[\Delta\left(X, f_{\theta_{\mathrm{f}}}\right)\right] \,.$$

Moreover, θ_{f} is the minimizer of the average distance $\mathbb{E}\left[\Delta\left(X, f_\theta\right)\right]$ over the interval $[\mu_-, \mu_+]$. $\square$

Proof of Theorem 6.5

Proof. First off, notice that $\theta_{\mathrm{f}}(\lambda)$ is the minimizer of $\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}}$ over $[\mu_-, \mu_+]$, where $\mathcal{L}_\theta^{\mathrm{fair}}$ is a shorthand for

$$\left|\Pr\left(f_\theta(X) \neq 1 \mid Y = 1\right) - \Pr\left(f_\theta(X) \neq -1 \mid Y = -1\right)\right| .$$

Thus, $\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}}$ can be presented in a piecewise way as follows:

1) if $\theta \leq \theta_{\mathrm{f}}$,

$$\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}} = (1-\lambda)\Pr\left(f_\theta(X) \neq 1 \mid Y = 1\right) + (1+\lambda)\Pr\left(f_\theta(X) \neq -1 \mid Y = -1\right) ;$$

2) if $\theta > \theta_{\mathrm{f}}$,

$$\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}} = (1+\lambda)\Pr\left(f_\theta(X) \neq 1 \mid Y = 1\right) + (1-\lambda)\Pr\left(f_\theta(X) \neq -1 \mid Y = -1\right) .$$

We start with the first case where θ is no greater than θ_{f}. The partial derivative of $\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}}$ is then given by

$$\frac{\partial\left(\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}}\right)}{\partial\theta} \;=\; \frac{1-\lambda}{2\sqrt{2\pi}K}\exp\left(-\frac{(\theta-\mu_+)^2}{2K^2}\right) \;-\; \frac{1+\lambda}{2\sqrt{2\pi}}\exp\left(-\frac{(\theta-\mu_-)^2}{2}\right) . \quad (6.27)$$

Observe that, when $\lambda \in [0,1]$, this partial derivative is increasing over $[\mu_-, \theta_{\mathrm{f}}]$ with its value at θ_{f} no greater than 0 while it is always non-positive for any $\lambda \in (1, +\infty)$ and $\theta \in [\mu_-, \theta_{\mathrm{f}}]$. Therefore, the partial derivative presented in Equation (6.27) proves to be non-positive, whatever λ, which implies that the function $\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}}$ decreases over $[\theta, \theta_{\mathrm{f}}]$ and we merely need to focus on the case of $\theta \in [\theta_{\mathrm{f}}, \mu_+]$ in pursuit of its minimizer. Then, we continue to investigate the case where θ is greater than θ_{f}. Likewise, the partial derivative of $\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}}$ can be expressed as

$$\frac{\partial\left(\mathcal{L}_\theta^{\mathrm{nat}} + \lambda \cdot \mathcal{L}_\theta^{\mathrm{fair}}\right)}{\partial\theta} \;=\; \frac{1+\lambda}{2\sqrt{2\pi}K}\exp\left(-\frac{(\theta-\mu_+)^2}{2K^2}\right) \;-\; \frac{1-\lambda}{2\sqrt{2\pi}}\exp\left(-\frac{(\theta-\mu_-)^2}{2}\right) . \quad (6.28)$$

We split our studies into the following three scenarios:

1) if $\lambda \in \left[0, \frac{K-1}{K+1}\right]$, $\mathcal{L}_\theta^{\text{nat}} + \lambda \cdot \mathcal{L}_\theta^{\text{fair}}$ first decreases over $[\theta_{\text{f}}, \theta_{\text{f}}(\lambda)]$ and then increases over $[\theta_{\text{f}}(\lambda), \mu_+]$ where its minimum takes place at

$$\mu_- \frac{\mu_+ - \mu_-}{K^2 - 1} + \frac{K}{K^2 - 1} \sqrt{2(K^2 - 1) \ln \left(\frac{1 - \lambda}{1 + \lambda} \cdot K \right) + (\mu_+ - \mu_-)^2} \,; \tag{6.29}$$

2) if $\lambda \in \left(\frac{K-1}{K+1}, 1\right]$, the partial derivative in Equation (6.28) turns out to be increasing in θ with its value at θ_{f} non-negative. It implies that the partial derivative is always non-negative and, thus, the function $\mathcal{L}_\theta^{\text{nat}} + \lambda \cdot \mathcal{L}_\theta^{\text{fair}}$ increases over $[\theta_{\text{f}}, \mu_+]$. As a consequence, the minimizer $\theta_{\text{f}}(\lambda)$ actually coincides with θ_{f}.

3) if $\lambda \in (1, +\infty)$, note that the partial derivative is always positive. Following the same reasoning in the previous scenario, we figure out that the minimizer $\theta_{\text{f}}(\lambda)$ is identical to θ_{f}.

Since the function $\lambda \mapsto \frac{1-\lambda}{1+\lambda}$ is decreasing over $\left[0, \frac{K-1}{K+1}\right]$, by Equation (6.29), we can argue that $\theta_{\text{f}}(\lambda)$ is decreasing in λ over $\mathbb{R}_+$ as well. Furthermore, by the proof of Theorem 6.2, the average distance to the decision boundary is an increasing function in θ over $[\theta_{\text{f}}, \mu_+]$, which indicates that the average distance associated with the classifier $f_{\theta_{\text{f}}(\lambda)}$ decreases, as λ increases. □

subsubsectionFairness and Robustness Models

6.10.1.1 Fair models

The main text mainly discussed penalty-based methods as a way to encourage accuracy parity in a classifier. This subsectiondiscusses a second methodology to achieve fairness denoted *group-loss focused* methods [76]. We will show that the main conclusion of this chapter (e.g., that fairness increase adversarial vulnerability) holds regardless of the methodology adopted to achieve fairness.

Group-loss focused methods. Methods in this category force the training to focus on the loss component of worst performing groups. An effective method to achieve this goal was proposed

in [76]:

$$\theta_{\mathrm{f}} = \underset{\theta}{\operatorname{argmin}} \sum_{a \in \mathcal{A}} \frac{1}{q+1} \mathcal{L}_{\theta}(D_a)^{q+1}, \tag{6.30}$$

where q is a non-negative constant. The intuition behind powering the loss by positive number $q+1$ is to penalize more the classes that have the larger losses. Thus, q plays the role of the fairness parameter, like λ in penalty-based methods: larger q or λ values are associated with fairer (but also often less accurate) models. The main differences between penalty-based methods and group-loss focused methods are the following. First, the loss function of group-loss focused methods is fully differentiable, in contrast to that of penalty-based methods, which is sub-differentiable when the group loss equals the population loss. Second, penalty-based methods try to equalize the losses across various subgroup, while group-focused based methods attempt at minimizing the maximum loss across all subgroups.

6.10.2 Datasets and settings

Datasets. Experiments were performed using three benchmark datasets: UTK Face [144], CIFAR-10 [69], and Fashion MNIST (FMNIST) [134].

1. UTK Face [144]. It consists of more than 20,000 facial images of 48x48 pixels resolution. The experiments consider two learning tasks: (1) The first splits the data into five ethnicities: White, Black, Asian, Indian, and Others. (2) The second splits the data into nine age bins: under-ten years old, 10-14, 15-19, 20-24, 25-29, 30-39, 40-49, 50-59, and over 59 years old. The classes are not uniformly distributed per number of groups and do not contain the same number of images in each group. An 80/20 train-test split is performed.

2. CIFAR-10 [69]. It consists of 60,000 32x32 coloured images belonging to 10 classes, with 6000 images per class. The training set has 50,000 images while the test set has 10,000 images.

3. Fashion MNIST(FMNIST) [134]. It consists of 60,000 28x28 gray-colored images belonging to 10 classes, with 6000 images per class. The training set has 50,000 images while the test set has 10,000 images.

Network architectures. The experiments consider three network architectures of increasing complexity:

1. A CNN consisting of 3 convolutions layers followed by 3 fully connected layers.
2. A ResNet 50 network [56] (over 23 million parameteres).
3. A VGG-13 network [115] (138 million parameters).

We use CNN to provide some examples in the main text, but in this Section we will focus on the ResNet and VGG networks.

Fair models and parameter settings. For penalty-based methods the experiments vary the range of fairness parameter $\lambda \in [0, 2]$. We note that larger λ values may have a detrimental effects to the models accuracy, which, we believe, limit the applicability of the fairness methods in real use cases, thus we focus on these more realistic scenarios. However, we also note that appropriate choices of hyperparameters λ will necessarily depend on the task and architecture at hand and can be used to balance the trade-off between accuracy and fairness.

For group-loss focused methods, the experiments consider $q \in [0, 2]$ for similar reasons as those stated above.

The set of hyper-parameters, learning rate (lr), batch-size (bs), and number of training epochs (epochs) adopted, for each dataset and architecture is reported in Table 6.1.

Dataset	lr	bs	epochs
UTK Face	1e-3	32	70
CIFAR-10	1e-2	32	200
FMNIST	1e-2	32	50

Table 6.1: Hyperparameters settings for each datset.

For each setting, the experiments report the average results of 10 runs, each initializing the models parameters using a different random seed.

Adversarial attacks. The experiments also consider two classes of adversarial attacks to test the model robustness: (1) The l_∞ RFGSM attacks [122] and (2) The l_2 PGD attacks [80]. The experiments adopt the implementations reported in the Python package *torchattacks* [67].

Code and Preprocessing. Follow standard setting, the range of the pixel values was normalized in $[0, 1]$ for all datasets adopted.

All codes were written in Python 3.7 and in Pytorch 1.5.0. The library *torchattacks* [67] was adopted to generate different adversarial attacks. The repository contained the dataset and implementation will be released publicly upon chapter acceptance.

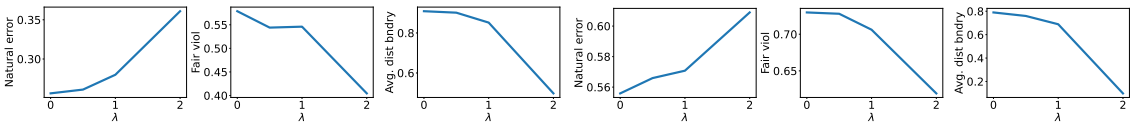
6.10.3 Additional experiments

This subsectiondescribes additional experiments to further support the claims reported in the main chapter. In particular, the experiments report results for deeper networks (e.g., ResNet-50) and for additional fairness methods and adversarial attacks.

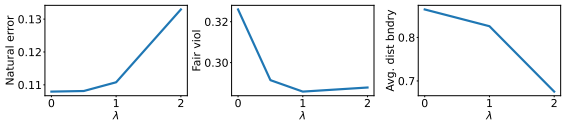
6.10.3.1 Fairness impacts on the decision boundary

Recall, from Theorem 6.3, that fairness reduces the average distance of the testing samples to the decision boundary. As a consequence the fair classifiers are more vulnerable against the adversarial attacks than the natural (unfair) classifiers. This subsection provides additional evidence to support this claim on a high-dimensional model (ResNet 50) and using both penalty based methods additional and group-loss focused methods (see subsection 6.10.1.1) to derive a fair classifier.

Penalty-based methods. Figure 6.9 and Figure 6.10 summarize the results obtained by a penalty-based fair model executed on different benchmark datasets. The experiments again report a consistent trend: as more fairness is enforced (increasing the values λ), the natural errors

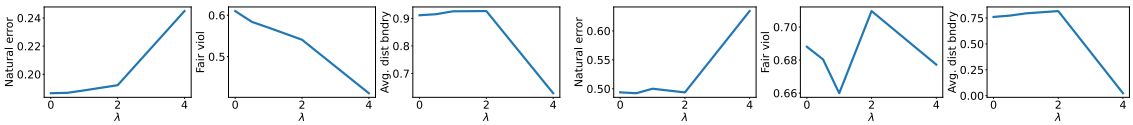


(a) UTKFace ethnicity dataset (b) UTKFace age bins dataset



(c) FMNIST dataset

Fig. 6.9: Natural errors (left), fairness violations (middle), and average distance to the decision boundary (right) for different datasets when varying the fairness parameter λ of **penalty-based methods** on ResNet-50 networks



(a) UTKFace ethnicity dataset (b) UTKFace age bins dataset

Fig. 6.10: Natural errors (left), fairness violations (middle), and average distance to the decision boundary (right) for different datasets when varying the fairness parameter λ of **penalty-based methods** on VGG 13 networks

(left plots) generally increase while both the fairness violations (middle plots) and the average distance to the decision boundary (right plots) decrease. Recall that the latter is a proxy for measuring the model robustness: the closer are samples to the decision boundary the less robust is a model. Thus the previous plots show that robustness decreases as fairness increase.

Group-loss focused methods A similar setting is reported for a model satisfying fairness using the group-loss focused method described in subsection 6.10.1.1. This method maximizes the worst group accuracy, which, in turn, attempts at equalizing the accuracy across groups. Figure 6.11 reports the results obtained using VGG-13 and the UTK-Face dataset. The results again illustrate similar trends: as the fairness parameter q increases, the natural errors (right) tend to increase while both the fairness violations (middle) and the average distance to the decision boundary (left) decrease. Notice that small enough q-values may also act as a regularizer and have a beneficial effect toward the natural error, as observed for the UTK-age bin task (bottom-left plot).

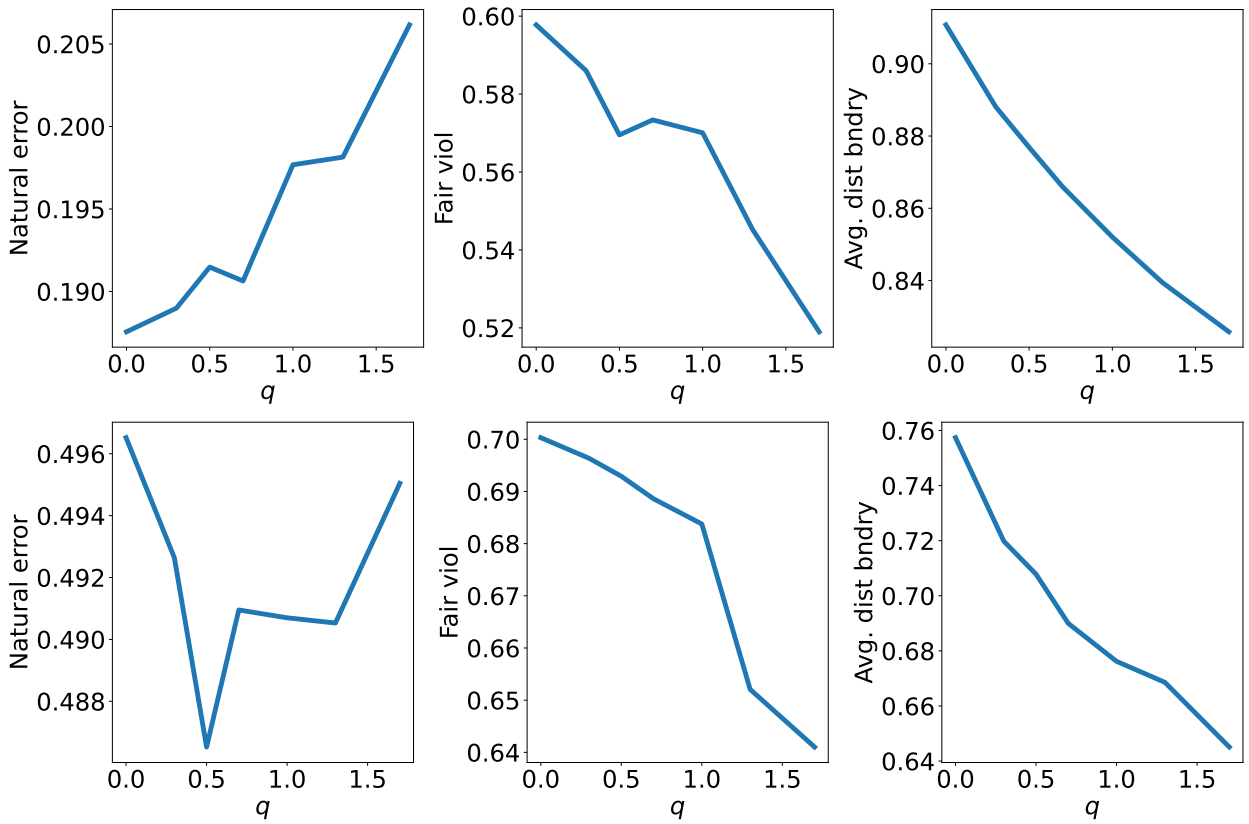


Fig. 6.11: Natural errors, fairness violations, and average distance to the decision boundary for the UTK-Face *ethnicity* (top), UTK-Face *age bins* (bottom) datasets when varying the fairness parameter q of **group-loss focused methods** on VGG-13 networks.

