Exploiting Problem Structure in Privacy-Preserving Optimization and Machine Learning

Paul Mangold

▶ **To cite this version:**

Paul Mangold. Exploiting Problem Structure in Privacy-Preserving Optimization and Machine Learning. Machine Learning [cs.LG]. Université de Lille, 2023. English. NNT : . tel-04443333v1

HAL Id: tel-04443333

https://hal.science/tel-04443333v1

Submitted on 15 Dec 2023 (v1), last revised 7 Feb 2024 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université de Lille

Thèse de Doctorat

pour obtenir le grade de :

Docteur de l'Université de Lille

dans la spécialité « Informatique et applications »

par

Paul Mangold

Exploiting Problem Structure in Privacy-Preserving Optimization and Machine Learning

---

Exploitation de la Structure des Problèmes en Optimisation et en Apprentissage Automatique Respectueux de la Vie Privée

Thèse soutenue le 11 octobre 2023 devant le jury composé de :

Président du jury :			
M.	Jamal Atif	Professeur,	Université Paris-Dauphine
Rapporteur·es :			
Mme	Katrina Ligett	Professor,	Hebrew University
M.	Eric Moulines	Professeur,	École Polytechnique
Examinateur·ices :			
M.	Cristóbal Guzmán	Assistant Professor,	Universidad Católica de Chile
Mme	Catuscia Palamidessi	Directrice de Recherche,	Inria
M.	Peter Richtárik	Professor,	KAUST
Directeurs de thèse :			
M.	Aurélien Bellet	Directeur de Recherche,	Inria
M.	Marc Tommasi	Professeur,	Université de Lille
Invité :			
M.	Joseph Salmon	Professeur,	Université de Montpellier

Abstract

In the past decades, concerns about the societal impact of machine learning have been growing. Indeed, if machine learning has proven its usefulness in science, day-to-day applications, and many other domains, its success is principally due to the availability of large datasets. This raises two concerns, the first about the confidentiality of the training data, and the second, about possible discrimination in a model's predictions. Trustworthy machine learning aims at providing technical answers to these concerns.

Unfortunately, guaranteeing the privacy of the training data and the fairness of the predictions often decreases the utility of the learned model. This problem has drawn significant interest in the past years, but most of existing methods (usually based on stochastic gradient descent) tend to fail in some common scenarios, like training of high-dimensional models. In this thesis, we study how structural properties of machine learning problems can be exploited to improve the trade-off between privacy and utility, and how this can impact the fairness of the predictions.

The first two contributions of this thesis are two new differentially private optimization algorithms, that are both based on coordinate descent. They aim at exploiting different structural properties of the problem at hand. The first algorithm is based on stochastic coordinate descent, and can exploit imbalance in the scale of the gradient's coordinates by using large step sizes. This allows our algorithm to obtain useful models in difficult problems, where stochastic gradient descent quickly stalls. The second algorithm is based on greedy coordinate descent. Its greedy updates allow to focus on the most important coordinates of the problem, which can sometimes drastically improve utility (*e.g.,* when the solution of the problem is sparse).

The third contribution of this thesis studies the interplay of differential privacy and fairness in machine learning. These two notions have rarely been studied simultaneously, and there are growing concerns that differential privacy may exacerbate unfairness. We show that group fairness measures have interesting regularity properties, provided that the predictions of the model are Lipschitz-continuous in its parameters. This result allows to derive a bound on the difference in fairness levels between a private model and its non-private counterpart.

Résumé

Au cours des dernières décennies, les préoccupations quant à l'impact sociétal de l'apprentissage automatique se sont multipliées. En effet, si l'apprentissage automatique a prouvé son utilité dans la science, dans la vie quotidienne, ainsi que dans de nombreux autres domaines, son succès est principalement dû à la disponibilité de grands ensembles de données. Cela soulève deux préoccupations : la première concerne la confidentialité des données d'entraînement et la seconde, la possibilité de discrimination dans les prédictions d'un modèle. Le domaine de l'apprentissage automatique fiable vise à apporter des réponses techniques à ces préoccupations.

Malheureusement, garantir la confidentialité des données d'entraînement, ainsi que l'équité des prédictions, diminue souvent l'utilité du modèle appris. Ce problème a suscité un grand intérêt au cours des dernières années. Cependant, la plupart des méthodes existantes (généralement basées sur la descente de gradient stochastique) ont tendance à échouer dans des scénarios courants, tels que l'entraînement de modèles en grande dimension. Dans cette thèse, nous étudions comment les propriétés structurelles des problèmes d'apprentissage automatique peuvent être exploitées pour améliorer le compromis entre la confidentialité et l'utilité, et comment cela peut affecter l'équité des prédictions.

Les deux premières contributions de cette thèse sont deux nouveaux algorithmes d'optimisation respectant la confidentialité différentielle, tous deux basés sur la descente par coordonnées, visant à exploiter les propriétés structurelles du problème. Le premier algorithme est basé sur la descente par coordonnées stochastique et est en mesure d'exploiter le déséquilibre dans l'échelle des coordonnées du gradient en utilisant des grands pas d'apprentissage. Cela lui permet de trouver des modèles pertinents dans des scénarios difficiles, où la descente de gradient stochastique échoue. Le deuxième algorithme est basé sur la descente par coordonnées gloutonne. Les mises à jour gloutonnes permettent de se concentrer sur les coordonnées les plus importantes du problème, ce qui peut parfois améliorer considérablement l'utilité (par exemple, lorsque la solution du problème est parcimonieuse).

La troisième contribution de cette thèse étudie les interactions entre confidentialité différentielle et équité en apprentissage automatique. Ces deux notions ont rarement

été étudiées simultanément, et il existe des inquiétudes croissantes selon lesquelles la confidentialité différentielle pourrait nuire à l'équité des prédictions. Nous montrons que quand les prédictions du modèle sont lipschitziennes (par rapport à ses paramètres), les mesures d'équité de groupe présentent des propriétés de régularité intéressantes, que nous caractérisons. Ce résultat permet d'obtenir une borne sur la différence de niveaux d'équité entre un modèle privé et le modèle non-privé correspondant.

Remerciements

Avant toute chose, je tiens à remercier autant que possible l'ensemble des personnes qui m'ont permis de vivre ces trois belles année de recherche. Mes premiers remerciements vont naturellement à Marc Tommasi, Aurélien Bellet et Joseph Salmon, mes directeurs de thèse. Merci de m'avoir donné la possibilité de poursuivre aussi librement des thématiques de recherche qui me tiennent à coeur, de m'avoir envoyé à tant de conférences et impliqué dans tant de projets passionnants. Je vous remercie pour votre bienveillance, votre confiance, et tout simplement votre soutien quotidien. Je n'aurais pu rêver d'une meilleure supervision durant ma thèse, j'ai énormément appris grâce à vous et je vous en suis grandement reconnaissant.

I sincerely thank all members of the jury, for their work that inspired and will undoubtedly continue to inspire me in the years to come. I deeply thank Katrina Ligett and Eric Moulines, for accepting to review my thesis, it was a real honour for me. Thank you Jamal Atif for being president of the jury, and thank you Cristóbal Guzmán, Catuscia Palamidessi and Peter Richtárik for being part of the jury. I really appreciated discussing with you during my defense, and I hope we will have many opportunities to discuss more in the future.

Merci Éric de m'avoir donné l'opportunité de travailler avec toi en post-doc. Je viens seulement de m'installer à Paris et j'ai déjà découvert beaucoup de nouvelles choses. J'ai hâte de découvrir plus en détails tous les secrets de l'approximation stochastique.

I am very grateful for all the people of the Magnet team. Working in the team was a real pleasure. Thank you for your friendliness, for the extended discussions around the coffee machine, for all the lunches shared all together, and most importantly, for the interminable games of baby-foot. Thank you for all the evenings spent discussing life, science, and all our doubts around a drink at les Sarrazins, le Café Jean, and so many other places that Lille has to offer.

Merci à tous·tes les ami·es lillois·es que j'ai rencontré·es durant ces années à Lille. C'est très certainement grâce à vous que j'ai pu garder le moral tout ce temps. C'était un délice de partager tous ces mardis soirs à la Moulinette, avec le prétexte parfait qu'était l'AMAP. Les vendredis soirs à jouer aux fléchettes place Casquette vont

certainement me manquer, tout comme toutes ces soirées impromptues, dont seul le “village” qu’est Wazemmes a le secret. Tant pour les karaokés que pour les frites au bord de la Deûle. Promis, je repasserai et on pourra de nouveau s’y croiser, par hasard ou non.

Bien entendu, je remercie tous·tes les ami·es que j’ai rencontré·es au cours de ma vie. C’est toujours un plaisir de croiser chacun·e d’entre vous. La liste serait trop longue pour vous nommer ici, mais vous vous reconnaîtrez. Pour bon nombre d’entre vous, on va pouvoir se voir plus souvent à Paris ! Je me dois toutefois de nommer et de remercier “Abracaladoua”, on s’est soutenu·es chaque jour pendant la thèse, et je n’ai nul doute que l’on va continuer ainsi pour toujours.

Enfin, je remercie ma famille. En particulier, merci à mes parents et ma soeur qui m’ont toujours soutenu et écouté dans la plus grande bienveillance, c’est avant tout grâce à vous que j’en suis là.

Merci à Sabine de partager ma vie et d’avoir été présente pour moi depuis si longtemps.

Contents

Chapter 1

Introduction

The past few decades have been marked by unprecedented advances in artificial intelligence, driven by machine learning. This success is due to the remarkable alignment of three factors: the development of more expressive model architectures, together with a gigantic increase in computing power, and, most importantly, the availability of voluminous data. This has led to the utilization of machine learning in many domains. Machine learning has notably become the backbone of many industrial products, ranging from recommendations in social networks to fraud detection or self-driving cars. It is also at the core of many important scientific discoveries across many fields of research like medicine, pharmacy, social sciences, and many others.

In most of these applications, there is a serious tension between the importance of the possible discoveries, the privacy of individuals whose data is used for training models, and the fairness of the predictions. Indeed, it is now well-known that machine learning models trained on sensitive data tend to leak confidential information. Similarly, usual model training procedures often transcribe and amplify underlying discrimination in the data, and can even create new sources of discrimination. While machine learning-enabled discoveries can be very profitable for humanity (*e.g.,* discovery of new drugs or risk management), failing to address these privacy and fairness issues can have dramatic consequences on individuals (*e.g.,* blackmailing, discrimination, public shame, and, in extreme cases, fatality). The increased awareness of the risks incurred by using sensitive data at such a large scale has given birth to the fields of privacy-preserving and fair machine learning.

To preserve data privacy, new algorithms for training machine learning models have emerged. These algorithms are designed to guarantee a robust notion of privacy, that has now become standard: differential privacy. Yet, training models in this way ineluctably results in more imprecise models. There is thus a trade-off between data privacy and the models' utility. This trade-off is harsh, and in many cases (*e.g.,* for high-dimensional models), existing algorithms have trouble learning useful models

while guaranteeing meaningful privacy. Furthermore, there are growing concerns that these differentially private training algorithms may result in disparate impact, which could exacerbate discrimination even more.

In this thesis, we explore how structural properties of the problem at hand influence the privacy-utility trade-off, and the impact of enforcing privacy on the fairness of predictions in machine learning. We propose new differentially private training methods based on coordinate descent. These methods can improve the privacy-utility trade-off beyond known lower bounds by exploiting structural properties like imbalance in the model's parameters or sparsity of the solution. We then study the impact of privacy on fairness and derive upper bounds on the difference in fairness between private and non-private models.

1.1 Context on Supervised Learning

In this thesis, we focus on supervised learning. In this paradigm of machine learning, we aim at predicting a label based on some features. This is a very general framework, as labels can take many different forms. It includes many different tasks, such as:

- classification: categorical labels (*e.g.,* identifying fractured bones on an X-ray),
- regression: continuous labels (*e.g.,* predicting the price of a house),

and some other problems, notably in computer vision (*e.g.,* image segmentation), that we do not consider in this thesis.

To make this prediction, we train a model on a set of training data. This data contains labeled records (*i.e.,* pair of features and label), that all describe the same unknown underlying phenomenon. We assess the soundness of a model's prediction on a training record with a *loss function*. This function measures how close the model's predicted label is to the true label: it is small if the predicted label is right, and high otherwise. For instance, in regression tasks, this can simply be the square of the difference between these two labels.

To measure the fitness of a model on a training dataset, we use the average value of the loss function over each record. This value is called the *empirical risk*, and it gives a measure of utility of the model: a small empirical risk means that the model's predictions on the training data are, on average, good.

Empirical Risk Minimization. Learning a model amounts to finding a model that has a small empirical risk. Typically, we define an *hypothesis class*, that is a set of models (*e.g.,* linear models), and search for the one whose empirical risk is minimal.

This process is called *empirical risk minimization*. To solve this problem, we generally chose a parametric hypothesis class, and we optimize over the parameters of this class. Since we minimize the *empirical* risk, it may happen that learned models overfit the data (*i.e.,* they memorize training data, but have poor performance on unseen data). A common practice to avoid overfitting is to add a *regularization term* to the empirical risk and minimize this regularized version. This also allows for enforcing desirable structural properties (*e.g.,* sparsity) on the learned model.

The regularized empirical risk minimization formulation encompasses many different problems. It can notably be instantiated to Ridge regression, LASSO, (dual) SVM, or deep neural networks.

Minimizing the (Regularized) Empirical Risk. The regularized empirical risk minimization is a composite optimization problem with two terms: the empirical risk, and the regularizer. These two terms have different regularity properties. The former is differentiable, and the latter is simple enough so that it can be dealt with proximal (projection-like) tools. The most widely used algorithm for solving this problem is surely (proximal) *gradient descent*, and its stochastic variant. This algorithm starts with a random model from the hypothesis class and iteratively refines it by updating its parameters. At each iteration, it computes the gradient of the loss (or a stochastic estimate) at the current iterate and uses it to improve the model.