6.10.3.2 Boundary errors increase as fairness decreases

This subsection provides additional experiments to illustrating that the result of the chapter (*fairness increases the adversarial vulnerability*) is invarant across different fair classifier implementations. The experiments adopt the group-loss focused methods for different values of the fairness parameter q. Notice that a natural classifier is obtained when $q = 0$. Figure 6.12 displays the natural (left) and boundary (center) errors attained under different level of RFGSM attacks (regulated by parameter ϵ) and the fairness violation (right) on CIFAR (top) FMNIST (middle) and UTK (bottom) datasets. Notice how increasing q to large enough values typically decreases the fairness violations. However, this comes at the cost of increasing the natural error and the boundary errors,

which, in turn, exacerbate the robust errors.

6.10.3.3 A Mitigating Solution with Bounded Losses

More intuition why bounded loss works. This subsection first provides more intuition about why bounded loss functions, such as the ramp loss adopted in the main text, can help reducing the impacts of fairness towards robustness. To guide the intuitions, we will refer to Figure 6.13, which plots the graph functions of the following loss function for binary classification tasks:

- Ramp loss [30, 50], defined by:

$$\ell(f_\theta(X), Y) = \min(1, \max(0, 1 - Y f_\theta(X))).$$

- Log-loss [55], defined by:

$$\ell(f_\theta(X), Y) = \log(1 + \exp(-Y f_\theta(X))).$$

- Exponential loss [55], defined by:

$$\ell(f_\theta(X), Y) = \exp(-Y f_\theta(X))).$$

Notice that, as illustrated in Figure 6.13, unbounded losses (such as log-loss or exponential loss) amplify the classification errors of misclassified samples X, in a way proportionately to the distance of X from the decision boundary. The misclassification of a sample is captured by the expression $f_\theta(X)Y < 0$, while the distance to the decision boundary by expression $|f_\theta(X)Y|$ (x-axis). Notably these losses are unbounded.

Now notice that a consequence of training a fair classifier is to push its decision boundary to dense region of the advantaged group. This is because such classifier attempts at aligning the groups classification losses, i.e., $\mathbb{E}[\ell(f_\theta(X), Y)|Y = -1] = \mathbb{E}[\ell(f_\theta(X), Y)|Y = 1]$. When the

decision boundary is moved closer to the input samples, the classifier will inevitably become less robust to small perturbations of adversarial noise.

On the contrary, using a bounded loss function, such as ramp loss, during fair learning, can greatly reduce the impact produced by such (outlier) samples.

Effectiveness of the mitigation solution. Next, this subsection provides additional experiments to demonstrate the effectiveness of the proposed solution to find a good tradeoff between fairness and robustness. The generality of the proposed solution is demonstrated across several architectures (VGG-13 and ResNet 50) and adversarial attacks (ℓ_∞ RFGSM and ℓ_2 PGD attacks under different level of attacks ϵ).

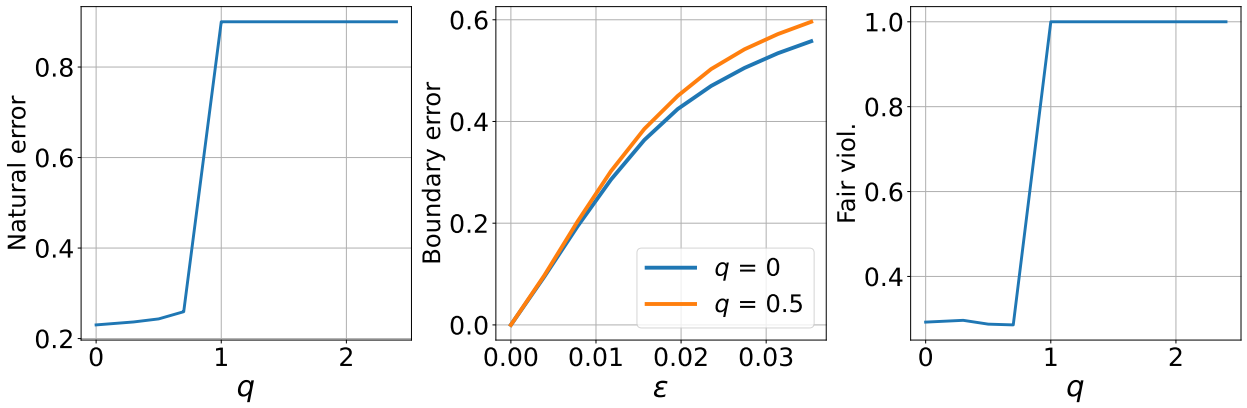
In summary, the proposed mitigation solution—which uses a bounded loss— result in classifiers that, in the vast majority of the cases, are fairer and more robust that those produced by models using a standard (cross entropy) loss.

VGG-13 and ℓ_∞ RFGSM attacks. Figures 6.14 and 6.15 report the boundary errors attained using RFGSM attacks on fair ($\lambda > 0$) and regular ($\lambda = 0$) classifiers on CIFAR and UTK datasets, respectively and using a VGG 13. The plots compare models obtained using a cross entropy loss (top plots) and those using a Ramp loss (bottom plots).

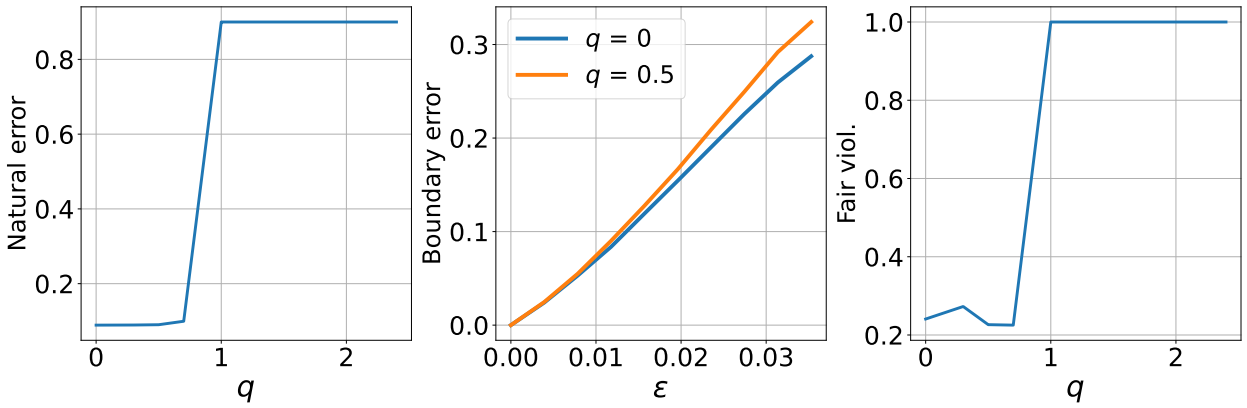
VGG-13 and ℓ_2 PGD attacks. Figures 6.16 and 6.17. report the boundary errors attained using a PGD attacks on fair ($\lambda > 0$) and regular ($\lambda = 0$) classifiers on UTK datasets and using a VGG 13. Once again, the plots compare models obtained using a cross entropy loss (top plots) and those using a Ramp loss (bottom plots).

ResNet 50 and ℓ_∞ RFGSM attacks Figures 6.18 and 6.19. report the boundary errors attained using a RFGSM attacks on fair ($\lambda > 0$) and regular ($\lambda = 0$) classifiers on UTK Face and FMNIST datasets, respectively and using a ResNet50. Once again, the plots compare models obtained using a cross entropy loss (top plots) and those using a Ramp loss (bottom plots).

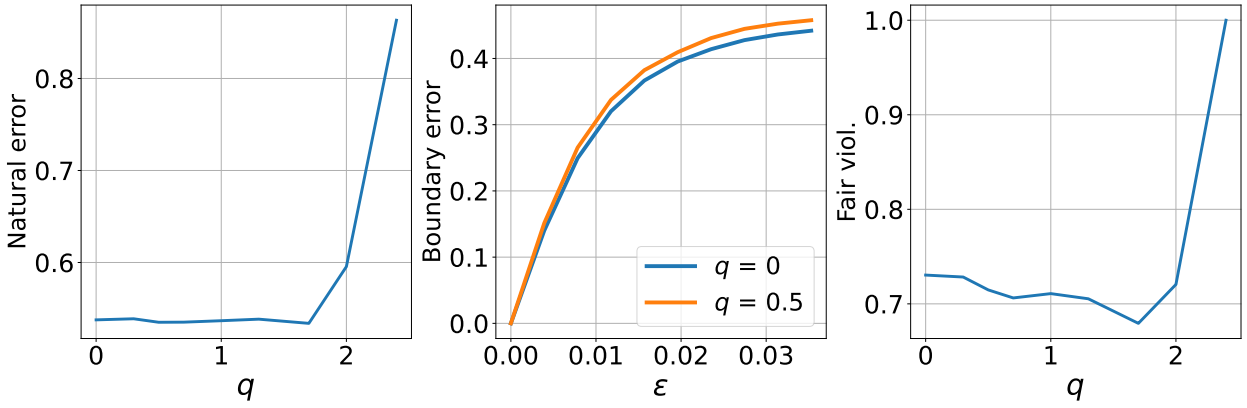
ResNet 50 and ℓ_2 PGD attacks Figures 6.20 and 6.21 report the boundary errors attained using a PGD attack on fair ($\lambda > 0$) and regular ($\lambda = 0$) classifiers on CIFAR10 and FMNIST datasets, respectively, and using a ResNet50. Once again, the plots compare models obtained using a cross entropy loss (top plots) and those using a Ramp loss (bottom plots).



(a) CIFAR dataset



(b) FMNIST dataset



(c) UTK age bins dataset

Fig. 6.12: Natural errors (left), boundary error under different RFGSM attacks (middle), and fairness violation (right) of **group-focused methods** on ResNet-50 networks

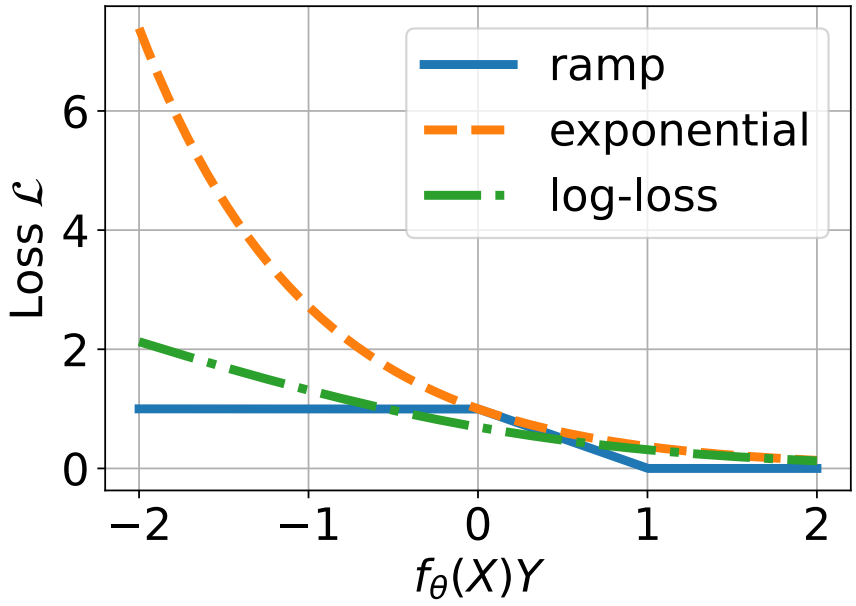


Fig. 6.13: Illustration of different loss functions

.

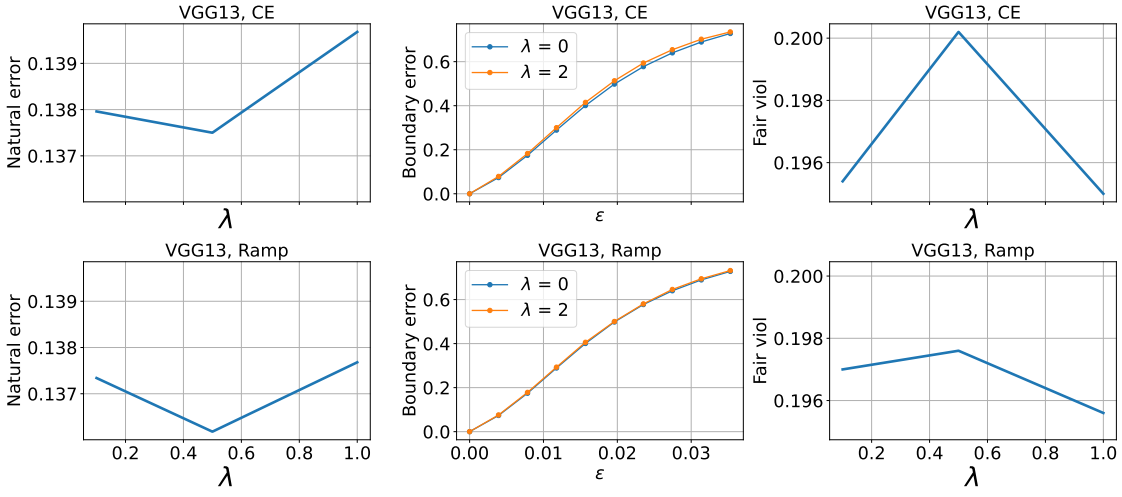


Fig. 6.14: **Top**: Natural errors (left) and fairness violations (right) on the CIFAR-10 *ethnicity* task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different RFGSM attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss. The base classifiers are VGG-13.

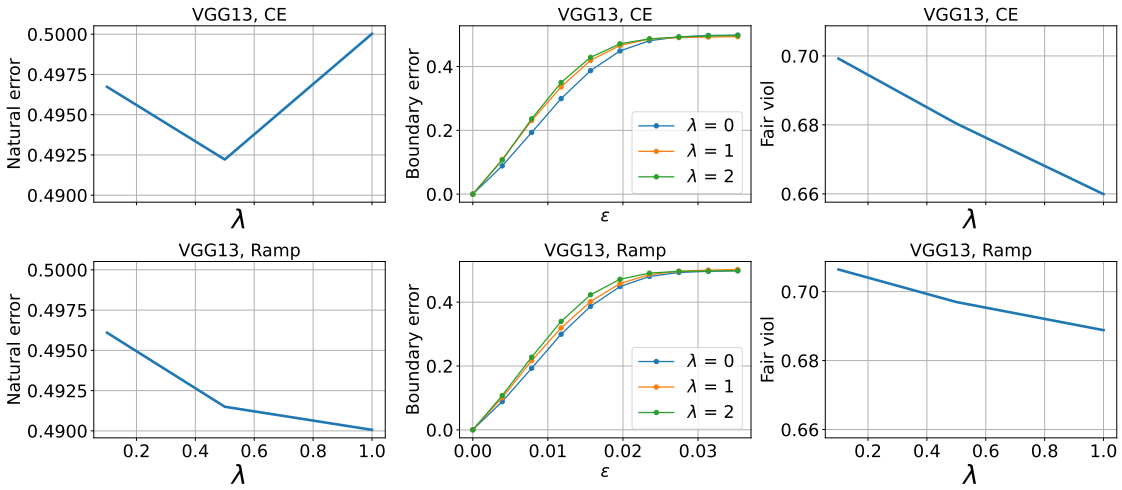


Fig. 6.15: **Top**: Natural errors (left) and fairness violations (right) on the UTKFace *age bins* task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different RFGSM attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss. The base classifiers are VGG-13.