Some problem have particular structure, that can be leveraged. Parts of the models may be more important than others, or have a different scale.Unfortunately, gradient descent is indifferent to these structural properties. As such, updating the model part by part may help to grasp its structural properties more finely. This has sparked interest in *coordinate descent* methods, that are capable of exploiting this structure. These methods are indeed extremely efficient on problems where coordinates have different scales, where gradient descent often struggles to make any progress.

These coordinate descent methods are the center of this thesis. In particular, we will show that their aforementioned properties can help to find better solutions when the privacy of data matters.

1.2 The Challenge of Privacy-Preserving Machine Learning

Privacy issues did not start with machine learning. They inevitably arise when collecting and processing personal data. Quantifying the information leakage incurred by releasing the result of a computation on a database has thus been at the core of a multitude of works. One of these proposed *differential privacy*, which is now

well-adopted, and generally recognized as a very robust measure of privacy.

Differential Privacy. Differential privacy emerged from the idea that releasing the result of a computation on a database should not reveal too precisely whether a specific individual was part of the database or not. The privacy leakage is measured by looking at how the probability of observing a given output is impacted when a record of the database is replaced by another. Differential privacy requires such replacement not to affect this probability by more than a constant multiplicative factor, that is parameterized by a value called the *privacy budget*.

Given the above intuition, we see that any (non-trivial) data-dependent deterministic algorithm cannot achieve any differential privacy guarantee. Therefore, to satisfy differential privacy, randomness must be incorporated into the algorithm. This ensures that an external observer cannot know too confidently if what they observe is due to this randomness, or to the content of the database. Of course, this process reduces the quality of the answer. While an observer will indeed not be able to reconstruct the sensitive information, the result will be imprecise. This highlights the fundamental tension between privacy and utility[1]: this is generally referred to as the *privacy-utility* trade-off. This trade-off can be seen as the answer to the following question: *under a fixed privacy budget, what is the best utility that can be achieved?*

Differentially Private Empirical Risk Minimization. Training a machine learning model on a dataset is typically done using optimization algorithms that iteratively query a database, and use the result of these queries to find a good model. As such, machine learning suffers from the curse described above: to release a useful model, sensitive information *must* be leaked.

In our supervised learning setting, differentially private optimization algorithms have been proposed for solving the empirical risk minimization problem. Notably, the differentially private variant of (stochastic) gradient descent is widely used in practice. This algorithm works similarly to the (stochastic) gradient descent algorithm, except that, at each iteration, it adds noise to the gradient before performing the gradient step. This guarantees that the algorithm is differentially private. Of course, this has an impact on its utility, and, as for any data-dependent computation, finding the exact result with a differentially private algorithm is not possible.

Privacy-Utility Trade-Off in Machine Learning. The privacy-utility trade-off of differentially private empirical risk minimization has been extensively studied. Tight lower bounds have been derived for the best possible utility (measured as the

[1]There, the term "utility" is a generic measure of the precision of the result. Depending on the applications, there exist different ways of measuring it.

excess empirical risk) under a given privacy budget. Therefore, for *any* differentially private algorithm solving the empirical risk minimization problem, there exists a problem for which the utility of the algorithm necessarily decreases *polynomially* in the number of parameters of the model, but improves as the number of training records increases. Notably, the utility achieved by differentially private (stochastic) gradient descent matches these lower bounds.

However, these lower bounds are *worst-case* bounds and hold under very general assumptions. It may therefore be possible to achieve better utility on *some* problems that satisfy additional assumptions. In a sense, and similarly to how choosing the right method for the right problem is important for computational efficiency, *choosing the right method for the right problem is also crucial for using the privacy budget efficiently.* This observation is at the core of the methods we explore in this thesis.

Interplay between Privacy and Fairness. In additional to privacy issues, concerns about fairness of machine learning have risen in the past decade. Fairness of a model's prediction can be measured in different ways, depending on the nature of the task. One category of fairness notions is *group fairness*, that measure discrepancies in a model's performance (for some metric) on different groups of the population.

Multiple factors can cause unfair predictions, from discrimination in the data collection process, to inappropriate algorithm design. Training models under differential privacy affects the predictions of the model, and may thus be one of these factors. This is often called the *disparate impact of differential privacy.* To this day, it is still unclear whether this disparate impact is fundamental in differentially private machine learning, or if it is due to the design of current differentially private training methods.

1.3 Contributions

This thesis explores new optimization algorithms, that can be used for training machine learning problems in a differentially private way. While existing algorithms can, in theory, learn differentially private models optimally (in terms of the privacy-utility trade-off), there are still many problems where learning a non-trivial model privately is difficult. We argue that some problems have a particular structure, that can be exploited to obtain better private models under the same privacy budget.

As concerns about the disparate impact of differential privacy are growing, we also investigate the impact of differential privacy on fairness. In particular, we derive an upper bound on this impact, and show that, depending on the problem structure, this upper bound can give meaningful guarantees.

This thesis is thus dedicated to the study of differentially private machine learning, where we aim at exploring what can be done when more is known about the problem

than the usual, very general, assumptions. In short, we aim at answering the following question:

> *How can structural properties of machine learning problems can be exploited to improve the privacy-utility trade-off, and how do they impact the fairness of the resulting model?*

To answer this question, we start by designing and analyzing two new differentially private algorithms for solving the empirical risk minimization problem. These two algorithms are variants of the *coordinate descent* algorithm, with two different rules for selecting the coordinate to update: *random selection*, and *greedy selection*. We show that these algorithms can improve utility by exploiting structural properties of the problem like imbalancedness of the gradient coordinates or sparsity of the solution. We then turn to study the fairness of privately learned models. To this end, we show that *many group fairness notions are pointwise Lipschitz*, and use this property to derive guarantees on the difference between fairness between private models and their non-private counterparts.

1.4 Outline of the Thesis

The first two chapters introduce the mathematical background that we use throughout the thesis.

- In Chapter 2 we introduce the empirical risk minimization problem. We describe the convexity and (coordinate-wise) smoothness assumptions, and describe proximal (stochastic) gradient descent and proximal coordinate descent. We discuss the convergence properties of these algorithms for composite problems, and explain how coordinate descent is able to exploit coordinate-wise smoothness to converge faster than gradient descent.
- Then, Chapter 3 turns to the differentially private variant of empirical risk minimization. We present differential privacy, and show how a differentially private variant of stochastic gradient descent can be used to solve empirical risk minimization privately. We describe existing lower bounds on the privacy-utility trade-off of differentially private empirical risk minimization, and show that, under the usual assumptions, it is notably matched by the differentially private stochastic gradient descent algorithm.

The next three chapters of this thesis describe our three contributions. Chapters 4 and 5 are dedicated to differentially private coordinate descent methods, and Chapter 6 studies the interplay of differential privacy with fairness.

- Chapter 4 introduces differentially private proximal coordinate descent. At each iteration, one coordinate is randomly selected and updated with a noisy proximal gradient step. These noisy updates allow the algorithm to satisfy differential privacy. The utility of this algorithm is analyzed, and we show that it can adapt to the coordinate-wise smoothness of the objective function to outperform differentially private stochastic gradient descent.

- Chapter 5 studies differentially privacy *greedy* coordinate descent. At each iteration, one coordinate is selected greedily as the (noisy) largest entry of the gradient. We analyze its utility on smooth objective, and show that this selection rule allows to reduce the dependence of utility on the dimension from polynomial to logarithmic *in unconstrained problems*. This notably happens when the structure of the problem is favorable (*e.g.,* for problems with sparse solutions, or imbalanced coordinates): the algorithm can exploit this structure to beat the (general) lower bounds. Importantly, this phenomenon arises without constraints on the problem, which shows that our algorithm automatically adapts to the underlying structure of the problem.

- Chapter 6 is devoted to the interplay between differential privacy and fairness. We show that the accuracy (on a part of the population) of a model is "pointwise" Lipschitz, and that this property is inherited by multiple group fairness notions. This allows to derive an upper bound on the difference of fairness between *any* pair of models. We then use this regularity property to show that the fairness of private models necessarily stays in a bounded region around the one of their non-private counterparts.

Finally, Chapter 7 concludes this work, summarizing our contributions as an answer to the question stated in the previous section. We also describe some perspectives that we find promising for future research.

1.5 List of Publications

This thesis is built around these three publications:

- P. Mangold, A. Bellet, J. Salmon, and M. Tommasi (June 2022). "Differentially Private Coordinate Descent for Composite Empirical Risk Minimization". In: *Proceedings of the 39th International Conference on Machine Learning*. PMLR, pp. 14948–14978.
- P. Mangold, A. Bellet, J. Salmon, and M. Tommasi (Apr. 2023a). "High-Dimensional Private Empirical Risk Minimization by Greedy Coordinate Descent". In: *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 4894–4916.
- P. Mangold, M. Perrot, A. Bellet, and M. Tommasi (Jan. 2023b). "Differential Privacy Has Bounded Impact on Fairness in Classification". In: *Proceedings of the 40th International Conference on Machine Learning*. PMLR.

The following papers have also been published during this thesis, but are excluded from this manuscript:

- P. Mangold, A. Filiot, M. Moussa, V. Sobanski, G. Ficheur, P. Andrey, and A. Lamer (Nov. 2020). "A Decentralized Framework for Biostatistics and Privacy Concerns". In: *Studies in Health Technology and Informatics*. Ed. by A. Värri, J. Delgado, P. Gallos, M. Hägglund, K. Häyrinen, U.-M. Kinnunen, L. B. Pape-Haugaard, L.-M. Peltonen, K. Saranto, and P. Scott. IOS Press.
- A. Lamer, A. Filiot, Y. Bouillard, P. Mangold, P. Andrey, and J. Schiro (May 2021). "Specifications for the Routine Implementation of Federated Learning in Hospitals Networks". In: *Studies in Health Technology and Informatics*. Vol. 281, pp. 128–132.
- J. O. du Terrail, S.-S. Ayed, E. Cyffers, F. Grimberg, C. He, R. Loeb, P. Mangold, T. Marchand, O. Marfoq, E. Mushtaq, B. Muzellec, C. Philippenko, S. Silva, M. Teleńczuk, S. Albarqouni, S. Avestimehr, A. Bellet, A. Dieuleveut, M. Jaggi, S. P. Karimireddy, M. Lorenzi, G. Neglia, M. Tommasi, and M. Andreux (Oct. 2022). "FLamby: Datasets and Benchmarks for Cross-Silo Federated Learning in Realistic Healthcare Settings". In: *Thirty-Sixth Conference on Neural Information Processing Systems*.
- H. Hendrikx, P. Mangold, and A. Bellet (2023). "The Relative Gaussian Mechanism and its Application to Private Gradient Descent". In: *arXiv preprint arXiv:2308.15250*.

Chapter 2

Background on Convex Optimization in Machine Learning

In supervised learning, models are often trained through empirical risk minimization. The goal of this problem is to find a model that minimizes the average of a *loss function* (*i.e.,* a function that evaluates the error of a model) on given training dataset. It is an optimization problem over a space of models, that we call the *hypothesis class*. This hypothesis class is generally parameterized by a real-valued vector, reducing the problem to finding the parameters of the best model. Therefore, we turn to the study of algorithms that solve (composite) finite-sum problems of the following form:

$$\min_{w \in \mathcal{W}} \left\{ F(w) := f(w) + \psi(w) \right\} \ , \quad \text{where } f(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w) \ , \tag{$\star$}$$

where $\mathcal{W} \subset \mathbb{R}^p$ is a set, and $f_i : \mathcal{W} \to \mathbb{R}$ (for $i \in [n]$) and $\psi : \mathcal{W} \to \mathbb{R}$ are functions. In machine learning applications, f_i is the loss function on the i-th data record, and ψ is a regularization term, that can be used to enforce some structure on the model.

In this chapter, we give an overview of the optimization algorithms that are generally used for solving machine learning problems that fit in the framework of ($\star$), under the assumptions that $\mathcal{W}$ is closed and convex, each f_i (for $i \in [n]$) is proper convex and smooth, and ψ is convex (and not necessarily differentiable). We describe these assumptions, as well as their most important properties, in Section 2.1. We then give in Section 2.2 an overview of first order methods for solving ($\star$) under these assumptions. We choose to focus on first-order methods since these are the most usual choice in machine learning applications, where at the number n of functions in the finite sum f and the number p of parameters in the model are often large.

2.1 Functions Regularity

The type of algorithms we use for solving problems like ($\star$) primarily depends on the properties of the functions themselves. In this thesis, we *always* assume that, for $i \in [n]$, f_i is convex and smooth, and that ψ is convex. This gives us a diverse set of tools that we can exploit to design and analyze efficient algorithms. This section is devoted to the description of these tools and their uses.

2.1.1 Differentiability, Gradient and Jacobian

The first indispensable tool that we need is differentiability. This allows approximating the local behavior of a function with a linear function. Such functions can be studied using their *differential function*, which we define in this section. We refer to Fleming (2012) and Garling (2014) for more details on differentiable functions.

In all the following, $p, k > 0$ are two integers, and for all integers $n > 0$, we denote $e_1, \dots, e_n$ the standard basis of $\mathbb{R}^n$. We also denote $\mathcal{W} \subseteq \mathbb{R}^p$ a subset of $\mathbb{R}^p$. The differential function df of a function $f : \mathbb{R}^p \to \mathbb{R}^k$ is defined as follows.

Definition 2.1.1 (Differentiable function)**.** *A function $f : \mathcal{W} \to \mathbb{R}^k$ is differentiable at a point $w \in \mathcal{W}$ if there exists a linear function $df_w : \mathcal{W} \to \mathbb{R}^k$ such that*

$$\lim_{h \to 0} \frac{\|f(w+h) - f(w) - df_w(h)\|}{\|h\|} = 0 \ . \tag{2.1.1}$$

When f is differentiable on all its domain $\mathcal{W}$, we say that f is differentiable, and we define its differential function as $df : w \to df_w$.

For $j \in [p]$, the partial derivative of f in the direction of e_j is $\frac{\partial f}{\partial x_j} : w \mapsto df(w)(e_j)$. These partial derivatives fully characterize the differential of f. In particular, when a function $f : \mathbb{R}^p \to \mathbb{R}$ is real-valued, its differential function is a linear form. At any point $w \in \mathcal{W}$, $df(w)$ can be expressed as a dot product with a specific vector. We call this vector the gradient of f at w.

Definition 2.1.2 (Gradient)**.** *Let $f : \mathcal{W} \to \mathbb{R}$ be a differentiable real-valued function. The gradient $\nabla f(w)$ of f at $w \in \mathcal{W}$ is the only vector such that, for all $h \in \mathbb{R}^p$, $df(w)(h) = \langle \nabla f(w), h \rangle$. The coefficients of $\nabla f(w)$ are the partial derivatives of f, and we denote $\nabla_j f(w) = \frac{\partial f}{\partial w_j}(w)$ the j-th coefficient of $\nabla f(w)$:*

$$\nabla f(w) = \left(\frac{\partial f}{w_1}(w), \dots, \frac{\partial f}{w_p}(w) \right) \in \mathbb{R}^p \ . \tag{2.1.2}$$

For vector-valued functions, the notion of gradient can be naturally extended by constructing a matrix whose lines are the gradients of each coordinate of the function. This matrix is called the Jacobian matrix.

Definition 2.1.3 (Jacobian and Hessian Matrix)**.** *Let $f : \mathbb{R}^p \to \mathbb{R}^k$ be a differentiable function, and, for $i \in [k]$, we denote $f_i = e_i^\top f$ the i-th coordinate of f. For all $w \in \mathbb{R}^p$, the Jacobian matrix $Jf(w)$ of f is the matrix of the linear map $df(w)$ in the standard bases of $\mathbb{R}^p$ and $\mathbb{R}^k$. Its coefficients are $J_{i,j}f(w) = df_i(w)(e_j) = \frac{\partial f_i}{\partial x_j}(w)$. Specifically,*

$$Jf(w) = \begin{pmatrix} \nabla f_1(w) \\ \vdots \\ \nabla f_k(w) \end{pmatrix} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1}(w) & \frac{\partial f_1}{\partial x_2}(w) & \dots & \frac{\partial f_1}{\partial x_p}(w) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_k}{\partial x_1}(w) & \frac{\partial f_k}{\partial x_2}(w) & \dots & \frac{\partial f_k}{\partial x_p}(w) \end{pmatrix} \in \mathbb{R}^{k \times p} \ . \tag{2.1.3}$$

When f is twice differentiable, we define the Hessian $\nabla^2 f$ of f as the transposed of the Jacobian of ∇f: $\nabla^2 f = J(\nabla f)^\top$.

Gradients lie at the core of first-order optimization algorithms. We will see in Section 2.2 and throughout the thesis that they are a key component of *gradient descent*, *coordinate descent* and their differentially private variants.

2.1.2 Mahalanobis Norms

Before jumping to convexity and smoothness, which are the two essential properties that we will use throughout this thesis, we need to define a way of measuring the ambient space. The most usual way of doing so is to use ℓ_q-norms:

$$\|w\|_q = \left(\sum_{j=1}^{p} |w_j|^q \right)^{1/q} , \qquad \text{for all } w \in \mathbb{R}^p, \text{ and } q \geq 0 \ . \tag{2.1.4}$$

The ℓ_1, ℓ_∞, and ℓ_2 norms will be at the core of the theory we develop in this thesis: they will serve to measure functions' regularity, and they will allow us to analyze the convergence of all the algorithms we study. Note that these norms measure each dimension of the space equally, but the functions we study may have different properties along each of these dimensions. To capture this, we define the following scaled norms, inspired by the work of Mahalanobis (1936).

Definition 2.1.4 (Mahalanobis Norms)**.** *Let $M_1, \dots, M_p > 0$ be positive real numbers and $M = \mathrm{diag}(M_1, \dots, M_p) \in \mathbb{R}^{p \times p}$ be a diagonal matrix. For $q \geq 0$, we define*

$$\|w\|_{M,q} = \|M^{1/2} w\|_q \ , \qquad \|w\|_{M^{-1},q} = \|M^{-1/2} w\|_q \ . \tag{2.1.5}$$

These norms account for each dimension differently, depending on the value of the M_j's. We give examples of the balls of radius 1 for various norms in Figure 2.1.1.

(a) ℓ_1-norm. (b) ℓ_2-norm.

Figure 2.1.1: ℓ_1 and ℓ_2 unit balls. In solid black lines, the usual unit ball (*i.e.,* $M = \mathbb{I}_2$ in Definition 2.1.4). In dashed purple lines, the unit balls for $M = \mathrm{diag}(0.1, 10)$.

These norms are at the core of the analysis of coordinate descent algorithms, as we will discuss in Section 2.2.3, Chapters 4 and 5.

Norms can be grouped by pairs, that we call conjugate (or dual) norms. The conjugate norm of a norm $\|\cdot\|$ is defined as

$$\|w\|^* = \sup\{\langle w, x\rangle \mid \|x\| \leq 1\} \ , \tag{2.1.6}$$

where $\langle \cdot, \cdot \rangle$ is usual euclidean dot product. An important special case is the (scaled) ℓ_q-norms, whose conjugate norm is

$$\|\cdot\|_{M,q}^* = \|\cdot\|_{M^{-1},q'} \ , \qquad \text{where } q' \text{ is such that } \tfrac{1}{q} + \tfrac{1}{q'} = 1 \ . \tag{2.1.7}$$

Conjugate norms are related to the usual Euclidean dot product through Hölder's inequality, which is a direct consequence of their definition as (2.1.6):

$$\langle w, w' \rangle \leq \|w\| \cdot \|w'\|^* \ , \qquad \text{for all } w, w' \in \mathbb{R}^p \ . \tag{2.1.8}$$

This inequality reduces to the Cauchy-Schwarz inequality when $\|\cdot\| = \|\cdot\|^* = \|\cdot\|_2$. It will be very useful in Section 2.2.4 and Chapter 5, where we study greedy coordinate descent algorithms.

2.1.3 Convex Sets and Convex Functions

We now turn to the study of a first type of regularity: convexity. In the following, (strong) convexity will play two major roles. First, it ensures that any extremal point of a function is a minimum. Second, it provides global linear (or quadratic) lower bounds on the function, which is a crucial property for the formal analysis of convex optimization algorithms. In the rest of this section, we define convex sets and convex functions, as well as the important properties that will be used throughout the thesis.

2.1.3 (a) Convex Sets

Let $\mathcal{W} \subseteq \mathbb{R}^p$ be a subset of $\mathbb{R}^p$. It is convex if all segments between any two points of $\mathcal{W}$ is included in $\mathcal{W}$. Formally, this means that for all $w, w' \in \mathcal{W}$ and $\lambda \in [0, 1]$, $(1 - \lambda)w + \lambda w' \in \mathcal{W}$.

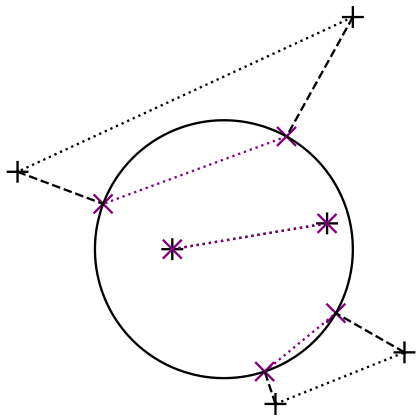


Figure 2.1.2: Projection on a convex set. Black "+" are initial points, and purple "x" are their projections. Pairs of projected points are always closer than the initial ones.

Whenever a point $w \in \mathbb{R}^p$ is not in $\mathcal{W}$, it can be projected on $\mathcal{W}$ using the following projection operator

$$\Pi_{\mathcal{W}}(w) = \operatorname*{arg\,min}_{z \in \mathcal{W}} \|z - w\|_2^2 \ , \tag{2.1.9}$$

where $\|\cdot\|_2$ is the usual ℓ_2-norm. By definition, for any $w \in \mathcal{W}$, $\Pi_{\mathcal{W}}(w)$ returns the element of $\mathcal{W}$ that is the closest to w, and this element is unique (see *e.g.,* Theorem 3.1.10 in Nesterov, 2004). These projection operators play a central role in constrained convex optimization (*i.e.,* $\mathcal{W} \neq \mathbb{R}^p$ in ($\star$)). Most of the algorithms we will study indeed use a projection step to meet to constraint. The crucial property of $\Pi_{\mathcal{W}}$ that makes these methods work, is its non-expansiveness.

Proposition 2.1.1 (Non-Expansiveness of Convex Projection)**.** *Let $\mathcal{W} \subseteq \mathbb{R}^p$ be a closed convex set, and $w, w' \in \mathbb{R}^p$, then*

$$\|\Pi_{\mathcal{W}}(w) - \Pi_{\mathcal{W}}(w')\| \leq \|w - w'\| \ .$$

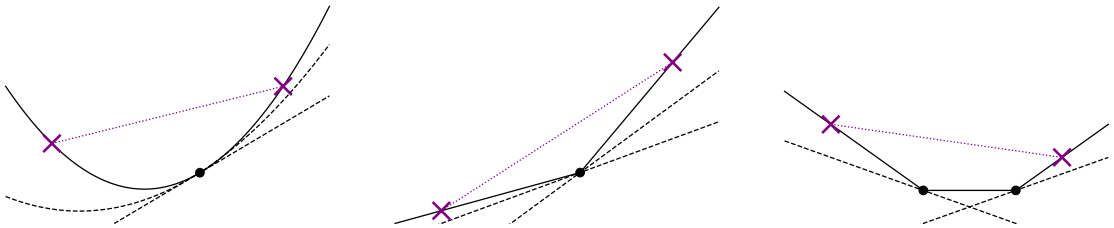
We will prove this property when we introduce the proximity operator (see Section 2.1.5), that can be seen as a generalization of the convex projection. We also give several geometric examples of this property in Figure 2.1.2.

2.1.3 (b) Convex Functions

Let $f : \mathbb{R}^p \to \mathbb{R} \cup \{\pm\infty\}$ be a (extended-)real-valued function. The domain of f is

$$\operatorname{dom}(f) = \{w \in \mathbb{R}^p \mid f(w) < +\infty\} \ .$$

When $\operatorname{dom}(f)$ is not empty and f does not take the value $-\infty$, we say f is *proper*. We are now ready to introduce the notion of *convex functions*.



(a) Strongly-convex function with one minimizer.

(b) Convex function without minimizers.

(c) Convex function with multiple minimizers.

Figure 2.1.3: Example of convex functions. The dotted purple line is a chord of f and is always above f. The dashed black lines are lower bounds that follow from (strong) convexity of f. In Figures 2.1.3b and 2.1.3c, the slope of these lower bounds are elements of the subdifferential ∂f of f.

Definition 2.1.5 (Convex and strongly-convex function)**.** *Let $\mu \geq 0$, and $\|\cdot\|$ be any norm. A function $f : \mathbb{R}^p \to \mathbb{R}$ is proper μ-strongly convex w.r.t. the norm $\|\cdot\|$ if f is proper,* $\mathrm{dom}(f)$ *is convex, and for all $w, w' \in \mathbb{R}^p$ and $\lambda \in [0,1]$,*

$$f((1-\lambda)w + \lambda w') \leq (1-\lambda)f(w) + \lambda f(w') + \tfrac{\mu}{2}\lambda(1-\lambda)\|w - w'\|^2 \ , \tag{2.1.10}$$

If f is differentiable, the above property is equivalent to

$$f(w') \geq f(w) + \langle \nabla f(w), w - w' \rangle + \tfrac{\mu}{2}\|w - w'\|^2 \ , \tag{2.1.11}$$

If f is twice differentiable, it is also equivalent to

$$\nabla^2 f \succcurlyeq \mu \mathbb{I}_p \ . \tag{2.1.12}$$

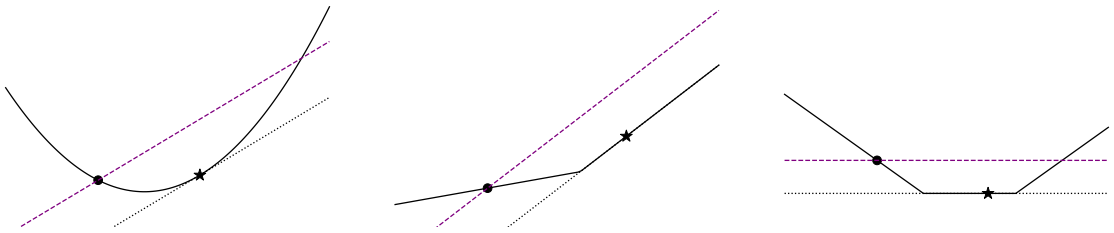
When $\mu = 0$, we simply say that f is convex.

We illustrate the first two definitions (2.1.10) and (2.1.11) in Figure 2.1.3. In Figure 2.1.3a, the function is twice differentiable and strongly-convex, and satisfies all three definitions with $\mu > 0$. In Figures 2.1.3b and 2.1.3c the functions are convex ($\mu = 0$) but not differentiable, and have respectively no minimums and an infinite number of minimums.

In the following, we will essentially use inequality (2.1.11), which provides a linear (or quadratic if $\mu \neq 0$) lower bound on the value of f. Sadly, this requires f to be *differentiable*, which will not always be the case (consider *e.g.,* the ℓ_1-norm). We can circumvent this limitation by defining a proxy for the gradient in (2.1.11) as follows.