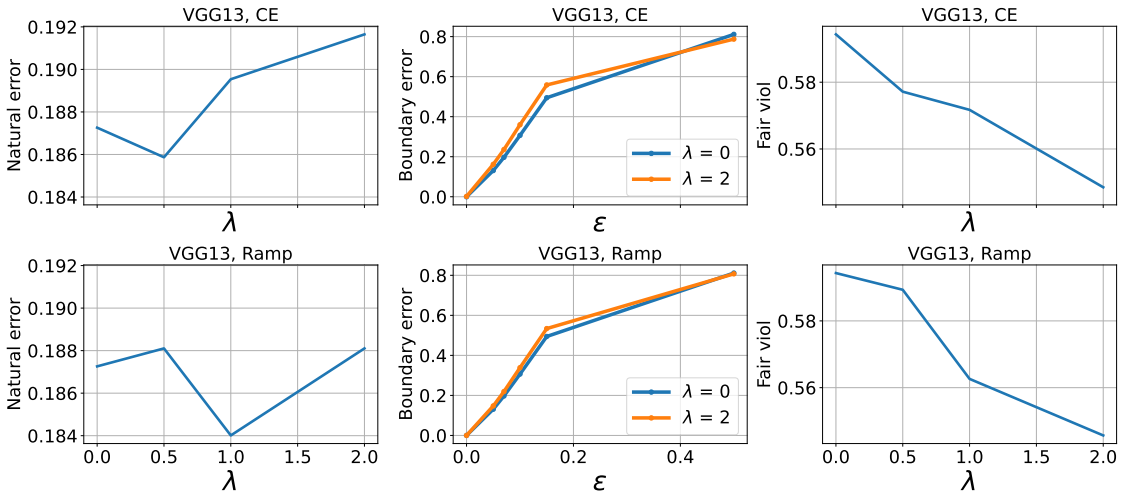


Fig. 6.16: **Top**: Natural errors (left) and fairness violations (right) on the UTKFace *ethnicity* task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different l_2 PGD attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss. The base classifier are VGG-13.

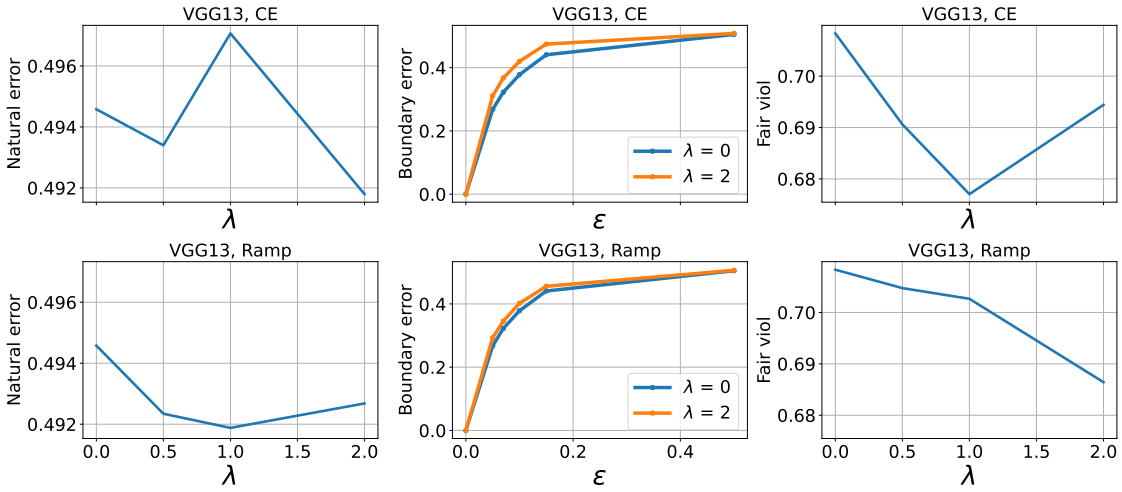


Fig. 6.17: **Top**: Natural errors (left) and fairness violations (right) on the UTKFace *age bins* task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different l_2 PGD attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss. The base classifiers are VGG-13.

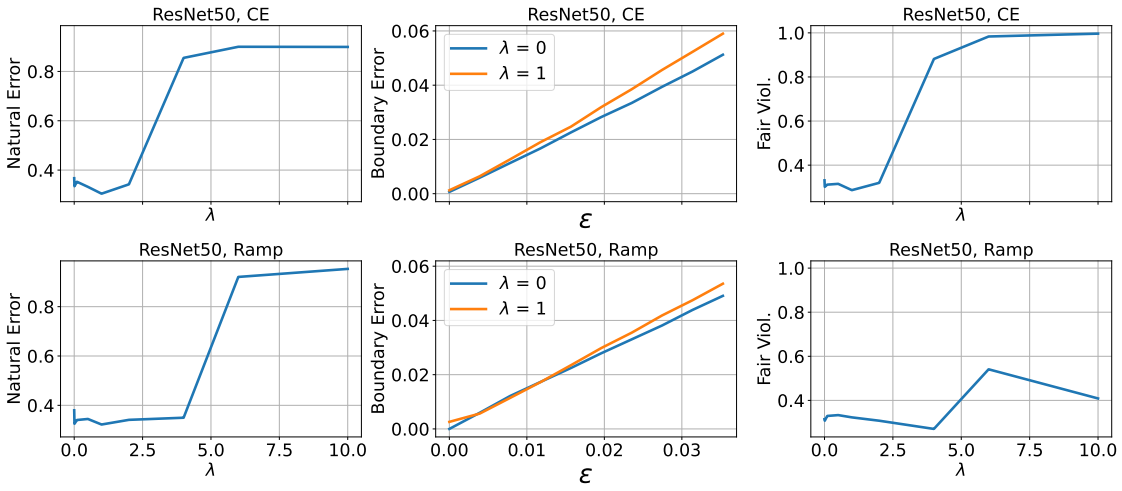


Fig. 6.18: **Top**: Natural errors (left) and fairness violations (right) on the UTKFace *ethnicity* task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different l_∞ RFGSM attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss. The base classifier are Res Net 50.

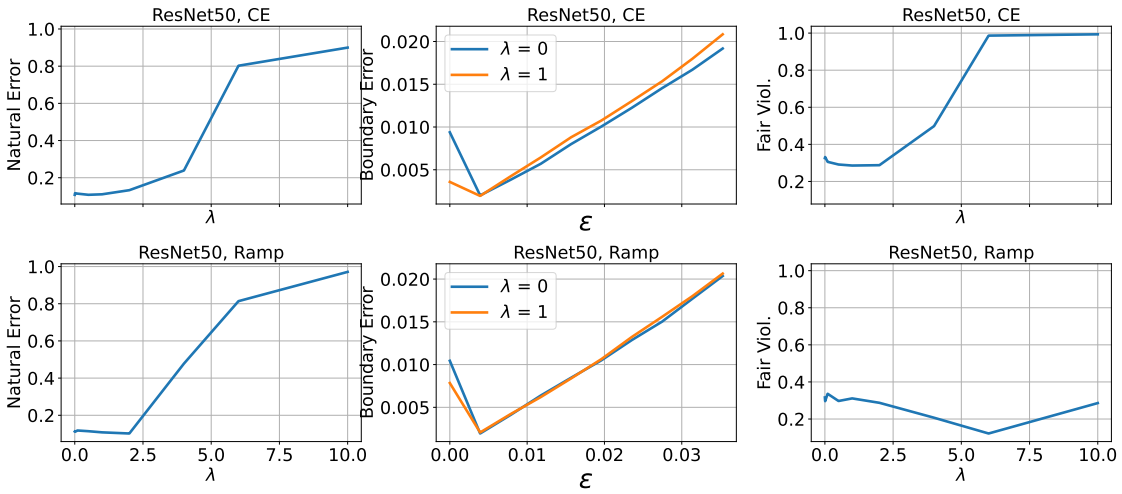


Fig. 6.19: **Top**: Natural errors (left) and fairness violations (right) on the FMNIST task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different l_∞ RFGSM attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss. The base classifier are Res Net 50.

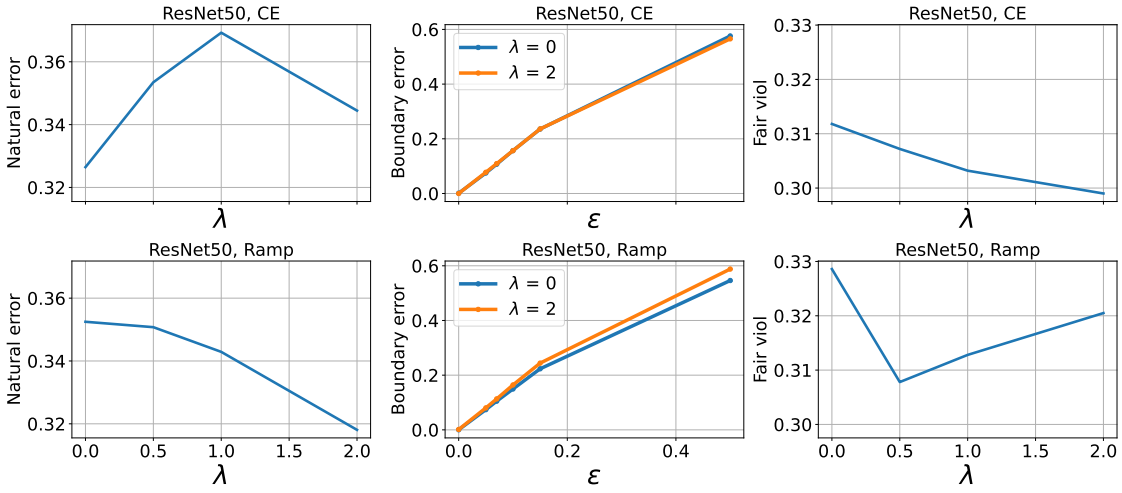


Fig. 6.20: **Top**: Natural errors (left) and fairness violations (right) on the CIFAR 10 task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different l_2 PGD attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss. The base classifiers are ResNet 50.

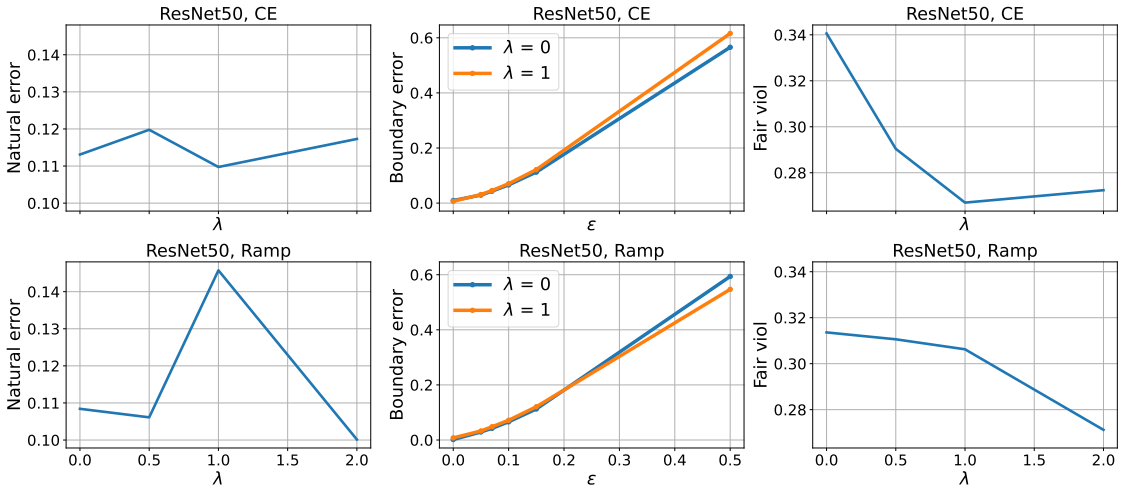


Fig. 6.21: **Top**: Natural errors (left) and fairness violations (right) on the FMNIST task at varying of the fairness parameters λ. The middle plots compares the robustness of fair ($\lambda > 0$) vs. natural ($\lambda = 0$) classifiers to different l_2 PGD attack levels. **Bottom**: Mitigating solution using the bounded Ramp loss. The base classifiers are ResNet 50.

CHAPTER 7
PERSONALIZED PRIVACY AUDITING AND OPTIMIZATION AT TEST TIME

A number of learning models used in consequential domains, such as to assist in legal, banking, hiring, and healthcare decisions, make use of potentially sensitive users' information to carry out inference. Further, the complete set of features is typically required to perform inference. This not only poses severe privacy risks for the individuals using the learning systems, but also requires companies and organizations massive human efforts to verify the correctness of the released information.

This chapter asks whether it is necessary to require *all* input features for a model to return accurate predictions at test time and shows that, under a personalized setting, each individual may need to release only a small subset of these features without impacting the final decisions. The chapter also provides an efficient sequential algorithm that chooses which attributes should be provided by each individual. Evaluation over several learning tasks shows that individuals may be able to report as little as 10% of their information to ensure the same level of accuracy of a model that uses the complete users' information.

7.1 Introduction

The remarkable success of learning models also brought with it pressing challenges at the interface of privacy and decision-making. Privacy, in particular, has been cited as one of the most pressing challenges of modern machine learning systems [95]. The requirement to protect personally identifiable information is especially important as machine learning pipelines become routinely adopted to guide consequential decisions, such as to assist in legal processes, banking, hiring, and healthcare decisions.

To contrast this challenge, several privacy-enhancing technologies have been proposed in the last decades. Among these *Differential Privacy* [37] has found its place as a strong and rigorous privacy notion, largely considered as the de-facto standard mechanism to protect sensitive users data in statistical data analysis with notable adoption by the US. Census Bureau [5], Google [42] and Apple [32].

While this framework has desirable properties its development has been focused on protecting the information contained in the training data, leaving thus possible exposure to the information being revealed during deployment by the users adopting the system. Further, to perform inference, each user is conventionally required to reveal the *complete* set of features describing its data, even if they may not be *all* essential to infer the intended prediction. This not only poses severe privacy risks for the individuals using the learning systems but also requires companies and organizations massive human efforts to verify the correctness of the released information. Importantly, this setting may also violate the EU General Data Protection Regulation in the principle called *data minimization*, which is cited as: "Personal data shall be adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed" [101, 103].

This chapter challenges this setting and asks whether it is necessary to require *all* input features for a model to return accurate or approximately accurate predictions at test time. We refer to this question as the *redundant information leakage release for inference* problem.

This unique question has profound implications for privacy in model personalization, where users are required to reveal large amounts of data. We show that, under a personalized setting,

each individual may need to release only a small subset of their features to produce the *same* prediction errors as those obtained when all features are available. Following this result, we also provide an efficient sequential algorithm that selects the smallest set of attributes to reveal by each individual. Evaluation over several learning tasks shows that individuals may be able to report as little as 10% of their information to ensure the same level of accuracy of a model that uses the complete users' information.

Contributions. In summary, the chapter makes the following contributions: **(1)** it initiates a study to analyze which subset of data features should be released by each individual at deployment time, to induce a model having the same level of accuracy as if all features were released; **(2)** it links this analysis to a new concept of *redundant information leakage* and privacy, **(3)** it proposes theoretically motivated and efficient algorithms that choose which attributes should be provided by each individual to minimize redundant information leakage, and **(4)** it conducts a comprehensive evaluation illustrating that individuals may be able to report as little as 10% of their information to ensure the same level of accuracy of a model that uses the complete users' information.

To the best of our knowledge, this is the first work studying this connection between privacy and accuracy at test time.

7.2 Related work

While we are not aware of studies on redundant information release for inference problems, we draw connections with differential privacy, feature selection, and active learning.

Differential Privacy. Differential Privacy (DP) [37] is a strong privacy notion which determines and bounds the risk of disclosing sensitive information of individuals participating into a computation. In the context of machine learning, DP ensures that algorithms can learn the relations between data and predictions while preventing them from memorizing sensitive information about any specific individual in the training data. In such a context, DP is primarily adopted to protect training data [3, 28, 135] and thus the setting contrasts with that studied in this work, which

focuses on identifying the superfluous features revealed by users at *test time* to attain high accuracy. Furthermore, achieving tight constraints in differential privacy often comes at the cost of sacrificing accuracy, while the proposed privacy framework can reduce privacy loss without sacrificing accuracy under the assumption of linear classifiers.

Feature selection. Feature selection [25] is the process of identifying and selecting a relevant subset of features from a larger set for use in model construction, with the goal of improving performance by reducing complexity and dimensionality of the data. The problem studied in this work can be considered as a specialized form of feature selection with the added consideration of personalized levels, where each individual may use a different subset of features. This contrasts standard feature selection [75], which select the same subset of features for each data sample. Additionally, and unlike traditional feature selection, which is performed during training and independent of the deployed classifier [25], the proposed framework performs feature selection at deployment time and is inherently dependent on the deployed classifier.

Active learning. Finally. the proposed framework shares similarities with active learning [47, 112], whose goal is to iteratively select samples for experts to label in order to construct an accurate classifier with the least number of labeled samples. Similarly, the proposed framework iteratively asks individuals to reveal one attribute given their released features so far, with the goal of minimizing the uncertainty in model predictions.

Despite these similarities, the proposed redundant information leakage concept is motivated by a privacy need and pertains to the analysis of features to release to induce the same level of accuracy as if all features were released.

7.3 Settings and objectives

We consider a dataset D consisting of samples (x, y) drawn from an unknown distribution Π. Here, x is a feature vector with $x \in \mathcal{X}$, and $y \in \mathcal{Y} = [L]$ is a label with L classes. The features in x can be divided into two categories: *public* x_P and *sensitive* features x_S. The sets of public

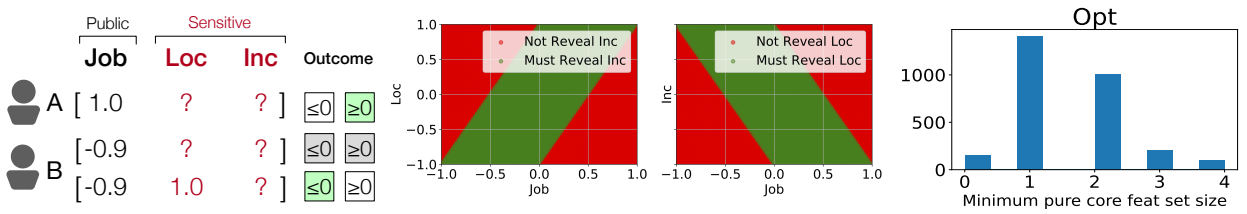


Fig. 7.1: Left: Motivating example. Middle: Feature spaces illustrate the need for users to reveal their sensitive values based on their public values. Right: Frequency associated with the size of the **minimum** pure core feature set in the Credit card dataset under a logistic regression classifier.

and sensitive features indexes in vector x are represented as P and S, respectively.We consider classifiers $f_\theta : \mathcal{X} \to \mathcal{Y}$, which are trained on a public dataset from the same data distribution Π above. The classifier produces a score over the classes, $\tilde{f}_\theta(x) \in \mathbb{R}^L$, and a final output class, $f_\theta(x) \in [L]$, given input x. The model's outputs $f_\theta(x)$ and $\tilde{f}_\theta(x)$ are also often referred to as hard and soft predictions, respectively.

Without loss of generality, we assume that all features in $\mathcal{X}$ are in the range of $[-1, 1]$. In this setting, we are given a trained model f_θ and, at prediction time, we have access to the public features x_P. These features may be revealed in response to a user query or may have been collected by the provider in a previous interaction. For the purpose of illustration, in the scope of the chapter we consider the binary classification, where $L = \{0, 1\}$ and $\tilde{f}_\theta \in \mathbb{R}$. We refer to the Section 7.9 for the multi-class settings where $L > 2$.

In this chapter, the term *redundant information leakage* of a model, refers to the number of sensitive features that are revealed unnecessarily, meaning that their exclusion would not significantly impact the model's output. *Our goal is to design algorithms that accurately predict the output of the model using the smallest possible number of sensitive features, thus minimizing the data leakage at test time*. This objective reflects our desire for privacy.

Before delving into the details of the chapter, we provide an example to serve as motivation for several key points discussed throughout the document

Consider the illustration in Figure 7.1 (left). It exemplifies a loan approval task in which individual features are represented by the set $\{$*Job*, *Loc(action)*, *Inc(ome)*$\}$. The example assumes that the feature *Job* is the public feature x_P while *Loc* and *Inc* are sensitive features x_S. The example

also considers a trained linear model $f_\theta = 1.0\,\textit{Job} - 0.5\,\textit{Loc} + 0.5\,\textit{Inc} \geq 0.$ and looks at a scenario in which a user (A) has a public feature $\textit{Job} = 1.0$ and a user (B) has a public feature $\textit{Job} = -0.9$. Both users have sensitive feature values that are not known. However, notice how, for user A, the outcome can be determined with certainty even if they do not reveal any additional information; No matter the realizations for the sensitive features of A, their outcome will be unaltered, as all features are bounded in $[-1, 1]$. For user B, in contrast, the outcome cannot be determined with certainty based on the public feature alone. But the release of sensitive feature $\textit{Loc} = 1.0$ is sufficient to determine, with certainty, the classifier outcome.

Figure 7.1 (middle) further illustrates the values of the sensitive features *Loc* and *Inc* in relation to the public feature *Job* which allows the classifiers' output to be determined without revealing additional information.

This example highlights two important observations that motivate our study: (1) not all sensitive attributes may be required for decision-making at the time of inference, and (2) the number of relevant sensitive attributes that need to be revealed to make a decision may differ among individuals.

7.4 Core feature sets

With these ideas in mind, this section introduces the concept of core feature set and its relationship with the uncertainty of the model predictions. We discuss the main results in the chapter and report all proofs in Section 7.9.1.

Throughout the chapter, we use R and U to denote, respectively, the set of all revealed and unrevealed features indices of the sensitive features S. Given a vector x and an index set I, we denote x_I as the vector of entries indexed by I and X_I as the associated random variable.

Finally, we write $f_\theta(X_U, X_R = x_R)$ as a shorthand for $f_\theta(X_U, X_R = x_R, X_P = x_P)$ to denote the prediction made by the model when the features in U are unrevealed.

We aim to create algorithms that can identify the smallest set of sensitive attributes to reveal

to render the model's output certain (with high probability) regardless of the unrevealed attributes' values. Such a set is denoted *core feature set*.

Definition 7.1 (Core feature set)**.** *Consider a subset R of sensitive features S, and let $U = S \setminus R$ be the unrevealed features. The set R is a core feature set if, for some $\tilde{y} \in \mathcal{Y}$,*

$$\Pr\left(f_\theta(X_U, X_R = x_R) = \tilde{y}\right) \geq 1 - \delta, \tag{7.1}$$

where $\delta \in [0, 1]$ is a failure probability.

When $\delta = 0$ the core feature set is called **pure**. Additionally, the label $\tilde{y}$ satisfying Equation (7.1) is called the *representative label* for the core feature set R. The concept of the representative label $\tilde{y}$ is crucial for the algorithms that will be discussed later. These algorithms use limited information to make predictions. When predictions are made using a set of unrevealed features, the representative label $\tilde{y}$ will be used in place of the model's prediction.

The following is a useful property of core feature sets used by this work to minimize redundant information leakage.

Proposition 7.1. *Let $R \subseteq S$ be a core feature set with failure probability $\delta < 0.5$. Then, there exists a monotonic decreasing function $\epsilon : \mathbb{R} \to \mathbb{R}$ with $\epsilon(0) = 0$ such that:*

$$H\left[f_\theta(X_U, X_R = x_R)\right] \leq \epsilon(\delta),$$

where $H[Z] = -\sum_{z \in [L]} \Pr(Z = z) \log \Pr(Z = z)$ is the entropy of the random variable Z.

This property highlights the relationship between core feature sets and entropy associated with the model that uses incomplete information. Smaller δ values result in less uncertainty in the model's predictions and when δ is equal to zero (or when R is a pure core feature set), we have complete knowledge of the model's predictions even without observing x_U. Thus this property also illustrates the relationship between the failure probability δ and the uncertainty of model predictions.

It is worth noticing that more accurate predictions also require revealing more information, as highlighted in the previous result and the following celebrated information theoretical result.

Proposition 7.2. *Given two subsets R and R' of sensitive features S, with $R \subseteq R'$,*

$$H\big(f_\theta(X_U, X_R = x_R)\big) \geq H\big(f_\theta(X_{U'}, X_{R'} = x_{R'})\big),$$

where $U = S \setminus R$ and $U' = S \setminus R'$.

Thus, the parameter δ plays an important role in balancing the trade-off between the *privacy loss* and the *model performance*. It controls how much sensitive information needs to be revealed to make accurate predictions (for a desired level of uncertainty in the model's predictions). As δ gets larger, less sensitive features need to be revealed, leading to smaller information leakage but also less accurate model predictions, and vice-versa.

Note that, as pointed out in the previous example, the core feature set is not unique for all users. This is also highlighted in Figure 7.1 (right), which illustrates the minimum pure core feature sets computed using a logistic regression classifier on the Credit dataset [18]. The figure shows that many individuals need to release *no* additional information to obtain the model predictions and that most individuals can get accurate model predictions with certainty by releasing just ≤ 2 sensitive features. These connections, together with the previous observations linking core feature sets to entropy motivate the proposed online algorithm.

7.5 Personalized feature release (PFR)

The goal of the proposed algorithm, called Personalized feature release (PFR), is to reveal sensitive features one at a time based on their *released* feature values. This section provides a high-level description of the algorithm and outlines the challenges in some of its aspects. Next, Section 7.6, applies PFR to linear classifiers and discusses its performance on several datasets and benchmarks. Further, Section 7.7, extends PFR to non-linear classifiers and considers an evaluation over a range

of standard datasets. In the subsequent sections, we assume that the input features are jointly distributed as Gaussians with mean vector μ and covariance matrix Σ, unless stated otherwise. Additionally, as our motivation suggests, we will concentrate solely on maintaining privacy at deployment time.

High-level ideas of PFR. At a high level, the algorithm chooses a feature to reveal by inspecting the posterior probabilities $\Pr(X_j|X_R = x_R, X_P = x_P)$ for each unrevealed feature $j \in U$ and with respect to the revealed sensitive features x_R and the public features x_P. Given the current set of features revealed x_R and unrevealed x_U, the algorithm chooses the next feature $j \in U$ such that:

$$\begin{aligned} j &= \operatorname*{argmax}_{j \in U} F(x_R, x_j; \theta) \\ &= \operatorname*{argmax}_{j \in U} -H\big[f_\theta(X_j = x_j, X_{U\setminus\{j\}}, X_R = x_R)\big], \end{aligned} \tag{7.2}$$

where F is a *scoring function* that measures how much information can be gained on the model's predictions if feature X_j is revealed. Upon revealing feature X_j with a value of x_j, the algorithm adjusts the posterior probabilities for all remaining unrevealed features. The process concludes when either all sensitive features have been disclosed or a core feature set has been identified.

The remainder of the section delves into the difficulties of calculating the scoring function F, including the unknown value of X_j beforehand and methods for determining if a set of revealed features constitutes a core feature set.

7.5.1 Computing the scoring function F

The scoring function F quantifies the level of certainty in model predictions when a user reveals the value of feature X_j. There are two challenges to consider. First, the value of X_j is unknown until the decision is made, challenging the computation of the entropy function. Second, even if the value of X_j were known, determining the entropy of model predictions in an efficient manner is a further difficulty. We next discuss how to overcome these challenges.

To address the first challenge, we exploit the information encoded in the revealed features to infer x_j. Thus, we can compute the posterior probability $\Pr(X_j|X_R{=}x_R)$ of the unrevealed feature X_j given the values of the revealed ones. This estimate allows us to modify the scoring function, abbreviated as $F(X_j)$, to be the expected negative entropy given the randomness of X_j.

$$\begin{aligned} F(X_j) &= \mathbb{E}_{X_j} - \big[H[f_\theta(X_j, X_{U\setminus\{j\}}, X_R{=}x_R)\big] \\ &= -\int H\big[f_\theta(X_j{=}z, X_{U\setminus\{j\}}, X_R{=}x_R)\big] && \text{(7.3a)} \\ &\qquad \times \Pr(X_j{=}z|X_R{=}x_R)dz, && \text{(7.3b)} \end{aligned}$$

where $z \in \mathcal{X}_j$ is a value in the support of X_j.

Estimating this scoring function efficiently is however challenged by the presence of two key components. The first (Equation (7.3a)) is the entropy of the model's prediction given a specific unrevealed feature value, $X_j = z$. This prediction is a function of the random variable $X_{U\setminus\{j\}}$, and, due to Proposition 7.1, its estimation is related to the conditional densities $\Pr(X_{U\setminus\{j\}}|X_R = x_R, X_j = z)$. The second component (Equation (7.3b)) is the conditional probability $Pr(X_j = z|X_R = x_R)$. Computing these conditional densities efficiently is discussed next.

The following result relies on the joint Gaussian assumption of the input features and will be useful in providing a computationally efficient method to estimate such conditional density functions. In the following, Σ_{IJ} represents a sub-matrix of size $|I| \times |J|$ of a matrix Σ formed by selecting rows indexed by I and columns indexed by J.

Proposition 7.3. *The conditional distribution of any subset of unrevealed features $U' \in U$, given the the values of released features $X_R = x_R$ is given by:*

$$\begin{aligned} \Pr(X_{U'}|X_R = x_R) = \mathcal{N}\Big(&\mu_{U'} + \Sigma_{U',R}\Sigma_{R,R}^{-1}(x_R - \mu_R), \\ &\Sigma_{U',U'} - \Sigma_{U',R}\Sigma_{R,R}^{-1}\Sigma_{R,U'}\Big), \end{aligned}$$

where Σ is the covariance matrix

To complete Equation 7.3, we must estimate the entropy $H[f_\theta(X_j = z, X_{U\setminus\{j\}}, X_R = x_R)]$ for a specific instance z, drawn from the distribution $\Pr(X_j|X_R = x_R)$ (see Equation (7.3a)). This can be achieved by estimating $\Pr(\tilde{f}_\theta(X_j = z, X_{U\setminus\{j\}}, X_R = x_R))$, as $f_\theta = \mathbf{1}\{\tilde{f}_\theta \geq 0\}$, where $\mathbf{1}$ is the indicator function and in the following sections, we will show how to assess this estimate for linear and non-linear classifiers. Finally, by approximating the distribution over soft model predictions through Monte Carlo sampling, we can compute the score function in $F(X_j)$, as

$$\begin{aligned} F(X_j) &= \mathbb{E}_{X_j} - \left[H\left[f_\theta(X_j, X_{U\setminus\{j\}}, X_R = x_R)\right]\right] \\ &\approx -\sum_{z' \in \boldsymbol{Z}} H\left[f_\theta(X_j = z', X_{U\setminus\{j\}}, X_R = x_R)\right], \end{aligned} \tag{7.4}$$

where $\boldsymbol{Z}$ is a set of random samples drawn from $\Pr(X_j|X_R = x_R)$ and estimated through Proposition 7.3, which thus can be computed efficiently.

7.5.2 Testing a core feature set

As reviewed above, the proposed iterative algorithm stops when it determines whether a subset R of the sensitive feature set S is a core feature set. We divide this verification process into two cases: When $\delta = 0$, verifying that R is a pure core feature set only requires checking if $f_\theta(X_U, X_R = x_R)$ is constant for all realizations of X_U. We demonstrate, in Section 7.6, that this can be accomplished in linear time for linear classifiers without any input distribution assumptions. When $\delta > 0$, such a property is no longer valid. Recall that, in order to verify a core feature set as per Definition 7.1, we need to estimate the distribution of $\Pr(\tilde{f}_\theta(X_U, X_R = x_R))$. In Section 7.6, we show that one can analytically estimate this distribution for linear classifiers, while in Section 7.7 we show how to approximate this distribution locally, and use this estimate to derive a simple, yet effective (in practice), estimator.

7.6 PFR for linear classifiers

This section will devote to estimating the distribution $\Pr(\tilde{f}_\theta(X_j{=}z, X_{U\setminus\{j\}}, X_R{=}x_R))$, or simply expressed as $\Pr(\tilde{f}_\theta(X_U, X_R{=}x_R))$ and provides an instantiation of the PFR algorithm for linear classifiers. In particular, it shows that when the input features are jointly Gaussian, both the estimation of the conditional distributions required to compute the scoring function $F(X_j)$ and the termination condition to test whether a set of revealed features is a core feature set, can be computed efficiently. This is an important property for the developed algorithms, which are considered online and interactive protocols.

7.6.1 Efficiently estimating $\Pr(\tilde{f}_\theta(X_U, X_R = x_R))$

For a linear classifier $\tilde{f}_\theta = \theta^\top x$, and under the Gaussian distribution assumption adopted, the model predictions $\tilde{f}_\theta(x)$ are also Gaussian, as highlighted by the following result.

Proposition 7.4. *The model predictions before thresholding, $\tilde{f}_\theta(X_U, X_R = x_R) = \theta_U X_U + \theta_R x_R$ is a random variable with a Gaussian distribution $\mathcal{N}\big(m_f, \sigma_f^2\big)$, where*

$$m_f = \theta_R x_R + \theta_U^\top \big(\mu_U + \Sigma_{U,R}\Sigma_{R,R}^{-1}(x_R - \mu_R)\big) \tag{7.5}$$

$$\sigma_f^2 = \theta_U^\top \big(\Sigma_{U,U} - \Sigma_{U,R}\Sigma_{R,R}^{-1}\Sigma_{R,U}\big)\theta_U, \tag{7.6}$$

where θ_U is the sub-vector of parameters θ corresponding to the unrevealed features U.