Definition 2.1.6 (Subdifferential)**.** *Let $f : \mathbb{R}^p \to \mathbb{R}$ be a proper convex function. The subdifferential of f at a point $w \in \mathrm{dom}(f)$ is*

$$\partial f(w) = \{g \in \mathbb{R}^p \mid f(w') \geq f(w) + \langle g, w' - w \rangle \text{ for all } w' \in \mathbb{R}^p\} \ .$$



(a) Strongly-convex function with one minimizer.

(b) Convex function without minimizers.

(c) Convex function with multiple minimizers.

Figure 2.1.4: Illustration of (2.1.15), where w is represented as a star and w' as a circle. When $\nabla f(w)$ is known, we can upper bound $f(w)$ using the value of f at w' and the tangent of f at w.

The subdifferential of a (μ-strongly) convex function f is a (strongly) monotone operator: let $x, y \in \mathbb{R}^p$ and $g_x \in \partial f(x), g_y \in \partial f(y)$, then

$$\langle g_x - g_y, x - y \rangle \geq \mu \|x - y\|^2 \ . \tag{2.1.13}$$

We give examples of elements of the subdifferential of f in Figures 2.1.3b and 2.1.3c (see the slope of the dotted black lines). Even on points where f is not differentiable, the subdifferential gives a linear lower bounds of f. This property will be extremely useful when working with proximal operators for composite problems (*i.e.,* when ψ is not differentiable in ($\star$)). We discuss this in more detail in Section 2.1.5.

More generally, convexity guarantees that whenever one finds a local extremum of f, it is guaranteed to be a global minimum of f.

Proposition 2.1.2 (Minimums are Global)**.** *Let $f : \mathcal{W} \to \mathbb{R}$ be a convex function. Then the set* $\arg\min(f)$ *of minimizers of f is convex, and any local minimum of f is a global minimum. If f is strongly convex, then it has at most one minimum.*

Moreover, when f is μ-strongly convex, and has a minimizer $w^* \in \arg\min(f)$, f is uniformly lower bounded by

$$f(w) \geq f(w^*) + \tfrac{\mu}{2}\|w - w^*\|^2 \ , \qquad \text{for all } w \in \mathbb{R}^p \ . \tag{2.1.14}$$

Finally, we remark that convexity also allows deriving upper bounds on functions, by reformulating (2.1.11) as follows, possibly with $\mu = 0$,

$$f(w) \leq f(w') + \langle \nabla f(w), w' - w \rangle - \tfrac{\mu}{2}\|w - w'\|^2 \ , \qquad \text{for all } w, w' \in \mathbb{R}^p \ . \tag{2.1.15}$$

We give examples of the upper bounds we obtain in this way in Figure 2.1.5. Unfortunately, using these upper bounds to bound $f(w)$ for some $w \in \mathbb{R}^p$ requires computing

$\nabla f(w)$, which is generally not directly available. We will see in Section 2.1.5 that this inequality can be used to analyze proximity operators computed on ψ, the non-smooth part of ($\star$). To analyze algorithms that work on the full composite problem, we will need more assumptions on f, the differentiable part of ($\star$).

2.1.4 Lipschitzness and Smoothness

2.1.4 (a) Lipschitzness and Sensitivity

A simple assumption to obtain an upper bound on a function f is to use linear upper bounds. This is commonly called Lipschitzness and is defined as follows.

Definition 2.1.7 (Lipschitzness)**.** *A function $f : \mathbb{R}^p \to \mathbb{R}^k$ is L-Lipschitz with respect to two norm $\|\cdot\|$ and $\|\cdot\|_f$ on $\mathbb{R}^p$ and $\mathbb{R}^k$ if for all $w, w' \in \mathrm{dom}(f)$,*

$$\|f(w) - f(w')\|_f \leq L\|w - w'\| \ . \tag{2.1.16}$$

We can also measure the Lipschitzness of f along each of its parameters. We say f is $(L_1, \ldots, L_p)$-coordinate-Lipschitz for $L_1, \ldots, L_p > 0$ if for all $j \in [p]$, and $w \in \mathcal{W}$,

$$|f(w + te_j) - f(w)| \leq L_j|t| \ . \tag{2.1.17}$$

When the function f is differentiable, this Lipschitz property directly gives an upper bound on the gradient of f.

Proposition 2.1.3 (Lemma 2.6 in Shalev-Shwartz, 2011)**.** *Let $f : \mathcal{W} \subseteq \mathbb{R}^p \to \mathbb{R}^k$ be a differentiable convex function. Then f is L-Lipschitz with respect to a norm $\|\cdot\|$ if and only for all $w \in \mathcal{W}$, $\|\nabla f(w)\|^* \leq L$, where $\|\cdot\|^*$ is the dual norm of $\|\cdot\|$.*

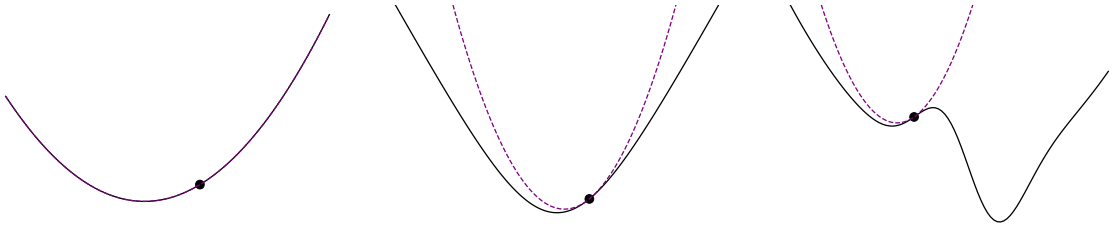
Similarly, if f is $(L_1, \ldots, L_p)$-coordinate-Lipschitz, then for $w \in \mathcal{W}$, $|\nabla_j f(w)| \leq L_j$.

This upper bound will be particularly useful in the design of differentially private optimization algorithms, that typically require a bound on the difference between two gradients (see Section 3.1). Such a bound directly follows from the Lipschitz property, since if $f : \mathcal{W} \to \mathbb{R}^k$ is L-Lipschitz w.r.t. $\|\cdot\|$, then for all $w, w' \in \mathcal{W}$,

$$\|\nabla f(w) - \nabla f(w')\|^* \leq \|\nabla f(w)\|^* + \|\nabla f(w')\|^* \leq 2L \ . \tag{2.1.18}$$

2.1.4 (b) Smoothness and Coordinate-wise Smoothness

In general, the Lipschitz assumption is too restrictive and does not provide enough information about the function. Instead of assuming the *function* f to be Lipschitz, we may assume that its *gradient* is.



(a) Quadratic smooth. (b) Convex smooth. (c) Non-convex smooth.

Figure 2.1.5: Example of convex functions together with the upper bounds (purple dashed line) we obtain using (2.1.15) with w' being the black dot.

Definition 2.1.8 (Smooth function)**.** *A differentiable function $f : \mathbb{R}^p \to \mathbb{R}$ is M-smooth w.r.t. a norm $\|\cdot\|$ if its gradient is M-Lipschitz,* i.e., *for all $w, w' \in \mathbb{R}^p$,*

$$\|\nabla f(w) - \nabla f(w')\| \leq M\|w - w'\| \ . \tag{2.1.19}$$

If f is twice differentiable, this is equivalent to $\nabla^2 f \preccurlyeq M\mathbb{I}_p$.

The most useful consequence of this assumption is that it gives a quadratic upper bound on the function f, that can be computed globally from the knowledge of the gradient of f at one point and the smoothness constant.

Proposition 2.1.4 (Quadratic Upper Bound)**.** *Let $f : \mathbb{R}^p \to \mathbb{R}$ a M-smooth function. Then for all $w, w' \in \mathbb{R}^p$,*

$$f(w') \leq f(w) + \langle \nabla f(w), w' - w \rangle + \tfrac{M}{2}\|w - w'\|^2 \ . \tag{2.1.20}$$

This property has a very important role in smooth first-order optimization. Indeed, a natural idea for finding a minimum of f is to iteratively minimize this quadratic upper bound: take a fixed $w \in \mathcal{W}$, then the quadratic upper bound from (2.1.20) is minimal when its gradient is zero, that is

$$\nabla f(w) + M(w' - w) = 0 \ ,$$

which implies that this quadratic upper bound is minimal when $w' = w - \frac{1}{M}\nabla f(w)$. This is exactly the gradient step that we will use in gradient descent for smooth functions (see Section 2.2.1).

Interestingly, the smoothness of f can be captured more tightly by measuring it on vectors that differ on only one coordinate.

Definition 2.1.9 (Coordinate-smooth function)**.** *Let $M_1, \ldots, M_p > 0$ and define $M = \operatorname{diag}(M_1, \ldots, M_p) \in \mathbb{R}^{p \times p}$. A differentiable function $f : \mathbb{R}^p \to \mathbb{R}$ is M-coordinate-smooth, if for $j \in [p]$, the j-th coordinate of its gradient is M_j-Lipschitz,*

meaning that for all $w \in \mathbb{R}^p$ *and* $t \in \mathbb{R}$,

$$\|\nabla f(w) - \nabla f(w + te_j)\| \leq M_j|t| \ . \tag{2.1.21}$$

Note that this assumption is in fact the same as smoothness, as Lipschitzness of the gradient directly implies Lipschitzness of its coordinates. But the coordinate wise constants M_j's can be much smaller than the global one. Coordinate-wise smoothness therefore simply measure the smoothness more finely along each of the coordinates of f. This allows to refine the quadratic upper bound (2.1.20) to the following, for $w \in \mathcal{W}$ and $t \in \mathbb{R}$,

$$f(w + te_j) \leq f(w) + \nabla_j f(w^t) \cdot t + \tfrac{M_j}{2}|t|^2 \ . \tag{2.1.22}$$

This upper bound will play a crucial role in the analysis of coordinate descent methods, as we will see in Section 2.2.3.

2.1.5 Proximal Operators

When the problem ($\star$) has both a smooth part f and a non-smooth part ψ, the inequality from (2.1.20) does not give an upper bound on $f + \psi$ anymore. To fix this, we may simply add ψ to each side of (2.1.20), which gives

$$f(w') + \psi(w') \leq f(w) + \langle \nabla f(w), w' - w \rangle + \tfrac{L}{2}\|w - w'\|^2 + \psi(w') \ .$$

Proceeding as above, we may want to minimize the right hand size of this inequality in w', which is minimal when $w' = \arg\min_{v \in \mathbb{R}^p} \left\{ \frac{1}{2}\|v - (w - \frac{1}{L}\nabla f(w))\|^2 + \psi(v) \right\}$. This motivates the study of *proximity operators*.

Definition 2.1.10 (Proximity Operator)**.** *Let* $\psi : \mathbb{R}^p \to \mathbb{R}$ *be a closed proper convex function. For all* $w \in \mathbb{R}^p$*, the proximal operator of* ψ *is*

$$\mathrm{prox}_{\psi}(w) = \operatorname*{arg\,min}_{v \in \mathbb{R}^p} \left\{ \tfrac{1}{2}\|v - w\|_2^2 + f(v) \right\} \ . \tag{2.1.23}$$

Two usual examples of proximal operators are

- $\psi = \iota_{\mathcal{W}}$ where $\mathcal{W}$ is a convex set and $\iota_{\mathcal{W}}$ its characteristic function. Then $\mathrm{prox}_{\iota_{\mathcal{W}}} = \Pi_{\mathcal{W}}$ is the projection operator on the set $\mathcal{W}$, as defined in (2.1.9).
- $\psi = \lambda\|\cdot\|_1$ for some $\lambda > 0$. Then $\mathrm{prox}_{\lambda\|\cdot\|_1}$ is the soft thresholding operator, which, for each coordinate j, gives $e_j^\top \mathrm{prox}_{\lambda\|\cdot\|_1}(w) = \mathrm{sign}(w_j) \cdot \max(0, |w_j| - \lambda)$.

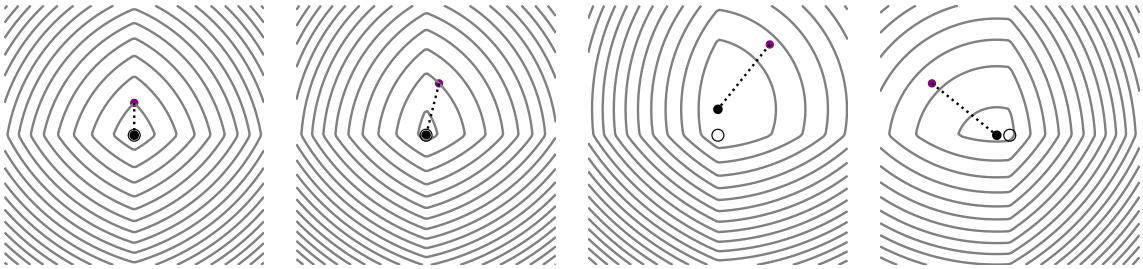


Figure 2.1.6: Objective function and solution (black point) of the optimization problem defined by the proximity operator of the ℓ_1-norm. Purple point is the initial point. Each coordinate of the point is shrinked, and is put to zero if it is small enough: this is why the ℓ_1-norm regularizer promotes sparsity.

The projection operator is the same as illustrated in Figure 2.1.2, and we illustrate the loss function solved by $\mathrm{prox}_{\lambda\|\cdot\|_1}$ in Figure 2.1.6. We refer to the Prox Repository[1] for more examples of proximity operators.

We now state the mystical property of proximity operators: they do an implicit gradient update, finding an element of the (sub)differential of a function at a point we do not know yet. This property will be crucial in every analysis of proximal algorithms, as it will allow using convexity to obtain *upper bounds* using (2.1.15).