The above result is used to assist in calculating the conditional distribution of model predictions $f_\theta(x)$, following thresholding. This is a random variable that adheres to a Gaussian distribution, as shown next, and will be used to compute the entropy of the model predictions, as well as to determine if a subset of features constitutes a core set.

Proposition 7.5. *Let the model predictions prior thresholding $\tilde{f}_\theta(X_U, X_R = x_R)$ be a random variable following a Gaussian distribution $\mathcal{N}(m_f, \sigma_f^2)$. Then, the model prediction following thresholding $f_\theta(X_U, X_R = x_R)$ is a random variable following a Bernoulli distribution $Bern(p)$ with*

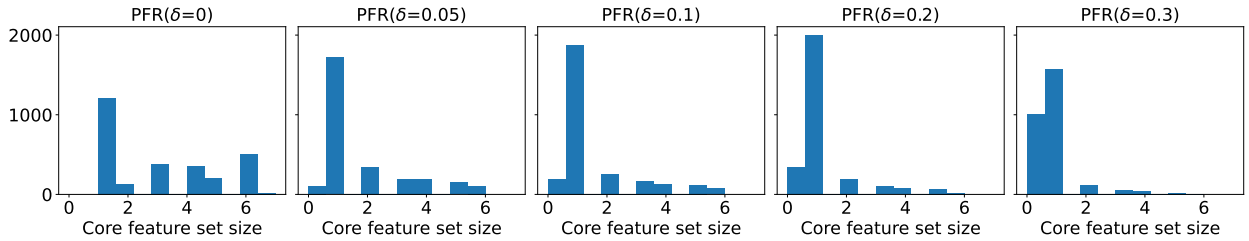


Fig. 7.2: Histogram of core feature set size for PFR under different δ on Bank dataset when $|S| = 7$ and the underlying classifier is Logistic Regression

Algorithm 2: PFR for linear classifiers

input : A test sample x; Training data D
output: A core feature set R and its representative label $\tilde{y}$

$\mu \leftarrow \frac{1}{|D|} \sum_{(x,y)\in D} x$
$\Sigma \leftarrow \frac{1}{|D|} \sum_{(x,y)\in D} (x-\mu)(x-\mu)^\top$
Initialize $R = \emptyset$
while *True* **do**
  if R *is a core feature set with repr. label* $\tilde{y}$ **then**
    return $(R, \tilde{y})$
  else
    foreach $j \in U$ **do**
      Compute $\Pr(X_j | X_R = x_R)$ (using Prop. 7.3)
      $\boldsymbol{Z} \leftarrow$ sample$(\Pr(X_j | X_R = x_R))$ T times
      Compute $\Pr\left(f_\theta(X_j = z, X_{U\setminus\{j\}} X_R = x_R)\right)$ using Prop. 7.4 and 7.5)
      Compute $F(X_j)$ (using Eq. (7.4))
  $j^* \leftarrow \text{argmax}_j F(X_j)$
  $(R, U) \leftarrow R \cup \{j^*\},\ U \setminus \{j^*\}$

$p = \Phi(\frac{m_f}{\sigma_f})$*, where* $\Phi(\cdot)$ *refers to the CDF of the standard normal distribution, and* m_f *and* σ_f*, are given in Equations* (7.5) *and* (7.6)*, respectively.*

7.6.2 Testing pure core feature sets

In this subsection, we outline the methods for determining if a subset U is a pure core feature set, and, if so, identifying its representative label. As per Definition 7.1, U is a pure core feature set if $f_\theta(X_U, X_R = x_R) = \tilde{y}$ for all X_U. Equivalently, $\tilde{f}_\theta(X_U, X_R = x_R) = \theta_U^\top X_U + \theta_R^\top x_R$ must have the same sign (either positive or negative) for all X_U in the range of $[-1, 1]^{|U|}$. Rather than

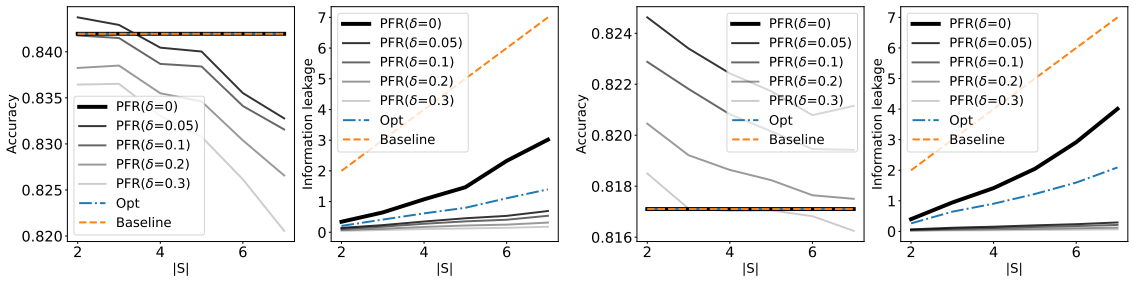


Fig. 7.3: Accuracy and redundant information leakage for different choices of number of sensitive features $|S|$ on Insurance (left) and Credit (right) datasets using a Logistic Regression classifier.

evaluating all possible values of $\tilde{f}_\theta(X_U, X_R = x_R)$, we only need to examine if the maximum and minimum values have the same sign. By virtue of the linear programming property under the box constraint $X_U \in [-1, 1]^{|U|}$, it follows that:

$$\begin{aligned} \max_{X_U} \theta_U^\top X_U + \theta_R^\top x_R &= \|\theta_U\|_1 + \theta_R^\top x_R \\ \min_{X_U} \theta_U^\top X_U + \theta_R^\top x_R &= -\|\theta_U\|_1 + \theta_R^\top x_R. \end{aligned} \tag{7.7}$$

Therefore, if the sum $\|\theta_U\|_1 + \theta_R^\top x_R$ and the difference $-\|\theta_U\|_1 + \theta_R^\top x_R$ are both negative (non-negative), then U is considered a pure core feature set with representative label $\tilde{y} = 0$ ($\tilde{y} = 1$), otherwise U is not a pure core feature set.

Importantly, determining whether a subset R of sensitive features S constitutes a pure core feature set can be accomplished in linear time with respect to the number of features.

Proposition 7.6. *Assume f_θ is a linear classifier. Then, determining if a subset U of sensitive features S is a* pure *core feature set can be performed in $O(|P| + |S|)$ time.*

7.6.3 PFR-linear algorithm and evaluation

A pseudo-code of PFR specialized for linear classifiers is reported in Algorithm 2. The algorithm takes as input a sample x (which only exposes the set of public features x_P) and uses the training data D to estimate the mean and covariance matrix needed to compute the conditional distribution of the model predictions given the unrevealed features (lines 1 and 2), as discussed above. After

initializing empty the set of revealed features to the (line 3) it iteratively releases one feature until a core feature set (and its associated representative label) are determined (line 5), as discussed in detail in Section 7.6.2. The released feature X_{j^*} is the one, among the unrevealed features U, that maximizes the scoring function F (line 13). Computing such a scoring function requires estimating the conditional distribution $\Pr(X_j|X_R = x_R)$ (line 9), constructing a sample set $\boldsymbol{Z}$ from such distribution (line 10), and approximating the distribution over soft model predictions through Monte Carlo sampling to compute (line 11). Finally, the algorithm updates the set of the revealed and unrevealed features (line 14).

Notice that PFR relies on estimating the mean vector and covariance matrix from the training data, which is considered public, for the scope of this chapter. If the training data is private, various techniques exist to release DP mean, and variance [11, 78] and can be readily adopted. However, the protection of training data is beyond the scope of this work.

Evaluation. Next, this section evaluates the effectiveness of PFR in minimizing information leakage. The experiments are conducted on six standard UCI datasets [18]. We discuss here a selection of these results and refer the reader to the Section 7.9 for additional experiments.

Figure 7.2 reports the snapshot on the redundant data leakage subject by various users on a Logistic regression classifier trained on the Bank dataset [18] (more details reported in the Section 7.9), when using the proposed PFR algorithm for various core feature set failure probability δ levels. The benefits of PFR are clearly evident from this histogram. For each testing sample, PFR finds core feature sets that are much smaller than the overall sensitive feature set size $|S| = 7$. Additionally, notice that when $\delta > 0$, it finds core features sets of size smaller than 2 for the vast majority of the individuals. *This suggests that a significant number of users would need to disclose only a small fraction of all of their sensitive information to allow the model to make accurate predictions either with complete certainty or with very high confidence.*

To further illustrate the advantages of PFR, we compare it to a *baseline* and an *optimal* model for various choices of the number of sensitive attributes $|S| \in [2, 7]$. The *baseline*, in this context, refers to the use of the original classifier, which requires users to disclose all sensitive features.

The *optimal* model refers to the process of using a brute force method to identify the minimum core feature set and its representative label by evaluating all possible subsets of sensitive features. Once identified, this representative label is used as the model prediction when not all sensitive features are disclosed. Verification tests are used to determine if a subset is a core feature set. It is important to note that this method is not only computationally inefficient due to the exponential number of cases, but also infeasible to implement in practice as it assumes that all sensitive features are known.

For each choice of $|S|$, we randomly select $|S|$ features from the entire set of features and designate them as sensitive attributes. The remaining features are considered as public attributes. The average accuracy and information leakage are then reported based on 100 random selections of the sensitive attributes. Additional details on the experimental settings can be found in Section 7.9.5.

The performance results in terms of accuracy (left subplots) and information leakage (right subplots) are presented in Figure 7.3. It is observed that across all datasets, PFR with $\delta = 0$ are able to identify a pure core feature set that is much smaller than the set of sensitive features. As a result, only a small percentage of sensitive features need to be disclosed by users, while maintaining the same level of accuracy. Furthermore, PFR with $\delta = 0$, identifying pure core feature sets, can retain the same accuracy as the Baseline models. This implies that under linear models, privacy (as defined in this chapter) can be achieved "for free"!. Additionally, notice that how δ increases, fewer features need to be revealed by users, but at the cost of a decrease in accuracy, generally. Notice also that there may be cases (e.g., right subplots) where such features do not correlate well with the predictions, and not revealing them may thus even improve the prediction accuracy (this aspect is related to feature selection). Generally, however, the larger the failure probability δ the more information leakage can be protected but at a cost of a larger drop in accuracy. At the same time, notice how marginal is the decrease in accuracy, which demonstrates the robustness of the proposed model.

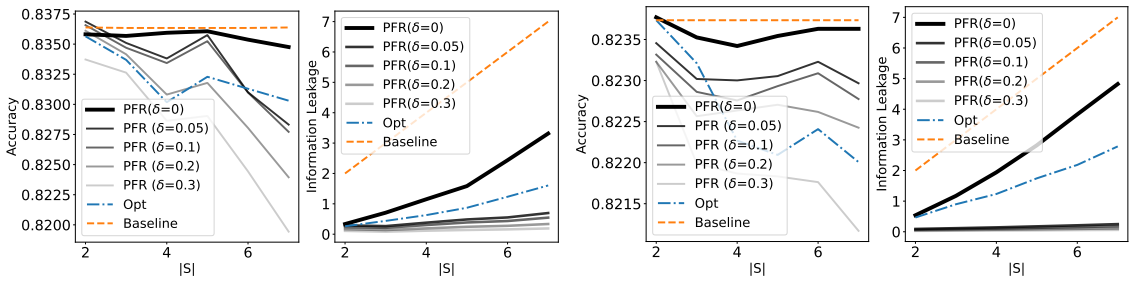


Fig. 7.4: Accuracy and redundant information leakage for different choices of number of sensitive features $|S|$ on Insurance (left) and Credit (right) datasets using a nonlinear (neural network) classifier.

7.7 PFR for non-linear classifiers

Next, the chapter focuses on computing the estimate $\Pr(\tilde{f}_\theta(X_U, X_R = x_R))$ and determining core feature sets when f_θ is a nonlinear classifier. Then, the section presents results that illustrate the practical benefits of PFR in minimizing information leakage on neural networks. The determination of core feature sets relies on the assumption that the classifiers are Δ-robust, i.e., $\forall x, x' \in \mathcal{X}$, s.t: $\|x - x'\|_\infty \leq \Delta$ then $f_\theta(x) = f_\theta(x')$. In practice, however, we show that, even in the presence of arbitrary classifiers, the proposed PFR is able to significantly reduce information leakage at test time.

7.7.1 Efficiently estimating $Pr(\tilde{f}_\theta(X_U, X_R = x_R))$

First notice that even if the input features x are jointly Gaussian, the outputs $f_\theta(x)$ of the classifier will no longer follow a Gaussian distribution after undergoing a non-linear transform. This makes estimating the distribution of $\Pr(\tilde{f}_\theta(X_U, X_R = x_R)$ more challenging. To address this issue, the chapter proposes to locally approximate the model predictions $\tilde{f}\theta$ using a Gaussian distribution. This approach is demonstrated in the following result.

Theorem 7.1. *The distribution of the random variable* $\tilde{f}_\theta = \tilde{f}_\theta(X_U, X_R = x_R)$ *where* $X_U \sim \mathcal{N}\big(\mu_U^{pos}, \Sigma_U^{pos}\big)$ *can be approximated by a Normal distribution as*

$$\tilde{f}_\theta \sim \mathcal{N}\big(\tilde{f}_\theta(X_U = \mu_U^{pos}, X_R = x_R), g_U^\top \Sigma_U^{pos} g_U\big) \tag{7.8}$$

where $g_U = \nabla_{X_U} \tilde{f}_\theta(X_U = \mu_U^{pos}, X_R = x_R)$ *is the gradient of model prediction at* $X_U = \mu_U^{pos}$.

In the above, the mean vector μ_U^{pos} and and covariance matrix Σ_U^{pos} of $Pr(X_U|X_R = x_R)$ are obtained from Proposition 7.3. The result above relies on a first-order Taylor approximation of the classifier f_θ around its mean.

7.7.2 Testing pure core feature sets

To determine if a subset U of the sensitive features S is a pure core feature set, we consider a set of $(\frac{1}{\Delta})^{|U|}$ input points, represented by $Q = [X_U, x_R]$. The entries corresponding to the revealed features are fixed with the value x_R, while the entries corresponding to the unrevealed features are evenly spaced over the cube $[-1, 1]^{|U|}$. The test verifies if the model predictions $f_\theta(x)$ remain constant for all x in Q. Note that the computational runtime of this verification process is affected by the degree of robustness Δ of the underlying classifier f. Rendering such a procedure more generally computationally efficient will be an interesting direction for future work. In the next section, we will show that even considering arbitrary classifiers (e.g., we use standard neural networks), PFR can reduce information leakage dramatically when compared to standard approaches.

7.7.3 PFR-nonlinear algorithm and evaluation

The FPR algorithm for non-linear classifiers differs from Algorithm 2 only in the method of calculating the estimates for the distribution of the soft model predictions, represented by $\Pr(f_\theta(X_j = z, X_{U\setminus j}, X_R = x_R))$ (line 11), by utilizing the results in Theorem 7.2 and Proposition 7.5. Additionally, the algorithm's termination test relies on the discussion presented in the previous section. A complete description of the algorithm is reported in Section 7.9.2.

Evaluation. Next, we assess the performance of PFR in reducing information leakage when standard non-linear classifiers are adopted. Specifically, we use a neural network with two hidden layers and ReLU activation functions as baselines classifiers and train models using stochastic

gradient descent (as specified in more detail in Section 7.9.5). The evaluation, baselines, and benchmarks adopted follow the same settings as those adopted in Section 7.6.3.

Figure 7.4 illustrates the results in terms of accuracy (left subplots) and information leakage (right subplots). Unlike linear classifiers, non-linear models using PFR with a failure probability $\delta = 0$ cannot ensure the same level of accuracy as the baseline models. However, notice how small this difference in accuracy is. Remarkably, a failure probability $\delta = 0$ allows users to release less than a half and up to 90% less sensitive features across different datasets while obtaining accuracies comparable to those of traditional classifiers. Notice also that when more relaxed failure probabilities are considered the information leakage reduces significantly. For example, when $\delta = 0.05$, users require to release only 5% of their sensitive features while retaining comparable accuracies to the baseline models (the largest accuracy difference reported was 0.005%). *These results are significant: They demonstrate that the introduced privacy leakage notion and the proposed algorithm can become an important tool to safeguard the privacy of individual's data at test time, without excessively compromising accuracy*.

7.8 Conclusion

This chapter introduced the concept of information leakage at test time whose goal is to minimize the number of features that individuals need to disclose during model inference while maintaining accurate predictions from the model. The motivations of this notion are grounded in the privacy risks imposed by the adoption of learning models in consequential domains, by the significant efforts required by organizations to verify the accuracy of the released information, and align with the data minimization principle outlined in the GDPR. The chapter then discusses an iterative and personalized algorithm that selects the features each individual should release with the goal of minimizing information leakage while retaining exact (in the case of linear classifiers) or high (for non-linear classifiers) accuracy. Experiments over a range of benchmarks and datasets indicate that individuals may be able to release as little as 10% of their information without compromising

the accuracy of the model, providing a strong argument for the effectiveness of this approach in protecting privacy while preserving the accuracy of the model.

7.9 Appendix

7.9.1 Missing proofs

Proposition 7.7. *Given a core feature set $R \subseteq S$ with failure probability $\delta < 0.5$, then there exists a function $\epsilon : \mathbb{R} \to \mathbb{R}$ that is monotonic decreasing function with $\epsilon(0) = 0$ such that:*

$$H\big[f_\theta(X_U, X_R = x_R)\big] \leq \epsilon(\delta),$$

where $H[Z]$= $-\sum_{z\in[L]} \Pr(Z = z) \log \Pr(Z = z)$ is the entropy of the random variable Z.

Proof. In this proof, we demonstrate the binary classification case. The extension to a multi-class scenario can be achieved through a similar process.

By the definition of the core feature set, there exists a representative label, denoted as $\tilde{y} \in \{0, 1\}$ such that the probability of $P(f_\theta(X_U, X_R = x_R) = \tilde{y})$ is greater than or equal to $1 - \delta$. Without loss of generality, we assume that the representative label is $\tilde{y} = 1$. Therefore, if we denote Z as the probability of $Pr(f_\theta(X_U, X_R = x_R) = 1)$, then the probability of $Pr(f_\theta(X_U, X_R = x_R) = 0) = 1 - Z$. Additionally, we have $Z \geq 1 - \delta > 0.5$ due to the assumption that $\delta < 0.5$. The entropy of the model's prediction can be represented as: $H\big[f_\theta(X_U, X_R = x_R)\big] = -Z \log Z - (1 - Z) \log(1 - Z)$.

Let $\epsilon(Z) = -Z \log Z - (1 - Z) \log(1 - Z)$. The derivative of $\epsilon(Z)$ is given by $\frac{d\epsilon(Z)}{dZ} = \log \frac{1-Z}{Z} < 0$, as $Z > 0.5$. As a result, $\epsilon(Z)$ is a monotonically decreasing function.

When $\delta = 0$, we have $Z = 1$, and by the property of the entropy $H\big[f_\theta(X_U, X_R = x_R)\big] = 0$. □

Proposition 7.8. *Given two subsets R and R' of sensitive features S, with $R \subseteq R'$,*

$$H\big(f_\theta(X_U, X_R = x_R)\big) \geq H\big(f_\theta(X_{U'}, X_{R'} = x_{R'})\big),$$

where $U = S \setminus R$ and $U' = S \setminus R'$.

Proof. This is due to the property that conditioning reduces the uncertainty, or the well-known *information never hurts* theorem in information theory [68]. □

Proposition 7.9. *The conditional distribution of any subset of unrevealed features $U' \in U$, given the the values of released features $X_R = x_R$ is given by:*

$$\Pr(X_{U'}|X_R = x_R) = \mathcal{N}\bigg(\mu_{U'} + \Sigma_{U',R}\Sigma_{R,R}^{-1}(x_R - \mu_R),\ \Sigma_{U',U'} - \Sigma_{U',R}\Sigma_{R,R}^{-1}\Sigma_{R,U'}\bigg),$$

where Σ is the covariance matrix

Proof. This is a well-known property of the Gaussian distribution and we refer the reader to Chapter 2.3.2 of the textbook [17] for further details. □

Proposition 7.10. *The model predictions before thresholding, $\tilde{f}_\theta(X_U, X_R = x_R) = \theta_U X_U + \theta_R x_R$ is a random variable with a Gaussian distribution $\mathcal{N}\big(m_f, \sigma_f\big)$, where*

$$m_f = \theta_R x_R + \theta_U^\top\big(\mu_U + \Sigma_{U,R}\Sigma_{R,R}^{-1}(x_R - \mu_R)\big) \tag{7.9}$$

$$\sigma_f^2 = \theta_U^\top\big(\Sigma_{U,U} - \Sigma_{U,R}\Sigma_{R,R}^{-1}\Sigma_{R,U}\big)\theta_U, \tag{7.10}$$

where θ_U is the sub-vector of parameters θ corresponding to the unrevealed features U.

Proof. The proof of this statement is straightforward due to the property that a linear combination of Gaussian variables X_U is also Gaussian. Additionally, the posterior distribution of X_U is already provided in Proposition 7.3. □

Proposition 7.11. *Let the model predictions prior thresholding $\tilde{f}_\theta(X_U, X_R = x_R)$, be a random variable following a Gaussian distribution $\mathcal{N}(m_f, \sigma_f^2)$. Then, the model prediction following*

thresholding $f_\theta(X_U, X_R = x_R)$ is a random variable following a Bernoulli distribution $Bern(p)$ with $p = \Phi(\frac{m_f}{\sigma_f})$, where $\Phi(\cdot)$ refers to the CDF of the standard normal distribution, and m_f and σ_f, are given in Equations (7.5) *and* (7.6)*, respectively.*

Proof. In the case of a binary classifier, we have $f_\theta(x) = \mathbf{1}\{\tilde{f}_\theta(x) \geq 0\}$. If $\tilde{f}$ follows a normal distribution, denoted as $\tilde{f} \sim \mathcal{N}(m_f, \sigma_f^2)$, then by the properties of the normal distribution, $f\theta$ follows a Bernoulli distribution, denoted as $f_\theta \sim Bern(p)$, with parameter $p = \Phi(\frac{m_f}{\sigma_f})$, where $\Phi(\cdot)$ is the cumulative density function of the standard normal distribution. □

Proposition 7.12. *Assume f_θ is a linear classifier. Then, determining if a subset U of sensitive features S is a* pure *core feature set can be performed in $O(|P| + |S|)$ time.*

Proof. As discussed in the main text, to test if a subset U is a core feature set or not, we need to check if the following two terms have the same sign (either negative or non-negative):

$$\begin{aligned} \max_{X_U} \theta_U^\top X_U + \theta_R^\top x_R &= \|\theta\|_1 + \theta_R^\top x_R \\ \min_{X_U} \theta_U^\top X_U + \theta_R^\top x_R &= -\|\theta\|_1 + \theta_R^\top x_R. \end{aligned} \tag{7.11}$$

These can be solved in time $O(|P| + |S|)$ due to the property of the linear equality above. □

Theorem 7.2. *The distribution of the random variable $\tilde{f}_\theta = \tilde{f}_\theta(X_U, X_R = x_R)$ where $X_U \sim \mathcal{N}\big(\mu_U^{pos}, \Sigma_U^{pos}\big)$ can be approximated by a Normal distribution as*

$$\tilde{f}_\theta \sim \mathcal{N}\big(\tilde{f}_\theta(X_U = \mu_U^{pos}, X_R = x_R), g_U^\top \Sigma_U^{pos} g_U\big) \tag{7.12}$$

where $g_U = \nabla_{X_U} \tilde{f}_\theta(X_U = \mu_U^{pos}, X_R = x_R)$ is the gradient of model prediction at $X_U = \mu_U^{pos}$.

Proof. The proof relies on the first Taylor approximation of classifier $\tilde{f}$ around its mean:

$$\tilde{f}_\theta(X_U, X_R = x_R,) \approx \tilde{f}_\theta(X_U = \mu_U^{pos}, X_R = x_R) + (X_U - \mu_U^{pos})^T \nabla_{X_U} \tilde{f}_\theta(X_U = \mu_U^{pos}, X_R = x_R) \tag{7.13}$$

Algorithm 3: PFR for non-linear classifiers

input : A test sample x; Training data D
output: A core feature set R and its representative label $\tilde{y}$

1 $\mu \leftarrow \frac{1}{|D|} \sum_{(x,y)\in D} x$
2 $\Sigma \leftarrow \frac{1}{|D|} \sum_{(x,y)\in D} (x-\mu)(x-\mu)^\top$
3 Initialize $R = \emptyset$
4 **while** *True* **do**
5 **if** R *is a core feature set with repr. label* $\tilde{y}$ **then**
6 **return** $(R, \tilde{y})$
7 **else**
8 **foreach** $j \in U$ **do**
9 Compute $\Pr(X_j | X_R = x_R)$ (using Prop. 7.3)
10 $\boldsymbol{Z} \leftarrow$ sample$(\Pr(X_j | X_R = x_R))$ T times
11 Compute $\Pr\left(f_\theta(X_j = z, X_{U\setminus\{j\}} X_R = x_R)\right)$ using Theorem 7.2)
12 Compute $F(X_j)$ (using Eq. (7.4))
13 $j^* \leftarrow \operatorname{argmax}_j F(X_j)$
14 $R \leftarrow R \cup \{j^*\}$
15 $U \leftarrow U \setminus \{j^*\}$

Since $X_U \sim \mathcal{N}\big(\mu_U^{\text{pos}}, \Sigma_U^{\text{pos}}\big)$ hence $X_U - \mu_U^{\text{pos}} \sim \mathcal{N}\big(\mathbf{0}, \Sigma_U^{\text{pos}}\big)$. By the properties of normal distribution, the right-hand side of Equation (7.13) is a linear combination of Gaussian variables, and it is also Gaussian. □

7.9.2 Algorithms pseudocode

The pseudocode for PFR for non-linear classifiers is presented in Algorithm 3. There are two main differences between this algorithm and the case of linear classifiers. Firstly, the procedure of pure core feature testing on line 5 takes exponential time with respect to $|U|$ instead of linear time as in the case of linear classifiers. Additionally, we use Theorem 7.2 to estimate the distribution of the soft prediction as seen on line 11, as the exact distribution cannot be computed analytically as in the case of linear classifiers.

7.9.3 Extension from binary to multiclass classification

In the main text, we provide the implementation of PFR for binary classification problem. In this section, we extend the method to the multiclass classification problem.

7.9.4 Estimating $P(f_\theta(X_U, X_R = x_R))$

In order to achieve our goals of determining if a subset is a core feature set for a given $\delta > 0$, and computing the entropy in the scoring function, we need to estimate the distribution of $f_\theta(X_U, X_R = x_R)$. In this section, we first discuss the method of computing the distribution of $\tilde{f}\theta(X_U, X_R = x_R)$ for both linear and non-linear models. Once this is done, we then address the challenge of estimating the hard label distribution $P(f\theta(X_U, X_R = x_R))$.

It is important to note that, under the assumption that the input features X are normally distributed with mean μ and covariance matrix Σ, the linear classifier $\tilde{f}_\theta = \theta^\top x$ will also have a multivariate normal distribution. Specifically, if $X_U \sim \mathcal{N}(\mu_U^{pos}, \Sigma_U^{pos})$, then $\tilde{f}\theta(X_U, X_R = x_R) \sim \mathcal{N}(\theta_R^\top x_R + \theta_U^T \mu_U^{pos}, \theta_U^\top \Sigma \theta_U)$.

For non-linear classifiers, the output $f_\theta(X_U, X_R = x_R)$ is not a Gaussian distribution due to the non-linear transformation. To approximate it, we use Theorem 7.2 which states that the non-linear function $\tilde{f}_\theta(X_U, X_R = x_R)$ can be approximated as a multivariate Gaussian distribution.

Challenges when estimating $P(f_\theta(X_U, X_R = x_R))$ For multi-class classification problems, the hard label $f_\theta(X_U, X_R = x_R)$ is obtained by selecting the class with the highest score, which is given by $\mathrm{argmax}_{i \in [L]} \tilde{f}_\theta^i(X_U, X_R = x_R)$. However, due to the non-analytical nature of the argmax function, even when $\tilde{f}\theta(X_U, X_R = x_R)$ follows a Gaussian distribution, the distribution of $f\theta(X_U, X_R = x_R)$ cannot be computed analytically. To estimate this distribution, we resort to Monte Carlo sampling. Specifically, we draw a number of samples from $P(f_\theta(X_U, X_R = x_R))$, and approximate the probability of each class as the proportion of samples that fall in that class.

We provide experiments of PFR for multi-class classification cases in Section 7.9.8.

7.9.5 Experiments details

Datasets information To show the advantages of the suggested PFR technique for safeguarding feature-level privacy, we employ benchmark datasets in our experiments. These datasets include both binary and multi-class classification datasets. The following are examples of binary datasets that we use to evaluate the method:

1. Bank dataset [18]. The objective of this task is to predict whether a customer will subscribe to a term deposit using data from various features, including but not limited to call duration and age. There are a total of 16 features available for this analysis.

2. Adult income dataset [18]. The goal of this task is to predict whether an individual earns more than $50,000 annually. After preprocessing the data, there are a total of 40 features available for analysis, including but not limited to occupation, gender, race, and age.

3. Credit card default dataset [18]. The objective of this task is to predict whether a customer will default on a loan. The data used for this analysis includes 22 different features, such as the customer's age, marital status, and payment history.

4. Car insurance dataset [106]. The task at hand is to predict whether a customer has filed a claim with their car insurance company. The dataset for this analysis is provided by the insurance company and includes 16 features related to the customer, such as their gender, driving experience, age, and credit score.

Furthermore, we also evaluate our method on two additional multi-class classification datasets:

1. Customer segmentation dataset [119]. The task at hand is to classify a customer into one of four distinct categories: A, B, C, and D. The dataset used for this task contains 9 different features, including profession, gender, and working experience, among others.

2. Children fetal health dataset [73]. The task at hand is to classify the health of a fetus into one of three categories: normal, suspect, or pathological, using data from CTG (cardiotocogra-

phy) recordings. The data includes approximately 21 different features, such as heart rate and the number of uterine contractions.

Settings: For each dataset, 70% of the data will be used for training the classifiers, while the remaining 30% will be used for testing. The number of sensitive features, denoted as $|S|$, will be chosen randomly from the set of all features, with $|S|$ ranging from 2 to 7. The remaining features will be considered as public. 100 repetition experiments will be performed for each choice of $|S|$, under different random seeds, and the results will be averaged. All methods that require Monte Carlo sampling will use 1000 random samples. The performance of different methods will be evaluated based on accuracy and information leakage. Two different classifiers will be considered.

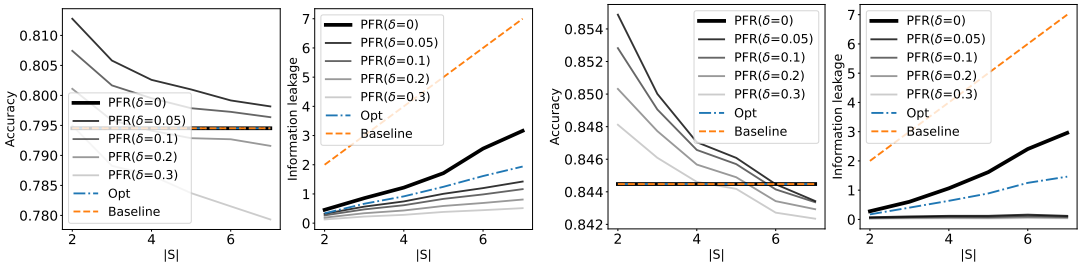
1. Linear classifiers: We use Logistic Regression as the base classifier.

2. Nonlinear classifiers: The nonlinear classifiers used in this study consist of a neural network with two hidden layers, using the ReLU activation function. The number of nodes in each hidden layer is set to 10. The network is trained using stochastic gradient descent (SGD) with a batch size of 32 and a learning rate of 0.001 for 300 epochs. A value of Δ = 0.2 is used when testing the pure core feature set for nonlinear classifiers.

Baseline models. We compare our proposed algorithms with the following baseline models:

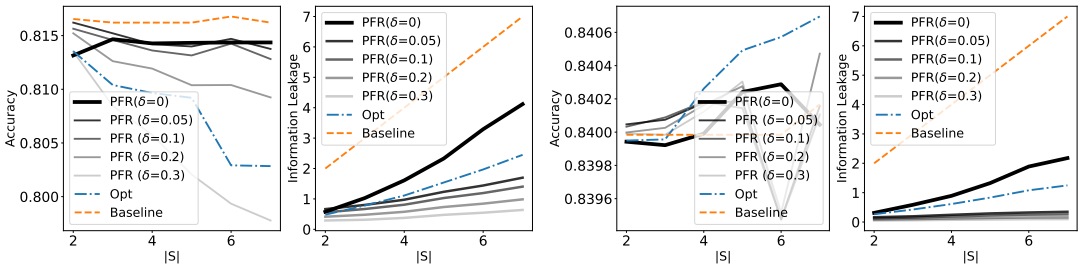
1. **Baseline**: This refers to the usage of original classifier which asks users to reveal **all** sensitive features.