Proposition 2.1.5 (Implicit gradient step)**.** *Let $x \in \mathbb{R}^p$, and $f : \mathbb{R}^p \to \mathbb{R}$ a convex function. There exists $\xi \in \partial\psi(\mathrm{prox}_f(x))$ such that*

$$\mathrm{prox}_f(x) = x - \xi \ . \tag{2.1.24}$$

Proof of Proposition 2.1.5. Let $x \in \mathbb{R}^p$ and $g : w \mapsto \frac{1}{2}\|x-w\|^2 + f(w)$. Note that g is strongly-convex, and has a unique minimizer $\mathrm{prox}_f(x)$, which satisfies $0 \in \mathrm{prox}_{\alpha f}(x) - x + \partial f(\mathrm{prox}_f(x))$. This guarantees the existence of a unique $\xi \in \partial f(\mathrm{prox}_f(x))$ such that $\mathrm{prox}_f(x) = x - \xi$, which is the result. □

Proposition 2.1.6 (Lemma 2.4 in Combettes and Wajs, 2005)**.** *Let $x, y \in \mathbb{R}^p$, and $f : \mathbb{R}^p \to \mathbb{R}$ a convex function. The proximal operator of f is firmly non-expansive, meaning that*

$$\|\mathrm{prox}_f(x) - \mathrm{prox}_f(y)\|^2 \leq \langle \mathrm{prox}_f(x) - \mathrm{prox}_f(y), x - y\rangle \ . \tag{2.1.25}$$

Which implies the usual non-expansiveness property

$$\|\mathrm{prox}_f(x) - \mathrm{prox}_f(y)\|^2 \leq \|x - y\|^2 \ . \tag{2.1.26}$$

[1] See http://proximity-operator.net/.

Proof of Proposition 2.1.6. By Proposition 2.1.5, we have that, for all $x, y \in \mathbb{R}^p$, $x - \operatorname{prox}_f(x) \in \partial f(\operatorname{prox}_f(x))$ and $y - \operatorname{prox}_f(y) \in \partial f(\operatorname{prox}_f(y))$. Since f is convex, it is monotone, and (2.1.13) gives

$$\langle (x - \operatorname{prox}_f(x)) - (y - \operatorname{prox}_f(y)), \operatorname{prox}_f(x) - \operatorname{prox}_f(y) \rangle \geq 0 \ , \tag{2.1.27}$$

which gives the first inequality. The second one follows from the Cauchy-Schwarz inequality. □

2.2 Convex Optimization

In this section, we describe the algorithms that are most widely used when solving Problem ($\star$) in machine learning applications. In these problems, datasets tend to be large, and models to be high-dimensional. This has oriented the community towards the use and study of first-order algorithms, and some of their variants. Recall that we aim at studying problems of the following form

$$\min_{w \in \mathcal{W}} \{F(w) := f(w) + \psi(w)\} \ , \quad \text{where } f(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w) \ , \tag{$\star$}$$

When F is differentiable, the most iconic algorithm for solving this problem is *gradient descent*, initially proposed by Cauchy (1847) and Hadamard (1908). This algorithm iteratively refines an initial guess $w^0 \in \mathbb{R}^p$ as follows

$$w^{t+1} = w^t - \gamma \nabla F(w^t) \qquad \text{for } t \geq 0 \ , \tag{GD}$$

where $\gamma > 0$ is a step size. The first analysis of this algorithm has been done by Curry (1944), who showed that GD asymptotically converges to a stationary point of F. The method has then been extensively studied (see for instance Himmelblau, 1972; Kantorovich and Akilov, 1982; Polyak, 1987). A prominent special case of this problem is when $\mathcal{W}$ and F are convex. Numerous books have been dedicated to this special case. The most classical ones date from late 20th or early 21st centuries (Rockafellar, 1970; Nesterov and Nemirovskii, 1994; Nesterov, 2004; Boyd and Vandenberghe, 2004), and more recent ones (Bubeck, 2015; Nesterov, 2018; Beck, 2017; Wright and Recht, 2022; Ryu and Yin, 2022).

In this thesis, we focus on these types of convex problems. We assume that:

- **(A1)** The set $\mathcal{W}$ is closed and convex.
- **(A2)** The function f is proper convex and smooth.
- **(A3)** The function ψ is convex (and not necessarily differentiable).

Note that we do not assume that ψ is differentiable: `GD` in itself is therefore not applicable. Fortunately, this can be dealt with proximity operators, that we introduced in Section 2.1.5, using the proximal gradient descent algorithm. In the remainder of this chapter, we will describe and analyze this algorithm, as well as two important variants. The first one is a stochastic algorithm, where the gradient ∇f is estimated approximately. Such variants are widely used when training models on very large datasets. The second one is a coordinate-wise variant, where we compute only one coordinate of the gradient at a time: these algorithms will be at the core of this thesis.

2.2.1 Proximal Gradient Descent

When the function ψ in $(\star)$ is not differentiable, using `GD` is naturally not possible. To remedy this, it is tempting to replace the gradient with a *subgradient* of ψ at the current point w^t. This is known as the *subgradient method*, and was proposed by Shor (1962) (see Polyak (1977) for more details). Unfortunately, this method generally suffers from a slow convergence rate, and tends to lose the structural properties we aim to enforce using the regularizer (like sparsity when ψ is the ℓ_1-norm of the parameters).

These rates can typically be improved if, instead of updating using a subgradient of ψ at w^t, we use a subgradient of ψ *at the next point* w^{t+1}. While this certainly seems crazy at first, this is what proximal gradient descent does! It does so by exploiting the implicit gradient update property of proximity operators from Proposition 2.1.5.

The exact formulation of the proximal gradient algorithm builds on two ideas. The first the use proximity operators for minimizing functions. This is the idea of the proximal point algorithm, which was proposed by Martinet (1970) and Martinet (1972), and studied by Rockafellar (1976). However, such methods may be oblivious to the regularity of f. This has led Passty (1979) and Bruck (1977) to consider forward-backward splitting schemes, that benefit from the best of both worlds. In these methods, we do a gradient step, followed by an implicit update through a proximity operator. This gives the `PGD` algorithm, that we list here as Algorithm 2.2.1.

Algorithm 2.2.1: `PGD`: Proximal Gradient Descent.

Input: initial point w^0, step size γ.

For $t = 0$ to $T - 1$:

$$w^{t+1} = \operatorname{prox}_{\gamma\psi}\left(w^t - \gamma \nabla f(w^t)\right)$$

Return: w^T.

Interestingly, PGD converges at a rate that is similar to GD on smooth functions. Beck and Teboulle (2009) proved that PGD converges at a rate $1/t$ on convex problems, and Schmidt et al. (2011) and Karimi et al. (2016) showed that under strong convexity, PGD converges linearly. We state the results more precisely in Theorem 2.2.1.

Theorem 2.2.1. *Assume f is M-smooth, and let w^* be a minimizer of F. Let w^t be the iterates of Algorithm 2.2.1 with step size $\gamma \leq 1/M$. Then, for general convex objectives (see Theorem 3.1 in Beck and Teboulle, 2009):*

$$F(w^t) - F(w^*) \leq \frac{\|w^0 - w^*\|^2}{2\gamma t} \ . \tag{2.2.1}$$

If, additionally, F is strongly convex, then (see Proposition 3 in Schmidt et al., 2011; or Theorem 3.5 in Garrigos and Gower, 2023)

$$\|w^t - w^*\|^2 \leq (1 - \tfrac{\gamma\mu}{2})^t \|w^0 - w^*\|^2 \ . \tag{2.2.2}$$

Proof. We start by expanding the norm

$$\begin{aligned} \|w^{t+1} - w^*\|^2 &= \|w^t - w^*\|^2 + \langle w^{t+1} - w^t, w^t - w^* \rangle + \|w^{t+1} - w^t\|^2 \\ &= \|w^t - w^*\|^2 + \langle w^{t+1} - w^t, w^{t+1} - w^* \rangle - \|w^{t+1} - w^t\|^2 \ . \end{aligned} \tag{2.2.3}$$

From Proposition 2.1.5, there exists $\xi^t \in \partial\psi(w^{t+1})$ (where $\partial\psi$ is the subdifferential of ψ, see Definition 2.1.6) such that $w^{t+1} = w^t - \gamma(\nabla f(w^t) + \xi^t)$. We replace w^{t+1} by its value in (2.2.3) to obtain

$$\|w^{t+1} - w^*\|^2 = \|w^t - w^*\|^2 - 2\gamma\langle \nabla f(w^t) + \xi^t, w^{t+1} - w^* \rangle - \|w^{t+1} - w^t\|^2 \ . \tag{2.2.4}$$

Now, since f is smooth, Proposition 2.1.4 gives

$$-2\gamma\langle \nabla f(w^t), w^{t+1} - w^t \rangle \leq -2\gamma(f(w^{t+1}) - f(w^t)) + M\gamma\|w^{t+1} - w^t\|^2 \ . \tag{2.2.5}$$

And by convexity of f and ψ, we have, from (2.1.11) and the definition of the subdifferential (Definition 2.1.6),

$$-2\gamma\langle \nabla f(w^t), w^t - w^* \rangle \leq -2\gamma(f(w^t) - f(w^*)) \ , \tag{2.2.6}$$

$$-2\gamma\langle \xi^t, w^{t+1} - w^* \rangle \leq -2\gamma(\psi(w^{t+1}) - \psi(w^*)) \ . \tag{2.2.7}$$

Summing (2.2.5), (2.2.6) and (2.2.7), then replacing in (2.2.3), we obtain

$$\begin{aligned} \|w^{t+1} - w^*\|^2 &\leq \|w^t - w^*\|^2 - 2\gamma(F(w^{t+1}) - F(w^*)) + (M\gamma - 1)\|w^{t+1} - w^t\|^2 \\ &\leq \|w^t - w^*\|^2 - 2\gamma(F(w^{t+1}) - F(w^*)) \ , \end{aligned} \tag{2.2.8}$$

where the second inequality comes from $\gamma \leq 1/M$. We now distinguish two cases:

- F is convex, then we sum this inequality for $t = 0$ to $t = T - 1$ and sum the telescoping sum to obtain

$$2\gamma \sum_{t=1}^{T} F(w^t) - F(w^*) \leq \|w^0 - w^*\|^2 \ . \tag{2.2.9}$$

  Then, remark that $F(w^t)$ is a decreasing function of t (see *e.g.,* the proof of Theorem 3.1 in Beck and Teboulle (2009)), therefore $F(w^t) \leq F(w^T)$ for all $t \leq T$, and the result follows.

- F is μ-strongly convex *w.r.t.,* $\|\cdot\|_2$, then by (2.1.14), we have $-2\gamma(F(w^{t+1}) - F(w^*)) \leq -\gamma\mu\|w^{t+1} - w^*\|^2$. This gives the inequality

$$(1 + \gamma\mu)\|w^{t+1} - w^*\|^2 \leq \|w^t - w^*\|^2 \ . \tag{2.2.10}$$

  The result follows from $\frac{1}{1+\gamma\mu} \leq 1 - \frac{\gamma\mu}{2}$, which holds since $\gamma\mu \leq 1$. □

Theorem 2.2.1 suggests that, both for convex and strongly-convex functions, setting the step size to $1/M$ is the best strategy. When the objective function is μ-strongly-convex, the convergence speed of `PGD` is governed by the ratio

$$\kappa = \frac{M}{\mu} \ , \tag{2.2.11}$$

which is called the *condition number* of the problem: `PGD` converges fast on problems with small condition number, and slow on problems with large condition number.

2.2.2 Proximal Stochastic Gradient Descent

In many applications, computing the gradient of f is expensive. This is notably the case in machine learning applications, where f depends on a large dataset. In such cases, it may be sufficient to compute a stochastic estimate of the gradient, for instance by using only the gradient of f_i for some $i \in [n]$ instead of the gradient of f. This is the idea of *stochastic gradient descent* (`SGD`), as introduced by Robbins and Monro (1951). On smooth functions, the non-asymptotic convergence of `SGD` was studied by Bach and Moulines (2011). They notably discuss rules for choosing adapting step sizes over the iteration of `SGD` to guarantee the convergence of its iterates. Their analysis was refined by Needell et al. (2016) and Gower et al. (2019), improving convergence rate and relaxing assumptions on the objective function. When the objective function is composite (*i.e.,* $\psi \neq 0$ in $(\star)$), we can consider a proximal variant of `SGD`, that we describe in Algorithm 2.2.1.

Algorithm 2.2.2: Proximal SGD: Proximal Stochastic Gradient Descent.

Input: initial point w^0, step size γ.

For $t = 0$ to $T - 1$:

Sample one index i uniformly randomly in $[n]$

Update $w^{t+1} = \text{prox}_{\gamma\psi}\left(w^t - \gamma\nabla f_i(w^t)\right)$

Return: w^T.

The Proximal SGD algorithm and its variants have notably been studied by Nitanda (2014), Atchadé et al. (2017), Rosasco et al. (2020), Cevher and Vũ (2019), and Gorbunov et al. (2020), The convergence of Proximal SGD with constant step size can be described in two different phases: (i) a convergence phase, where the iterates get closer to a solution, and (ii) an oscillation phase, where iterates oscillate around a solution due to the variance in the estimation of the gradient. This is what we describe in the next theorem.