2. **Opt**: This method involves evaluating all possible subsets of sensitive features in order to identify the minimum *pure* core feature set. For each subset, the verification algorithm is used to determine whether it is a pure core feature set. The minimum pure core feature set that is found is then selected. It should be noted that as all possible subsets are evaluated, all sensitive feature values must be revealed. Therefore, this approach is not practical in real-world scenarios. However, it does provide a lower bound on information leakage for PFR (when $\delta = 0$).



(a) Bank dataset (b) Adult income dataset

Fig. 7.5: Accuracy and information leakage for different choices of number of private features m under Logistic Regression classifiers



(a) Bank dataset (b) Adult income dataset

Fig. 7.6: Accuracy and information leakage for different choices of number of sensitive features $|S|$ under non-linear classifiers

Metrics. We compare all different algorithms in terms of accuracy and information leakage:

1. Accuracy. For algorithms that are based on the core feature set, such as our PFR and "Opt," the representative label is used as the model's prediction. The accuracy is then determined by comparing this label to the ground truth.

2. Information leakage. We compute the average number of sensitive features that need to be revealed over the test set. A smaller number is considered better.

7.9.6 Additional experiments on linear binary classifiers

Additional experiments were conducted to compare the performance of PFR to that of the baseline methods using linear classifiers on the Bank and Adult income datasets, as shown in Figure 7.5. As in the main text, a consistent trend in terms of performance is observed. As the number of sensitive

attributes, $|S|$, increases, the information leakage introduced by PFR with various values of δ increases at a slower rate. With different choices of $|S|$, PFR (with $\delta = 0$) requires the revelation of at most 50% of sensitive information. To significantly reduce the information leakage of PFR, the value of δ can be relaxed. By choosing an appropriate value for the failure probability, such as $\delta = 0.1$, only minimal accuracy is sacrificed (at most 0.002%), while the information leakage can be reduced to as low as 5% of the total number of sensitive attributes.

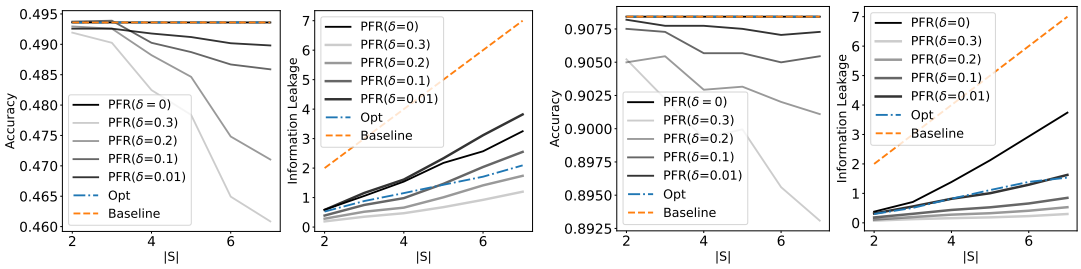
7.9.7 Additional experiments on non-linear binary classifiers

Additional experiments were conducted to compare the performance of PFR to that of the baseline methods using non-linear classifiers on the Bank and Adult income datasets, as shown in Figure 7.6. As seen, while the Baseline method requires the revelation of all sensitive attributes, PFR with different values of δ only requires the revelation of a much smaller number of sensitive attributes. The accuracy difference between the Baseline method and PFR is also minimal (at most 2%). These results demonstrate the effectiveness of PFR in protecting privacy while maintaining a good prediction performance for test data.

7.9.8 Evaluation of PFR on multi-class classifiers

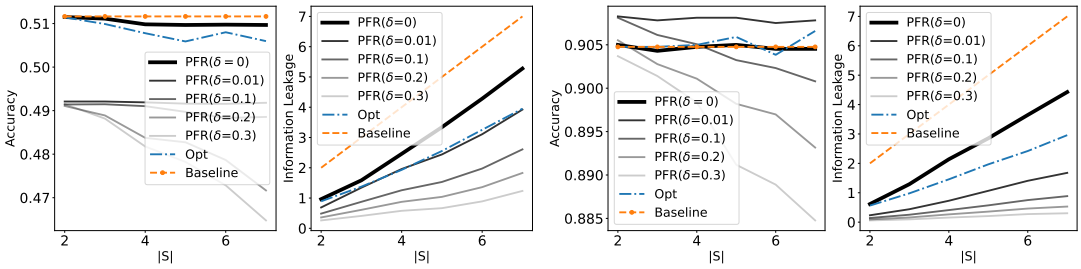
Linear classifiers We also provide a comparison of accuracy and information leakage between our proposed FPR and the baseline models for linear classifiers. These metrics are reported for the Customer and Children Fetal Health datasets in Figure 7.7. The figure clearly shows the benefits of FPR in reducing information leakage while maintaining a comparable accuracy to the baseline models.

Nonlinear classifiers Similarly, we present a comparison of our proposed algorithms with the baseline methods when using non-linear classifiers. These metrics are reported for the Customer and Children Fetal Health datasets in Figure 7.8. The results show that using PFR with a value of $\delta = 0$ results in a minimal decrease in accuracy, but significantly reduces the amount of information



(a) Customer dataset (b) Children fetal health dataset

Fig. 7.7: Accuracy and information leakage for different choices of number of sensitive features $|S|$ under multinomial Logistic Regression



(a) Customer segmentation dataset (b) Children fetal health dataset

Fig. 7.8: Accuracy and information leakage for different choices of number of sensitive features $|S|$ under non-linear classifiers

leakage compared to the Baseline method.

CHAPTER 8
CONCLUSION

The current research at the intersection of differential privacy and fairness has shown promise in building solutions to realize more trustworthy systems. Furthermore, the analysis of the disparate impacts arising in several learning and decision tasks has paved the way to develop promising mitigating strategies. Despite these encouraging results, a number of challenges must be addressed to have a full understanding of the trade-offs between privacy, fairness, and accuracy.

1. The development of a unified theoretical framework to characterize and reason about fairness issues arising in general decision tasks is still missing. Of particular importance would be to capture the relation between the privacy loss values and the fairness violations resulting in both decision-making and learning settings.

2. While the current focus in the analysis of fairness in private ML tasks has focused on data and algorithmic properties, it has also been observed that batch-size and learning rate may affect the Hessian spectrum of a network classifier [Yao et al., 2018]. These observations may suggest that fairness in private ML tasks may be impacted by key hyper-parameters, including batch size, learning rates, and the depth of neural networks.

3. Another aspect that has been observed repeatedly when connecting privacy and fairness is their link with model robustness. While this observation arises both in decision and in learn-

ing tasks, an understanding of this link is currently missing.

4. A further important direction is the study of the disparate impacts that may arise in algorithms and generative models producing private synthetic datasets as well the development of mitigation measures.

5. Finally, the development of software library to facilitate auditing fairness and bias issues in a private decision or learning task would be crucial to broaden the knowledge and adoption of these important issues.

REFERENCES

[1] Healthcare dataset stroke data. 79

[2] Title 13. Title 13, u.s. code. `www.census.gov/history/www/reference/privacy_confidentiality/title_13_us_code.html`, 2006. Accessed: 2021-01-15. 7

[3] Abadi and et al. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016. 93, 203

[4] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016. 47, 48, 55

[5] John M Abowd. The us census bureau adopts differential privacy. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2867–2867, 2018. 1, 7, 202

[6] John M Abowd and Ian M Schmutte. An economic analysis of privacy protection and statistical accuracy as social choices. *American Economic Review*, 2019. 43

[7] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69. PMLR, 2018. 160, 172

[8] Nima Aghli and Eraldo Ribeiro. Combining weight pruning and knowledge distillation for cnn compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3191–3198, 2021. 129, 130

[9] Orevaoghene Ahia, Julia Kreutzer, and Sara Hooker. The low-resource double bind: An empirical study of pruning for low-resource machine translation. In *EMNLP*, 2021. 131

[10] Ahmet Aktay, Shailesh Bavadekar, Gwen Cossoul, John Davis, Damien Desfontaines, Alex Fabrikant, Evgeniy Gabrilovich, Krishna Gadepalli, Bryant Gipson, Miguel Guevara, Chaitanya Kamath, Mansi Kansal, Ali Lange, Chinmoy Mandayam, Andrew Oplinger, Christopher Pluntke, Thomas Roessler, Arran Schlosberg, Tomer Shekel, Swapnil Vispute, Mia Vu, Gregory Wellenius, Brian Williams, and Royce J Wilson. Google covid-19 community mobility reports: Anonymization process description (version 1.1). *arXiv*, 2004.04145, 2020. 1

[11] Kareem Amin, Travis Dick, Alex Kulesza, Andres Munoz, and Sergei Vassilvitskii. Differentially private covariance estimation. *Advances in Neural Information Processing Systems*, 32, 2019. 215

[12] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. Differential privacy has disparate impact on model accuracy. In *NeurIPS*, volume 32, 2019. xii, 2, 45, 47, 48, 49, 50, 60, 92, 94, 97, 109, 130

[13] Borja Balle and Yu-Xiang Wang. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International Conference on Machine Learning*, pages 394–403. PMLR, 2018. 79

[14] Solon Barocas, Moritz Hardt, and Arvind Narayanan. Fairness in machine learning. *Nips tutorial*, 1:2, 2017. 130

[15] Cenk Baykal, Lucas Liebenwein, Igor Gilitschenski, Dan Feldman, and Daniela Rus. Sipping neural networks: Sensitivity-informed provable pruning of neural networks. *arXiv preprint arXiv:1910.05422*, 2019. 129, 130

[16] Chris Bishop. Exact calculation of the hessian matrix for the multilayer perceptron, 1992. 89, 90

[17] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006. 221

[18] C.L. Blake and C.J. Merz. Uci repository of machine learning databases, 1988. 78, 117, 208, 215, 225

[19] Cody Blakeney, Nathaniel Huish, Yan Yan, and Ziliang Zong. Simon says: Evaluating and mitigating bias in pruned neural networks with knowledge distillation. *ArXiv*, abs/2106.07849, 2021. 130

[20] Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. What is the state of neural network pruning? *arXiv preprint arXiv:2003.03033*, 2020. 129, 130

[21] Dankmar Böhning. Multinomial logistic regression algorithm. *Annals of the institute of Statistical Mathematics*, 44(1):197–200, 1992. 60, 62

[22] Robin Burke. Hybrid systems for personalized recommendations. In *IJCAI Workshop on Intelligent Techniques for Web Personalization*, pages 133–152. Springer, 2003. 157

[23] Fabrizio Carcillo, Yann-Aël Le Borgne, Olivier Caelen, Yacine Kessaci, Frédéric Oblé, and Gianluca Bontempi. Combining unsupervised and supervised learning in credit card fraud detection, 05 2019. 79, 118

[24] Simon Caton and Christian Haas. Fairness in machine learning: A survey. *arXiv preprint arXiv:2010.04053*, 2020. 130

[25] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014. 204

[26] Hongyan Chang and Reza Shokri. On the privacy risks of algorithmic fairness. *arXiv preprint arXiv:2011.03731*, 2020. 45, 49

[27] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(3), 2011. 47, 48, 52

[28] Kamalika Chaudhuri, Claire Monteleoni, and Anand D. Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 2011. 65, 203

[29] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR, 2019. 139

[30] Ronan Collobert, Fabian Sinz, Jason Weston, and Léon Bottou. Trading convexity for scalability. In *Proceedings of the 23rd international conference on Machine learning*, pages 201–208, 2006. 174, 192

[31] IBM Analytics Communities. Telco customer churn dataset, 2015. 78

[32] Graham Cormode, Somesh Jha, Tejas Kulkarni, Ninghui Li, Divesh Srivastava, and Tianhao Wang. Privacy at scale: Local differential privacy in practice. In *Proceedings of the 2018 International Conference on Management of Data*, pages 1655–1658, 2018. 202

[33] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. On the compatibility of privacy and fairness. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, pages 309–315, 2019. 45, 49

[34] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. On the compatibility of privacy and fairness. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, pages 309–315, 2019. 93, 109

[35] Mengnan Du, Subhabrata Mukherjee, Yu Cheng, Milad Shokouhi, Xia Hu, and Ahmed Hassan Awadallah. What do compressed large language models forget? robustness challenges in model compression. *ArXiv*, abs/2110.08419, 2021. 131

[36] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012. 45, 49, 130

[37] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006. 1, 8, 9, 47, 48, 50, 92, 94, 202, 203

[38] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Theoretical Computer Science*, 9(3-4):211–407, 2013. 9

[39] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014. 51, 94

[40] Michael D Ekstrand, Rezvan Joshaghani, and Hoda Mehrpouyan. Privacy for all: Ensuring fair and equitable privacy protections. In *Conference on Fairness, Accountability and Transparency*, pages 35–47, 2018. 45

[41] Michael D Ekstrand, Rezvan Joshaghani, and Hoda Mehrpouyan. Privacy for all: Ensuring fair and equitable privacy protections. In *Conference on Fairness, Accountability and Transparency*, pages 35–47, 2018. 93, 109

[42] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067. ACM, 2014. 202

[43] Tom Farrand, Fatemehsadat Mireshghallah, Sahib Singh, and Andrew Trask. Neither private nor fair: Impact of data imbalance on utility and fairness in differential privacy. In *Proceedings of the 2020 Workshop on Privacy-Preserving Machine Learning in Practice*, pages 15–19, 2020. 49, 50, 83, 94, 109

[44] Ferdinando Fioretto, Pascal Van Hentenryck, Terrence WK Mak, Cuong Tran, Federico Baldo, and Michele Lombardi. Lagrangian duality for constrained deep learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 118–135. Springer, 2020. 141, 152

[45] Ferdinando Fioretto, Cuong Tran, and Pascal Van Hentenryck. Decision making with differential privacy under a fairness lens, 2021. 2, 92

[46] Ferdinando Fioretto, Cuong Tran, Pascal Van Hentenryck, and Keyu Zhu. Differential privacy and fairness in decisions and learning tasks: A survey. *CoRR*, abs/2202.08187, 2022. 93, 130

[47] Yifan Fu, Xingquan Zhu, and Bin Li. A survey on instance selection for active learning. *Knowledge and information systems*, 35(2):249–283, 2013. 204

[48] GDPR. What is gdpr, the eu's new data protection law? `https://gdpr.eu/what-is-gdpr`, 2020. Accessed: 2021-01-15. 7

[49] Badih Ghazi, Noah Golowich, Ravi Kumar, Pasin Manurangsi, and Chiyuan Zhang. Deep learning with label differential privacy. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34*, 2021. 93

[50] Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P Friedlander. Satisfying real-world goals with dataset constraints. *Advances in Neural Information Processing Systems*, 29, 2016. 174, 192

[51] Aidan Good, Jiaqi Lin, Xin Yu, Hannah Sieg, Mikey Ferguson, Shandian Zhe, Jerzy Wieczorek, and Thiago Serra. Recall distortion in neural network pruning and the undecayed pruning algorithm. In *Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 131

[52] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2014. 146

[53] Yiwen Guo, Chao Zhang, Changshui Zhang, and Yurong Chen. Sparse dnns with improved adversarial robustness. In *Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS)*, page 240–249, 2018. 146

[54] Moritz Hardt, Eric Price, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 130

[55] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. The elements of statistical learning: data mining, inference, and prediction. 2, 2009. 192

[56] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. xv, 129, 168, 188

[57] Amaç Herdagdelen, Alex Dow, Bogdan State, Payman Mohassel, and Alex Pompe. Protecting privacy in facebook mobility data during the covid-19 response. online, 2020. 1

[58] Sara Hooker, Aaron C. Courville, Gregory Clark, Yann Dauphin, and Andrea Frome. What do compressed deep neural networks forget. *arXiv: Learning*, 2020. 129, 130

[59] Sara Hooker, Nyalleng Moorosi, Gregory Clark, Samy Bengio, and Emily L. Denton. Characterising bias in compressed models. *ArXiv*, abs/2010.03058, 2020. 129, 130, 135