Theorem 2.2.2 (Convergence of Proximal SGD)**.** *Assume f is M-smooth, and let w^* be a minimizer of F. Let w^t be the iterates of Algorithm 2.2.2 with step size $\gamma \leq 1/8M$. We denote $\sigma_*^2 = \mathbb{E}_s(\|g_s(w^*)\|^2)$ the variance of the gradient estimate at the optimum. Then, for general convex objectives (see Corollary 11.6 in Garrigos and Gower, 2023, based the general results of Khaled et al., 2020)*

$$\mathbb{E}(F(\bar{w}^t) - F(w^*)) \leq \frac{\|w^0 - w^*\|^2 + 2\gamma(F(w^0) - F(w^*))}{\gamma t} + 4\gamma\sigma_*^2 \ , \tag{2.2.12}$$

where $\bar{w}^t = \sum_{k=1}^t w^k$. If f is μ-strongly convex, then (see Corollary A.1 in Gorbunov et al., 2020)

$$\mathbb{E}(\|w^t - w^*\|^2) \leq (1 - \gamma\mu)^t\|w^0 - w^*\|^2 + \frac{2\gamma\sigma_*^2}{\mu} \ . \tag{2.2.13}$$

In both results of Theorem 2.2.2, a variance term remains. This is due to the oscillation phase, where the noise in the estimation of the gradient dominates, and the iterates remain in a ball around the optimum. The radius of this ball is determined by the variance at the optimum σ_*^2, and the step size. Therefore, setting smaller step sizes allows finding better solutions, but slows down the convergence of Proximal SGD.

We illustrate this phenomenon in Figure 2.2.1: Proximal SGD eventually reaches a plateau, where it does not progress towards the optimum anymore. The distance

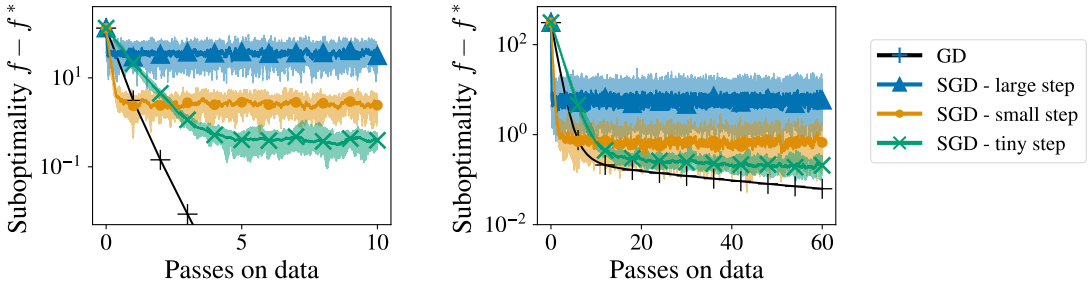


Figure 2.2.1: Evolution of the suboptimality gap for `PGD` and Proximal `SGD` with different step sizes. When step sizes are large, Proximal `SGD` converges faster than `GD` in its first iterations, but quickly reaches a plateau. With smaller step sizes, it converges slower, but is able to find better solutions. For both algorithms, the condition number $\kappa = M/\mu$ (defined in (2.2.11)) determines the convergence rate.

from this optimum is determined by the step size, and as in `PGD`, the convergence speed is determined by the condition number $\kappa = M/\mu$.

We note that this oscillating phenomenon can be compensated for by using variance reduction schemes. Many schemes have been proposed over the ten past years (see for instance Johnson and Zhang, 2013; Defazio et al., 2014; Xiao and Zhang, 2014, and many many others). We refer to Gower et al. (2020), Gorbunov et al. (2020), and Khaled et al. (2020) for overviews and unified analyses of such methods.

In the following of this thesis, we will study differentially private variants of these algorithms (see Section 3.2). In this setting, the variance in the estimation of f's gradient is due to the privacy constraints, and variance reduction does not allow us to get rid of this additive term. We, therefore, do not discuss them further.

2.2.3 Proximal Coordinate Descent

In some problems, it may be interesting to update iterates only *one coordinate at a time*. This is the idea of *coordinate descent*. It has two important advantages:

- In high-dimensional problems, computing one coordinate of the gradient is much cheaper than computing the full gradient, which can make the method very fast.
- Updating coordinates one at a time can allow the use of larger step sizes.

Coordinate descent methods have encountered large success due to their simplicity and effectiveness (Liu et al., 2009; Friedman et al., 2010; Chang et al., 2008; Sardy et al., 2000), and have seen a surge of practical and theoretical interest in the last decade (Wright, 2015; Shi et al., 2017; Richtárik and Takáč, 2014; Fercoq and Richtárik,

2014; Tappenden et al., 2016; Hanzely et al., 2020; Nutini et al., 2015; Karimireddy et al., 2019). This theoretical study started with the works of Luo and Tseng (1992), Tseng (2001), and Tseng and Yun (2009), who studied coordinate descent for non-smooth optimization problems. Then, Nesterov (2010) analyzed coordinate descent with random selection of the updated coordinate for smooth problems. They derived convergence results in expectation, showing that coordinate descent algorithms can be extremely efficient on large scale problems. In general, we refer to Wright (2015) and Shi et al. (2017) for a general overview of results on coordinate descent methods.

To design proximal variants of coordinate descent, we need to assume that the non-smooth part ψ of $(\star)$ is *separable*:

$$\psi(w) = \sum_{j=1}^{p} \psi_j(w_j) \ . \tag{2.2.14}$$

This assumption means that each the function can be split in an ensemble of functions, that each depend on only one of the coordinates. This is notably the case of the ℓ_1-norm and of the characteristic function of a box-set.

The separability assumption allows to do coordinate-wise proximal updates. This gives the following proximal stochastic coordinate descent algorithm.

Algorithm 2.2.3: `PCD`: Proximal Coordinate Descent.

Input: initial point w^0, step sizes $\gamma_1, \dots, \gamma_p$.

For $t = 0$ to $T - 1$:

Sample index j uniformly randomly in $[p]$

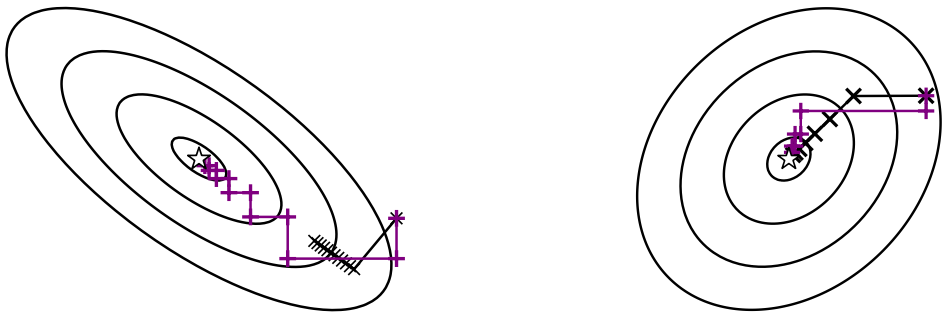
Set $w^{t+1} = w^t$

Update $w_j^{t+1} = \mathrm{prox}_{\gamma_j \psi_j}\left(w_j^t - \gamma_j \nabla_j f(w^t)\right)$

Return: w^T.

The theoretical convergence properties of this algorithm were notably studied by Richtárik and Takáč (2014), Fercoq and Richtárik (2014), and Karimi et al. (2016). We study the convergence rate of `PCD` in the following theorem.

Theorem 2.2.3 (Convergence of `PCD`)**.** *Assume f is M-coordinate-smooth, where $M = \mathrm{diag}(M_1, \dots, M_p) \in \mathbb{R}^{p \times p}$ for $M_1, \dots, M_p > 0$, and let w^* be a minimizer of F.*



(a) High condition number $\kappa \approx 37.6$. (b) Low condition number $\kappa \approx 2.7$.

Figure 2.2.2: Trajectory of coordinate descent (purple `+`) and gradient descent (black `x`) for two quadratic problems $f = w^\top A w$ with different condition numbers $\kappa = M/\mu$ (as defined in (2.2.11)), where M and μ are the largest and smallest eigenvalues of the Hessian of f. `CD` is much less sensitive to bad conditioning than `GD`: on both problems, it finds good solutions fast, while `GD` stalls after a few iterations.

Let w^t be the iterates of Algorithm 2.2.3 with step size $\gamma_j = 1/M_j$ for $j \in [p]$. Then for convex objectives (see Richtárik and Takáč (2014), Theorem 5)

$$F(w^t) - F(w^*) \leq \frac{2p \max\left(F(w^0) - F(w^*), \|w^0 - w^*\|_M^2\right)}{t} \ . \tag{2.2.15}$$

If additionally F is μ_M-strongly convex w.r.t., $\|\cdot\|_M$*, then (see Richtárik and Takáč (2014), Theorem 7)*

$$F(w^t) - F(w^*) \leq \left(1 - \frac{\mu_M}{p}\right)^t (F(w^0) - F(w^*)) \ . \tag{2.2.16}$$

Theorem 2.2.3 shows that the convergence rate of `PCD` is determined by $\|w^0 - w^*\|_M^2$ and μ_M, for convex and strongly-convex objectives respectively. These values scale with the *coordinate-wise smoothness* of the objective function. As such, if f is β-smooth and M-coordinate-smooth, $\|w^0 - w^*\|_M^2$ can be much smaller than $\beta\|w^0 - w^*\|_2^2$, and μ_M can be much larger than μ/β. Therefore, `PCD` is less sensitive to poor conditioning of the problem at hand than `PGD` and Proximal `SGD`. This is due to its ability to make much larger step sizes on coordinates with small smoothness constants. We illustrate this phenomenon in Figure 2.2.2. We will show in Chapter 4 that this property can be used to improve the privacy-utility trade-off in differentially private optimization.

2.2.4 Greedy Coordinate Descent

In some problems, it may be interesting not to choose the updated coordinate uniformly randomly. One possibility is to choose it as the one with the largest gradient entry. This strategy is sometimes named the *Gauss-Southwell rule*, and the corresponding algorithm is called greedy coordinate descent. This algorithm was notably discussed by Luo and Tseng (1992), Tseng and Yun (2009), and Dhillon et al. (2011). It can also be seen as a special case of the steepest descent method (see Section 9.4.3 in Boyd and Vandenberghe, 2004). We describe this algorithm for optimizing the smooth variant of ($\star$) (with $\psi = 0$) in Algorithm 2.2.4.

The first theoretical analyses of greedy coordinate descent's convergence did not show improvement over the stochastic greedy coordinate descent we described in the previous section. These results therefore suggest that it is no use to select the coordinate greedily rather than randomly. But greedy updates do help in many cases. This is what Dhillon et al. (2011) and Nutini et al. (2015) showed by proposing refined convergence results for convex and strongly-convex objectives, that we state in the following theorem.

Algorithm 2.2.4: `GCD`: Greedy Coordinate Descent.

Input: initial point w^0, step sizes $\gamma_1, \dots, \gamma_p > 0$.

For $t = 0$ to $T - 1$:

Compute $j = \arg\max_{j \in [p]} \left\{ \frac{1}{M_j} |\nabla_j f(w^t)|^2 \right\}$

Set $w^{t+1} = w^t$

Update $w_j^{t+1} = w_j^t - \gamma_j \nabla_j f(w^t)$

Return: w^T.

Theorem 2.2.4 (Convergence of `GCD`)**.** *Assume f is M-coordinate-smooth with $M = \mathrm{diag}(M_1, \dots, M_P) \in \mathbb{R}^{p \times p}$ for some $M_1, \dots, M_p > 0$, and let w^* be a minimizer of F. Let w^t be the iterates of Algorithm 2.2.4 with step size $\gamma_j = 1/M_j$. Then, let $R_{M,1} = \max_{w \in \mathbb{R}^p} \min_{w^* \in \mathcal{W}^*} \left\{ \|w - w^*\|_{M,1} \mid f(w) \leq f(w^0) \right\}$. For general convex objectives (see Lemma 1 in Dhillon et al., 2011, or Theorem 3 in Karimireddy et al., 2019)*

$$f(w^t) - f(w^0) \leq \frac{R_{M,1}^2}{2t} \quad . \tag{2.2.17}$$

If, additionally, F is $\mu_{M,1}$-strongly convex w.r.t., *the norm $\|\cdot\|_{M,1}$, then (see Nutini et al., 2015, Section 4)*

$$f(w^t) - f(w^*) \leq \left(1 - \mu_{M,1}\right)^t (F(w^0) - F(w^*)) \ . \tag{2.2.18}$$

In these results, $R_{M,1}$ and $\mu_{M,1}$ are defined using the (scaled) ℓ_1-norm. This allows to get rid of the explicit dependence on the dimension that appears in the analysis of stochastic coordinate descent (see Theorem 2.2.3). Importantly, since for any vector $w \in \mathbb{R}^p$, $\|w\|_2 \leq \|w\|_1 \leq \sqrt{p}\|w\|_2$, these result imply that greedy coordinate descent is always better than stochastic coordinate descent. Most interestingly, in the best case, *greedy coordinate descent enjoys the same rate as gradient descent.*[2] This will be at the core of Chapter 5, where we propose a differentially private greedy coordinate descent method and formally analyze its privacy-utility trade-off.

Sometimes, it may still be interesting, theoretically, to use greedy coordinate descent rather than gradient descent. Indeed, in some specific settings, it is possible to approximate the greedy update rule in sublinear time. This can notably be done using fast nearest-neighbor schemes when fitting (generalized) linear models (Dhillon et al., 2011; Nutini et al., 2015; Karimireddy et al., 2019). In practice, however, greedy coordinate descent methods are often slower (in wall-clock time) than their randomized or cyclic counterparts (Massias et al., 2017). We will see in Chapter 5 that the private variant of this algorithm can obtain near-dimension independent utility, which may be worth the high computational cost.

Note that the analysis of proximal extensions of greedy coordinate descent for composite problems is challenging. Karimireddy et al. (2019) proved convergence rates for ℓ_1-regularized and box-constrained problems, using a modified greedy coordinate algorithm. Nonetheless, we remark that proximal variants of greedy coordinate (see *e.g.,* Section 2.3.3 in Shi et al., 2017) seem to work well in practice, even without such tricks.

[2]We refer to the supplementary of Nutini et al. (2015) for examples of problems where gradient descent and greedy coordinate descent perform similarly.

Chapter 3

Background on Differential Privacy in Machine Learning

Privacy is now commonly recognized as a human right. The term "privacy" was first used by Warren and Brandeis (1890), who described it as a right that should be protected by law. Their point of view on privacy stems from the following observation:

> *" Instantaneous photographs and newspaper enterprise have invaded the sacred precincts of private and domestic life; and numerous mechanical devices threaten to make good the prediction that 'what is whispered in the closet shall be proclaimed from the house-top'. "*
>
> — *Warren and Brandeis (1890).*

At that time, the concern was that, due to new technology (like photographs and newspapers), parts of people's private life could be publicly exposed. This led Warren and Brandeis (1890) to the definition of a right to privacy. Simply put, they describe it as "the right to be let alone". Nowadays, technology has evolved far beyond the printed press, and privacy concerns are more important than ever. Indeed, as you are reading these lines, massive data is being collected, everywhere. This data holds information about virtually everyone. And this information is personal, thus sensitive.

Describing and protecting the personal nature of this data is nowadays what *privacy* is all about. Modern definitions of privacy have changed accordingly: they are now centered around *personal information* and *surveillance*. One such definition is:

> *Freedom from damaging publicity, public scrutiny, surveillance, and disclosure of personal information, usually by a government or a private organization.*
>
> — *from the Wiktionary*[1].

[1] Available online: https://en.wiktionary.org/wiki/privacy.

This definition is aligned with a general societal reflection on the role of personal data and the importance of privacy. Such ideas have recently made their way into the law through the GDPR (2016) in Europe, and similar privacy laws in some countries. These laws provide a general framework for the protection of individuals' right to privacy. But this framework is not usable as-is: we need a more rigorous notion of privacy, that can be used as a foundation to privacy-preserving algorithms.

In this chapter, we introduce *differential privacy* (Dwork, 2006) and *differentially private machine learning* (Chaudhuri et al., 2011). We start in Section 3.1 with an overview of a few notions of privacy that preceded differential privacy, and discuss their limitations. Then, we formally define differential privacy and describe a few basic mechanisms, that can be used to enforce it. We also show how to combine these mechanisms to build complex differentially private algorithms. In Section 3.2, we discuss the problem of training machine learning models in a differentially private way. We describe two algorithms that can be used to solve this problem: output perturbation and differentially private stochastic gradient descent. We prove that they satisfy differential privacy, and discuss their utility.

3.1 Differential Privacy

To formalize what privacy means, we study what happens when we release the result of a computation done on a database. In the remainder of this thesis, we refer to such computations as *queries*[2]. Differential privacy is a way of measuring how much information the output of a query leaks about an individual. It is built on the idea that if the presence (or the absence) of a specific individual in the database does not have "too much" influence on the output of the query, then it should be difficult for an external observer to guess anything about them.

Before introducing differential privacy formally, we discuss in Section 3.1.1 two other attempts at mathematically defining privacy: k-anonymity and perfect secrecy. We discuss their limitations, that laid the foundations of differential privacy (and some of its variants). We then define differential privacy in Section 3.1.2, and discuss its important properties. In particular, we explain why differential privacy is considered a robust notion of privacy. Sections 3.1.3 and 3.1.4 then describe how to build useful differentially private algorithms: we start by describing an ensemble of basic mechanisms, then explain how they can be combined into more complex algorithms.

[2]In general, in the differential privacy literature, the term "query" is broader than its usual database sense. It refers to any function that takes data as input. This can, for instance, be an algorithm that uses data to train a machine learning model.

3.1.1 Towards a Mathematical Definition of Privacy

Several approaches have been proposed to define privacy. The most usual idea is *data anonymization*, that intends to remove personally identifiable information from the data. Ideally, this allows to release entire datasets without compromising individuals' privacy. However, proper anonymization is hard to obtain in practice. Indeed, individuals can often be re-identified even after their personal information has been redacted. Fortunately, there are many situations where releasing the full database is not useful, since we are only interested in some aggregated values (*e.g.,* general statistics). This observation has motivated the study of *information theoretic* notions of privacy, that characterize how much information may leak from releasing the value of a function computed using data.

In Section 3.1.1 (a), we discuss why pseudonymization (*i.e.,* replacing personal identifiers by pseudonyms) is generally not sufficient to preserve the privacy of individuals. We then describe k-anonymity (and some of its variants) in Section 3.1.1 (b), which tries to define a rigorous framework to address the limitations of pseudonymization. This approach also suffers from important caveats: essentially, it is difficult to obtain a meaningful privacy without destroying all the relevant information from the database. This encourages to study the problem from another point of view, based on information theory, that we describe in Section 3.1.1 (c).

3.1.1 (a) Heuristic: Data Pseudonymization

Pseudonymization is the act of replacing identifying information (*e.g.,* names, social security number) by randomly generated pseudonyms. We illustrate this procedure on an example database in Figure 3.1.2.

Name	Birth date	ZIP	Diagnosis
Jacques	09/1929	48202	Healthy
Madeleine	05/1937	48137	Cancer
Mathilde	11/1982	21090	Diabetes
Guillaume	08/2000	21202	Diabetes

$\Longrightarrow$

Name	Birth date	ZIP	Diagnosis
uXzg	09/1929	48202	Healthy
Hayd	05/1937	48137	Cancer
xZs5	11/1982	21090	Diabetes
9zsW	08/2000	21202	Diabetes

Figure 3.1.1: Pseudonymization: names have been replaced by pseudonyms.

In the collective imagination, this is generally seen as a reasonable anonymization strategy. It does indeed prevent the honest observer from inferring information about individuals whose names have been redacted. But a less-honest person could exploit the fact that most people can be re-identified from combining multiple features. Indeed, the specific combination of a set of features' values often allows to re-identify people: we call such a set of features *quasi-identifiers*. A glaring example of this is

that, as demonstrated by Sweeney (2000), 87% of Americans are uniquely identified from just birth date, gender and ZIP code. Similarly, about 90% of web browsers have a unique fingerprint[3] (Eckersley, 2010; Laperdrix et al., 2016).

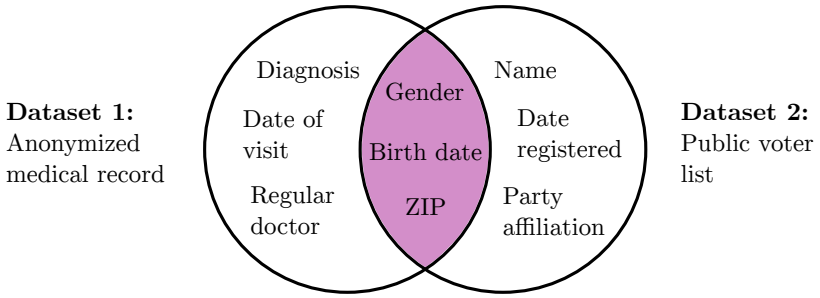


Figure 3.1.2: Linkage Attacks: anonymized dataset (without names) can be joined with a public dataset (with names) based on a few attributes. Sweeney (2000) estimated that age, zip code and birth date suffice to uniquely identify 87% of Americans.

This possibility of uniquely identifying individuals in the database makes the data sensitive to *linkage attacks*. These attacks exploit the fact that individuals are often part of multiple databases. This gives some side knowledge (*i.e.,* knowledge that is not part of the database itself), that can be used to reconstruct pseudonymized information. This is done by exploiting the uniqueness of individuals to join an anonymized dataset with a public dataset using a subset of shared attributes as keys. When specific combinations of these attributes are unique, the join operation succeeds. This enables the reconstruction of private information that was stripped from the original dataset by the anonymization procedure. We illustrate this operation in Figure 3.1.2, where the set of variables {age, ZIP and birth date} are used as keys. Using this method, Hawes (2021) was able to re-identify between 50 and 180 million Americans from linking the 2010 US Census data with commercially available data. Similarly, Narayanan and Shmatikov (2007) claimed that most of the users in the Netflix Prize Dataset could be re-identified by linkage with IMDb's data.

3.1.1 (b) Hiding among the Others: k-anonymity

To prevent linkage attacks, Sweeney (2002) proposed k-anonymity. Given a set of quasi-identifiers (*i.e.,* features that may allow re-identification), a dataset satisfies k-anonymity if every record shares the same combination of quasi-identifiers with at least $k-1$ others. This effectively prevents linkage attacks: as the quasi-identifiers have the same value for k different individuals, one cannot re-identify a specific person uniquely based on their values.

[3]To know whether you are unique, visit `https://www.amiunique.org/`. I, unfortunately, am.

To transform a database to fit k-anonymity (w.r.t. a set of quasi-identifiers), one generally removes some attributes, and categorize others into a set of bins. These bins are defined so that multiple users fall in each bin. With properly chosen bins, it is possible to satisfy k-anonymity. We give an example of this procedure in Figure 3.1.3.

Name	Birth date	ZIP	Diagnosis
Jacques	09/1929	13741	Healthy
Madeleine	05/1937	13440	Cancer
Mathilde	11/1982	21090	Diabetes
Guillaume	08/2000	21202	Diabetes

$\Longrightarrow$

Name	Birth date	ZIP	Diagnosis
xxx	Before 1949	48xxx	Healthy
xxx	Before 1949	48xxx	Cancer
xxx	After 1950	21xxx	Diabetes
xxx	After 1950	21xxx	Diabetes

Figure 3.1.3: Preventing linkage attacks: names are removed, birth dates are binarized, and ZIP codes are reduced to the first two digits. The resulting dataset satisfies 2-anonymity for the set of attributes (Names, Birth date, ZIP).

Although k-anonymity prevents linkage attacks that use the specified quasi-identifiers, it still suffers from several serious drawbacks. First, choosing the set of quasi-identifiers is difficult. If this set is too narrow, it may not prevent linkage attacks against adversaries that have background knowledge on the other attributes. Conversely, if it is too large, too much information may have been removed. For instance, in Figure 3.1.3's data, guaranteeing 4-anonymity would force *all records to be the same*, which destroys utility. This phenomenon is especially common in high-dimensional sparse datasets (*e.g.,* the Netflix database), where very few records share the same value for a given feature. A second caveat of k-anonymity is *homogeneity attacks*: if all k records have the same value for a sensitive attribute, this still allows the attacker to infer it. For instance, the last two records in Figure 3.1.3's anonymized dataset share the same name, birth date and ZIP code: it is thus impossible to determine whether the modified record is about Guillaume or Mathilde. But all individuals from the group have diabetes, which is sufficient for the attacker to complete its attack successfully, which constitutes a serious privacy leak.

Remark 3.1.1. *Multiple refinements of k-anonymity have been proposed. Notably, ℓ-diversity (Machanavajjhala et al., 2007) prevents homogeneity attacks by promoting diversity in groups that share the same quasi-identifiers; and t-closeness (Li et al., 2007) further imposes that some attributes have similar distribution in a specific group and in the complete dataset. These can improve privacy guarantees, although at the cost of destroying even more information, which sometimes makes the resulting dataset useless.*

3.1.1 (c) Information Theoretic Privacy: Perfect Secrecy

The methods we presented in Section 3.1.1 (b) all suffer from the same problem: either the privacy guarantee is rather weak, or the useful information is completely removed from the data. This is due to the fact that they aim at guaranteeing the privacy of the *complete* database. Generally, we are more interested in the output of a query[4] than on the data itself. Queries typically reveal only part of the information present in the data, which allows to give more precise answers while preserving privacy.

Ideally, a query that preserves privacy should *reveal no information* on the data it used in its computations. Such queries satisfy *perfect secrecy*, which was introduced by Shannon (1949). Perfect secrecy ensures that all outputs of a query (on a database) arise with the same probability, no matter what this database is. Formally, let $\mathcal{D}$ be a set of datasets and $\mathcal{E}$ an arbitrary set. A (randomized) query $\mathcal{A} : \mathcal{D} \to \mathcal{E}$ satisfies perfect secrecy if

$$\mathbb{P}(\mathcal{A}(D) \subseteq S) = \mathbb{P}(\mathcal{A}(D') \subseteq S) \ , \quad \text{for all } S \subseteq \mathcal{E}, \text{ and all } D, D' \in \mathcal{D} \ . \tag{3.1.1}$$

Intuitively, this means that anyone who observes the output of $\mathcal{A}$ does not gain any knowledge on its input (as all input are as likely to have produced this result). This gives a very strong notion of privacy, and is typically achieved by cryptographic protocols. Examples of schemes that achieve this goal (in various settings) are one time pads (Miller, 1882), Shamir's secret sharing (Shamir, 1979) or symmetric (Pub, FIPS, 1999) and asymmetric key algorithms (Diffie and Hellman, 2022). In these protocols, an authorized party (who does not aim at publishing the message) knows the randomness of $\mathcal{A}$, and can thus inverse it to find the initial message.

As is, perfect secrecy provides an extremely strong notion of privacy, but is not suitable for controlling how much information may leak upon publication of the result. Indeed, if nothing is learned from the data, it does not seem very reasonable to use the data in the first place. Therefore, we need to relax the constraint (3.1.1) imposed by perfect secrecy: this is the goal of differential privacy.

3.1.2 Definition of Differential Privacy

We now introduce differential privacy, as proposed by Dwork (2006). Similar to perfect secrecy (see Section 3.1.1 (c)), differential privacy is a property of a *query* that takes a database as input. Informally, we can describe differential privacy as follows.

> *A query is differentially private if observing its output does not allow to tell too confidently whether an individual was part of the database or not.*

[4] Recall here that the term query refers to any functions computed on a dataset.

As we will see later, differential privacy has multiple strengths. First, it allows us to give a precise meaning to the words "too confidently". As such, it quantifies the information leakage that is induced by the release of a query's output. Second, the differential privacy guarantee holds regardless of the knowledge of the adversary.

In the remainder of this section, we formally introduce the notion of differential privacy. To this end, we start by defining *divergences*, that appear as natural tools to relax the condition (3.1.1) from perfect secrecy. We then use these divergences to define several flavors of differential privacy.