[60] Sepidehsadat Hosseini, Mohammad Amin Shabani, Mohammad Mahdi Jahanara, and Bahar Salamatian. Learning fair from unfair teachers. 131

[61] Sunhee Hwang, Sungho Park, Dohyung Kim, Mirae Do, and Hyeran Byun. Fairfacegan: Fairness-aware facial image-to-image translation, 2020. 118

[62] Matthew Jagielski, Michael Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. Differentially private fair learning. *arXiv preprint arXiv:1812.02696*, 2018. 109

[63] Matthew Jagielski, Michael Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. Differentially private fair learning. In *International Conference on Machine Learning*, pages 3000–3008. PMLR, 2019. 45, 49

[64] Senerath Mudalige Don Alexis Chinthaka Jayatilake and Gamage Upeksha Ganegoda. Involvement of machine learning tools in healthcare decision making. *Journal of Healthcare Engineering*, 2021, 2021. 157

[65] Vinu Joseph, Shoaib Ahmed Siddiqui, Aditya Bhaskara, Ganesh Gopalakrishnan, Saurav Muralidharan, Michael Garland, Sheraz Ahmed, and Andreas R. Dengel. Going beyond classification accuracy metrics in model compression. 2020. 130

[66] Fereshte Khani and Percy Liang. Feature noise induces loss discrepancy across groups. In *International Conference on Machine Learning*, pages 5209–5219. PMLR, 2020. 158

[67] Hoki Kim. Torchattacks: A pytorch repository for adversarial attacks. *arXiv preprint arXiv:2010.01950*, 2020. 189

[68] Andreas Krause and Carlos Guestrin. *A note on the budgeted maximization of submodular functions*. Citeseer, 2005. 221

[69] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 168, 187

[70] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). 143, 151

[71] Satya Kuppam, Ryan Mckenna, David Pujol, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. Fair decision making using privacy-protected data, 2020. 7, 12, 45, 47, 48, 49, 50, 93, 109, 130

[72] 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, 2020. 108

[73] Larxel. Children fetal health dataset. `https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification`, 2021. 225

[74] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 656–672. IEEE, 2019. 168

[75] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu. Feature selection: A data perspective. *ACM computing surveys (CSUR)*, 50(6):1–45, 2017. 204

[76] Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. Fair resource allocation in federated learning. *arXiv preprint arXiv:1905.10497*, 2019. 186, 187

[77] Max Little, Patrick Mcsharry, Stephen Roberts, Declan Costello, and Irene Moroz. Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *Biomedical engineering online*, 6:23, 02 2007. 78, 118

[78] Xiyang Liu, Weihao Kong, Sham Kakade, and Sewoong Oh. Robust and differentially private mean estimation. *Advances in neural information processing systems*, 34:3887–3901, 2021. 215

[79] Xinsong Ma, Zekai Wang, and Weiwei Liu. On the tradeoff between robustness and fairness. *Advances in Neural Information Processing Systems*, 35, 2022. 168

[80] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 172, 189

[81] Mohammad Mahdi Khalili, Xueru Zhang, Mahed Abroshan, and Somayeh Sojoudi. Improving fairness and privacy in selection problems. *arXiv e-prints*, pages arXiv–2012, 2020. 45, 49

[82] Ata Mahjoubfar, Claire Lifan Chen, and Bahram Jalali. Deep learning and classification. In *Artificial Intelligence in Label-free Microscopy*, pages 73–85. Springer, 2017. 101

[83] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 94–103. IEEE, 2007. 46

[84] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021. 130, 157

[85] Ilya Mironov. Rényi differential privacy. *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, Aug 2017. 116

[86] S'ergio Moro, P. Cortez, and P. Rita. A data-driven approach to predict the success of bank telemarketing. *Decis. Support Syst.*, 62:22–31, 2014. 78

[87] Hussein Mozannar, Mesrob I. Ohannessian, and Nathan Srebro. Fair learning with private demographic data. In *Proceedings of the 37th International Conference on Machine Learning*, 2020. 49, 93, 109

[88] Michael C Mozer and Paul Smolensky. Skeletonization: A technique for trimming the fat from a network via relevance assessment. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 1. Morgan-Kaufmann, 1988. 132

[89] Vedant Nanda, Samuel Dooley, Sahil Singla, Soheil Feizi, and John P Dickerson. Fairness through robustness: Investigating robustness disparity in deep learning. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 466–477, 2021. 158

[90] Yuval Netzer, Tao Wang, Adam Coates, A. Bissacco, Bo Wu, and A. Ng. Reading digits in natural images with unsupervised feature learning. 2011. 143, 151

[91] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015. 168

[92] Michela Paganini. Prune responsibly. *arXiv preprint arXiv:2009.09936*, 2020. 130

[93] Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. 10 2016. 92

[94] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv: Arxiv-1605.07277*, 2016. 146

[95] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman. Towards the science of security and privacy in machine learning. *arXiv preprint arXiv:1611.03814*, 2016. 202

[96] Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE symposium on security and privacy (SP)*, pages 582–597. IEEE, 2016. 108

[97] Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Úlfar Erlingsson. Scalable private learning with pate. 02 2018. 96, 108, 116

[98] Nicolas Papernot and Thomas Steinke. Hyperparameter tuning with renyi differential privacy. *arXiv preprint arXiv:2110.03620*, 2021. 108

[99] Giorgio Patrini, Richard Nock, Paul Rivera, and Tiberio Caetano. (almost) no label no cry. *Advances in Neural Information Processing Systems*, 27:190–198, 2014. 99

[100] Inioluwa Deborah Raji, Timnit Gebru, Margaret Mitchell, Joy Buolamwini, Joonseok Lee, and Emily Denton. Saving face: Investigating the ethical concerns of facial recognition auditing. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 145–151, 2020. 162

[101] Bashir Rastegarpanah, Krishna Gummadi, and Mark Crovella. Auditing black-box prediction models for data minimization compliance. *Advances in Neural Information Processing Systems*, 34:20621–20632, 2021. 202

[102] Steffen Rebennack and Vitaliy Krasko. Piecewise linear function fitting via mixed-integer linear programming. *INFORMS Journal on Computing*, 32(2):507–530, 2020. 20

[103] Protection Regulation. Regulation (eu) 2016/679 of the european parliament and of the council. *Regulation (eu)*, 679:2016, 2016. 202

[104] Alex Renda, Jonathan Frankle, and Michael Carbin. Comparing rewinding and fine-tuning in neural network pruning. In *International Conference on Learning Representations*, 2020. 129, 130

[105] Kit T Rodolfa, Hemank Lamba, and Rayid Ghani. Empirical observation of negligible fairness–accuracy trade-offs in machine learning for public policy. *Nature Machine Intelligence*, 3(10):896–904, 2021. 157

[106] Sagnik Roy. Car insurance data. `https://www.kaggle.com/datasets/sagnik1511/car-insurance-data`, 2021. 225

[107] J. Tran S. Han, J. Pool and W. J. Dally. Learning both weights and connections for efficient neural networks. In *NIPS*, 2015. 129, 130, 132

[108] Peter Sadowski. Lecture Notes: Notes on Backpropagation, 2021. URL: `https://www.ics.uci.edu/~pjsadows/notes.pdf`. Last visited on 2021/05/01. 89, 124

[109] Amartya Sanyal, Yaxi Hu, and Fanny Yang. How unfair is private learning? In *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, Proceedings of Machine Learning Research. PMLR, 2022. 93

[110] Candice Schumann, Jeffrey Foster, Nicholas Mattei, and John Dickerson. We need fairness and explainability in algorithmic hiring. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2020. 157

[111] Vikash Sehwag, Shiqi Wang, Prateek Mittal, and Suman Jana. Towards compact and robust deep neural networks. *preprint arXiv:1906.06110*, 2019. 129, 130

[112] Burr Settles. Active learning literature survey. 2009. 204

[113] Shai Shalev-Shwartz. Online learning: Theory, algorithms, and applications. 08 2007. 111, 112

[114] Zheng Shi, Nicolas Loizou, Peter Richtárik, and Martin Takáč. Ai-sarah: Adaptive and implicit stochastic recursive gradient methods, 2021. 104, 122

[115] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 168, 188

[116] Lisa Simunaci. Pro rata vaccine distribution is fair, equitable. t.ly/sDa9, 2021. 7

[117] W. Sonnenberg. Allocating grants for title i. *U.S. Department of Education, Institute for Education Science*, 2016. 10

[118] Alexander Stevens, Peter Deruyck, Ziboud Van Veldhoven, and Jan Vanthienen. Explainability and fairness in machine learning: Improve fair end-to-end lending for kiva. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1241–1248. IEEE, 2020. 157

[119] Abishek Sudarshan. Customer segmentation data. https://www.kaggle.com/datasets/abisheksudarshan/customer-segmentation, 2021. 225

[120] Apple Differential Privacy Team. Learning with privacy at scale. *Apple Machine Learning Journal*, 1(8), 2017. 1

[121] Wiebke Toussaint, Akhil Mathur, Fahim Kawsar, and Aaron Yi Ding. Tiny, always-on and fragile: Bias propagation through design choices in on-device machine learning workflows, 2022. 130

[122] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017. xvi, 159, 169, 189

[123] Cuong Tran, My Dinh, and Ferdinando Fioretto. Differentially private empirical risk minimization under the fairness lens. In *Advances in Neural Information Processing Systems*, 2021. 130

[124] Cuong Tran, My H. Dinh, and Ferdinando Fioretto. Differentially private deep learning under the fairness lens, 2021. 105

[125] Cuong Tran, Ferdinando Fioretto, Pascal Van Hentenryck, and Zhiyan Yao. Decision making with differential privacy under a fairness lens. In Zhi-Hua Zhou, editor, *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 560–566, 2021. xii, 2, 130

[126] Cuong Tran, Ferdinando Fioretto, Pascal Van Hentenryck, and Zhiyan Yao. Decision making with differential privacy under a fairness lens. In Zhi-Hua Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 560–566, 2021. 94, 109

[127] Cuong Tran, Ferdinando Fioretto, Jung-Eun Kim, and Rakshit Naidu. Pruning has a disparate impact on model accuracy. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 168

[128] Cuong Tran, Ferdinando Fioretto, and Pascal Van Hentenryck. Differentially private and fair deep learning: A lagrangian dual approach. *arXiv preprint arXiv:2009.12562*, 2020. 49, 57, 65, 93, 109

[129] Cuong Tran, Ferdinando Fioretto, and Pascal Van Hentenryck. Differentially private and fair deep learning: A lagrangian dual approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9932–9939, 2021. 160

[130] Archit Uniyal, Rakshit Naidu, Sasikanth Kotti, Sahib Singh, Patrik Kenfack, Fatemehsadat Mireshghallah, and Andrew Trask. Dp-sgd vs pate: Which has less disparate impact on model accuracy?, 06 2021. 93, 94, 109

[131] Archit Uniyal, Rakshit Naidu, Sasikanth Kotti, Sahib Singh, Patrik Joslin Kenfack, FatemehSadat Mireshghallah, and Andrew Trask. Dp-sgd vs pate: Which has less disparate impact on model accuracy? *ArXiv*, abs/2106.12576, 2021. 130

[132] Di Wang, Minwei Ye, and Jinhui Xu. Differentially private empirical risk minimization revisited: Faster and more general. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach,

R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 52

[133] Ziqi Wang and Marco Loog. Enhancing classifier conservativeness and robustness by polynomiality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13327–13336, 2022. 168

[134] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 168, 187, 188

[135] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*, 2018. 203

[136] Canwen Xu, Wangchunshu Zhou, Tao Ge, Ke Xu, Julian McAuley, and Furu Wei. Beyond preserved accuracy: Evaluating loyalty and robustness of BERT compression. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10653–10659, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. 131

[137] Depeng Xu, Wei Du, and Xintao Wu. Removing disparate impact of differentially private stochastic gradient descent on model accuracy, 2020. 47, 48, 49, 92

[138] Guangxuan Xu and Qingyuan Hu. Can model compression improve nlp fairness. *ArXiv*, abs/2201.08542, 2022. 131

[139] Han Xu, Xiaorui Liu, Yaxin Li, Anil K. Jain, and Jiliang Tang. To be robust or to be fair: Towards fairness in adversarial training, 2021. 130, 158, 164, 165, 168, 169, 172, 173

[140] Runhua Xu, Nathalie Baracaldo, Yi Zhou, Ali Anwar, and Heiko Ludwig. Hybridalpha. *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security - AISec'19*, 2019. 49

[141] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019. 161

[142] Jiaqi Zhang, Kai Zheng, Wenlong Mou, and Liwei Wang. Efficient private erm for smooth objectives. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 3922–3928, 2017. 52, 65, 96

[143] Tianyun Zhang, Shaokai Ye, Kaiqi Zhang, Jian Tang, Wujie Wen, Makan Fardad, and Yanzhi Wang. A systematic dnn weight pruning framework using alternating direction method of multipliers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 184–199, 2018. 129, 130

[144] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017. 134, 143, 151, 168, 187

[145] Han Zhao and Geoff Gordon. Inherent tradeoffs in learning fair representations. *Advances in neural information processing systems*, 32:15675–15685, 2019. 130, 134, 159

[146] Keyu Zhu, Pascal Van Hentenryck, and Ferdinando Fioretto. Bias and variance of post-processing in differential privacy, 2020. 40

[147] Keyu Zhu, Pascal Van Hentenryck, and Ferdinando Fioretto. Bias and variance of post-processing in differential privacy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11177–11184, 2021. 130

[148] Zhaowei Zhu, Tianyi Luo, and Yang Liu. The rich get richer: Disparate impact of semi-supervised learning. *arXiv preprint arXiv:2110.06282*, 2021. 130

VITA

NAME OF AUTHOR: Cuong Tran

MAJOR: Computer and Information Science and Engineering

EDUCATION:

M.S. 2023 Syracuse University, New York, USA

B.S. 20012 Hanoi University of Science and Technology, Ha Noi, Vietnam

AWARDS AND HONORS:

1. Fifth place in *Pushback to the Future: Predict Pushback Time at US Airports challenge* (hosted by NASA). Received 7500 USD cash award.
2. AAAI 2023/NeurIPS Scholar Award.
3. **Winner** of 2022 **Caspar Bowden** Award for Outstanding Research in Privacy Enhancing Technologies. Link: `https://petsymposium.org/award/winners.php`.
4. **Third Place (5000 USD award)** in *NIST Differential Privacy Temporal Map Challenge 2020*. Individually designed and implemented end to end solution.
5. Syracuse ECS Graduate assistantship, 2020
6. Rutgers TA/GA Development Fund, 2017
7. Vietnam Education Foundation fellowship, 2013.

WORKING EXPERIENCE:

1. Applied Scientist Intern at Amazon Alexa (Summer 2021 and 2022)

2. Data Scientist at Trusting Social (2018-2020).

SERVICES

PC member of AAAI, IJCAI, NeurIPS, CVPR, ICCV, AAMAS (2021 to current).

SELECTED PUBLICATION:

1. **Cuong Tran** and Ferdinando Fioretto. Personalized privacy auditing and optimization at test time. Under submission to ICML, 2023

2. **Cuong Tran**, Keyu Zhu, Ferdinando Fioretto, and Pascal Van Hentenryck. Fairness increases adversarial vulnerability. Under submission to CVPR, 2023

3. **Cuong Tran**, Ferdinando Fioretto, Jung-Eun Kim, and Rakshit Naidu. Pruning has a disparate impact on model accuracy. In Advances in Neural Information Processing Systems (NeurIPS), 2022. **Spotlight talk, acceptance rate: 3%**

4. Ferdinando Fioretto, **Cuong Tran**, Keyu Zhu, and Pascal Van Hentenryck. Differential privacy and fairness in decisions and learning tasks: A survey. In In IJCAI Survey Track, 2022

5. **Cuong Tran**, Ferdinando Fioretto, and Pascal Van Hentenryck. SF-PATE: Scalable, fair, and private aggregation of teacher ensembles. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2023

6. **Cuong Tran** and Ferdinando Fioretto. On the fairness impacts of private ensembles models. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2023

7. **Cuong Tran**, My H. Dinh, and Ferdinando Fioretto. Differentially private deep learning under the fairness lens. In Advances in Neural Information Processing Systems (NeurIPS), 2021

8. **Cuong Tran**, Ferdinando Fioretto, Pascal Van Hentenryck, and Zhiyan Yao. Decision

making with differential privacy under the fairness lens. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2021

9. **Cuong Tran**, Ferdinando Fioretto, and Pascal Van Hentenryck. Differentially private and fair deep learning: A lagrangian dual approach. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2021.