3.1.2 (a) Divergences

To define differential privacy, we need to compare probability distributions. This will allow to relax the definition of perfect secrecy from (3.1.1), that required two probability distributions to be perfectly equal. A very powerful tool for this kind of comparison is the family of Rényi divergences (van Erven and Harremos, 2014).

Definition 3.1.1 (Rényi Divergences, Equation (9) in van Erven and Harremos, 2014)**.** *Let $\alpha > 1$ and P, Q be two probability distributions over the same set $\mathcal{X}$. The α-Rényi divergence between P and Q is defined as*

$$\mathrm{Div}_\alpha(P\|Q) = \frac{1}{\alpha - 1} \log \int_{\mathcal{X}} \frac{P(x)^\alpha}{Q(x)^\alpha} dQ(x) \ , \tag{3.1.2}$$

with the conventions that ${}^0\!/_0 = 0$ and ${}^x\!/_0 = +\infty$ for $x > 0$.

Rényi divergences measure the dissimilarity of two probability distributions. They are based on the moments of the variable $\frac{P(X)}{Q(X)}$, where X follows the distribution Q. They interpolate between the two following divergences:

- The Kullback-Leibler divergence ($\alpha \to 1$): $\mathrm{Div}_{KL}(P\|Q) = \int_{\mathcal{X}} \log \frac{P(x)}{Q(x)} dP(x)$.
- The max divergence ($\alpha \to +\infty$): $\mathrm{Div}_\infty(P\|Q) = \sup_{S \subseteq \mathcal{X}} \log \frac{P(S)}{Q(S)}$.

In the first case ($\alpha \to 1$), the Rényi divergence only controls the moment of order 1 (*i.e.,* the mean) of $P(X)/Q(X)$. In the second one ($\alpha \to \infty$), it controls its moment of order ∞ (*i.e.,* its maximal value). In short, the parameter α controls the importance of the tail of $\frac{P(X)}{Q(X)}$ in the value of the divergence Div_α. This property of Rényi divergences can be linked to the following divergence, which we call the "hockey-stick" divergence[5] (Dwork and Roth, 2014; Sason and Verdú, 2016), for $\delta \in [0, 1]$:

$$\mathrm{Div}_\infty^\delta(P\|Q) = \sup_{S \subseteq \mathcal{X} \ \mid \ P(S) \geq \delta} \log \frac{P(S) - \delta}{Q(S)} \ . \tag{3.1.3}$$

[5]The name comes from the hockey stick shape of the ReLU function $x \mapsto \max(0, x)$.

Intuitively, this divergence "ignores" the sets S that are too unlikely to arise in P. It can be shown, for $\alpha > 1$, that

$$\mathrm{Div}_\infty^\delta(P\|Q) \leq \mathrm{Div}_\alpha(P\|Q) + \frac{\log(1/\delta)}{\alpha - 1} \ . \tag{3.1.4}$$

This result follows from the probability preservation property stated in Lemma 4.1 of Langlois et al. (2014), using the derivations from Proposition 3 of Mironov (2017).

3.1.2 (b) Differential Privacy

We now introduce the notion of (approximate) differential privacy (Dwork, 2006) and its Rényi differential privacy variant (Mironov et al., 2019). Let $\mathcal{D}$ be a set of datasets. We say that $D, D' \in \mathcal{D}$ are neighboring (and denote $D \sim D'$) whenever they have the same size and differ on at most one element. In the following, $\mathcal{E}$ is a set, and $\mathcal{A} : \mathcal{D} \to \mathcal{E}$ is a randomized algorithm that takes a dataset as input and outputs some (random) value.

Definition 3.1.2 (Differential Privacy)**.** *Let $\mathcal{A} : \mathcal{D} \to \mathcal{E}$ be a randomized algorithm. For $\epsilon \geq 0$, the algorithm $\mathcal{A}$ is ϵ-*Div*-differentially private if for all datasets $D, D' \in \mathcal{D}$ that differ on at most one element,*

$$\mathrm{Div}(\mathcal{A}(D)\|\mathcal{A}(D')) \leq \epsilon \ . \tag{3.1.5}$$

Depending on the divergence, we obtain different flavors of differential privacy:

- $\mathrm{Div} = \mathrm{Div}_\infty$*: pure ϵ-differential privacy (Dwork, 2006):*

$$\mathbb{P}(\mathcal{A}(D) \subseteq S) \leq \exp(\epsilon)\, \mathbb{P}(\mathcal{A}(D') \subseteq S) \ , \quad \text{for all } S \subseteq \mathcal{E} \ .$$

- $\mathrm{Div} = \mathrm{Div}_\infty^\delta$ $(\delta \in [0, 1])$*: approximate (ϵ, δ)-differential privacy (Dwork, 2006):*

$$\mathbb{P}(\mathcal{A}(D) \subseteq S) \leq \exp(\epsilon)\, \mathbb{P}(\mathcal{A}(D') \subseteq S) + \delta \ , \quad \text{for all } S \subseteq \mathcal{E} \ .$$

- $\mathrm{Div} = \mathrm{Div}_\alpha$ $(\alpha > 1)$*: (α, ϵ)-Rényi differential privacy (Mironov et al., 2019).*

In general, the lower ϵ and δ are, and the higher α is, the stronger the privacy guarantees become. Typically, when $\epsilon \to 0$ and $\delta = 0$ or $\alpha = \infty$, Definition 3.1.2 boils down to perfect secrecy (see Section 3.1.1 (c)).

Differential privacy, and its approximate variant, mean that for two neighboring dataset $D \sim D' \in \mathcal{D}$, a given output of $\mathcal{A}$ run on D is not more than $\exp(\epsilon)$ more

likely[6] (up to some slack δ) to be observed than the same output if $\mathcal{A}$ was run on D'. This is also the case for Rényi differential privacy, that can be converted back to (approximate) differential privacy using the following proposition.

Proposition 3.1.1 (Proposition 3 in Mironov, 2017)**.** *Let $\mathcal{A} : \mathcal{D} \to \mathcal{E}$ be a (α, ϵ)-Rényi differentially private mechanism. Then, it follows from Equation (3.1.3) that $\mathcal{A}$ is $(\epsilon + \frac{\log(1/\delta)}{\alpha - 1}, \delta)$-differentially private.*

The main strength of differential privacy is that it is robust to prior knowledge of the adversary. Informally, an adversary learns little from the output of a query, regardless of their knowledge of the data. This is very different from the notions presented above. For instance, with k-anonymity, an adversary with external knowledge was still able to do homogeneity attacks (see discussions in Section 3.1.1 (b)), therefore learning more than an adversary without any knowledge of the data. This idea has been formalized by Kasiviswanathan and Smith (2014), who proposed a bayesian view on the maximal knowledge an adversary can gain from a differentially private query.

The second important ingredient to the robustness of differential privacy is its post-processing property. It states that the differential privacy guarantee of an algorithm cannot be diminished by further processing its output, as long as this processing is independent on the data and on the initial algorithm.

Proposition 3.1.2 (Post-Processing Dwork and Roth, 2014)**.** *Let $\mathcal{E}, \mathcal{F}$ be two sets, and $\mathcal{D}$ the set of possible datasets. Let $\epsilon > 0$ and* Div *be a divergence. If $\mathcal{A} : \mathcal{D} \to \mathcal{E}$ is ϵ-*Div*-differentially private, and $f : \mathcal{E} \to \mathcal{F}$ is a function that is independent from the data and the randomness of $\mathcal{A}$, then $f \circ \mathcal{A}$ is also ϵ-*Div*-differentially private.*

Proposition 3.1.2 is a consequence of the *data processing inequality* (see *e.g.,* Beaudry and Renner, 2012). Seeing $\mathcal{A}(D)$ as a noisy signal produced from a dataset $D \in \mathcal{D}$, this means that the signal due to D present in $\mathcal{A}$ cannot be amplified by post-processing. This property is fundamental to the robustness of differential privacy, as it ensures that once a value is released, it cannot get less private by any means.

3.1.3 Basic Building Blocks for Differential Privacy

In practice, it is common to compute deterministic queries on the data. Being deterministic, such queries do not satisfy differential privacy. Therefore, we need to incorporate some kind of randomness before releasing the query's output. This allows to transform the query into a differentially private one, but reduces the utility of the result. To choose the right amount of noise, one can use an ensemble of basic

[6]For small value, note that $\exp^{\epsilon} \approx 1 + \epsilon$. Small values of ϵ therefore really mean that the two probabilities are very close.

differentially private mechanisms. These mechanisms allow releasing the value of a query on a database under a fixed privacy budget. We will see in the next section that these mechanisms can then be combined to create more complex algorithms. We describe four of these basic mechanisms in this section.

3.1.3 (a) Randomized Response

The most basic (and historic!) differentially private mechanism is the randomized response mechanism. This mechanism was initially proposed by Warner (1965), 41 years before differential privacy (Dwork, 2006). In this mechanism, the query is a closed-ended question[7]. We describe it for binary questions, but variants of this mechanism have been proposed when it is not the case (Kairouz et al., 2014).

In randomized response, respondents flip a coin before answering, then:

(i) if it comes up heads, they answer truthfully,

(ii) if it comes up tails, they answer Yes or No uniformly randomly.

We describe the complete procedure in Algorithm 3.1.1.

Algorithm 3.1.1: $\texttt{RR}_p$: Randomized Response Mechanism.

Input: probability p of answering randomly.

Sample $\theta \sim \mathcal{B}(p)$ from a Bernoulli distribution.

If $\theta = 0$: set r to the true answer.

Else: set r to Yes or No uniformly randomly.

Return: randomized response r.

The proportion of "Yes" in the output of Algorithm 3.1.1 is $q_{\text{priv}} = pq_{\text{true}} + \frac{1}{2}(1-p)$, where q_{true} is the underlying probability of answering "Yes". We can therefore build an estimator of q_{true} from the output of the $\texttt{RR}_p$ mechanism as follows

$$q_{\text{estimated}} = \frac{1}{2p}(2r - 1 + p) \ ,$$

where r is the output of the $\texttt{RR}_p$ mechanism. Additionally, the $\texttt{RR}_p$ satisfies differential privacy. Depending on the probability that the coin comes up heads (or tails),

[7] In the historical context of the cold war, an example of such a question would be "Are you a member of the Communist party?".

different levels of privacy can be obtained. We state the privacy guarantees of Algorithm 3.1.1 in the following theorem (see *e.g.,* Dwork and Roth, 2014; Erlingsson et al., 2014).

Theorem 3.1.1 (Differential privacy guarantees for $\texttt{RR}_p$)**.** *Let $\epsilon > 0$. Then, the $\texttt{RR}_p$ mechanism with $p = \frac{2}{1+\exp(\epsilon)}$ is ϵ-differentially private.*

Proof. Let R, T be two random variables, respectively representing the response and the truth of a respondent. We can compute

$$\begin{aligned}\mathbb{P}(R = \text{Yes} \mid T = \text{Yes}) &= 1 - p + \tfrac{1}{2}p = 1 - \tfrac{1}{2}p \ , \\ \mathbb{P}(R = \text{Yes} \mid T = \text{No}) &= \tfrac{1}{2}p \ .\end{aligned}$$

Consequently, $\frac{\mathbb{P}(R=\text{Yes}|T=\text{Yes})}{\mathbb{P}(R=\text{Yes}|T=\text{No})} = \frac{1-p/2}{p/2}$. Setting $p = \frac{2}{1+\exp(\epsilon)}$ gives the result. □

The proof of this theorem is very simple, but highlights one key asset of these mechanisms: they can guarantee privacy *without assumptions on the specific distribution of the data*. There, no matter the true probability of answering "Yes", the mechanism will still guarantee differential privacy.

3.1.3 (b) Sensitivity of a Query

When queries are not closed-ended questions, we will need to assess how much a query can change between two datasets. The maximal change in the value of a query is called the *sensitivity* of the query.

Definition 3.1.3 (Sensitivity of a Query)**.** *Let $f : \mathcal{D} \to \mathcal{E}$ be a query and $\|\cdot\|$ be an arbitrary norm on $\mathcal{E}$. We define the sensitivity of f associated to the norm $\|\cdot\|$ as*

$$\Delta_{\|\cdot\|}(f) = \max_{D \sim D' \in \mathcal{D}} \|f(D) - f(D')\| \ , \tag{3.1.6}$$

where the maximum is computed over neighboring datasets. When $\mathcal{E} = \mathbb{R}^p$, we denote $\Delta_1(f) = \Delta_{\|\cdot\|_1}(f)$ and $\Delta_2(f) = \Delta_{\|\cdot\|_2}(f)$ the ℓ_1 and ℓ_2 sensitivities of f.

This sensitivity plays in a key role in the design of mechanisms that release a differentially private estimate of the query f. It will be used to calibrate the noise to release this value under a fixed privacy budget. We discuss that in Section 3.1.3 (c).

A fundamental query is the averaging query. These types of queries will notably arise in the design of optimization algorithms (see Section 3.2.3 (b)). Indeed, in these algorithms, the gradient of the empirical risk is the average of the gradient of the loss across the dataset. This query is also central in the derivation of utility lower bounds in Section 3.2.4. We now define this query and give its sensitivity.

Example 3.1.1 (Average)**.** *Let $\mathcal{D} \subseteq \{0,1\}^n$, and for $D = \{d_1, \dots, d_n\} \in \mathcal{D}$, the averaging query be $f(D) = \frac{1}{n}\sum_{i=1}^n d_i$. The sensitivity of f is*

$$\Delta(f) = \sup_{D \sim D' \in \mathcal{D}} \left| \frac{1}{n}\sum_{i=1}^n d_i - \frac{1}{n}\sum_{i=1}^n d'_i \right| = \frac{1}{n}\left|d_{i^*} - d'_{i^*}\right| \leq \frac{1}{n} \ , \tag{3.1.7}$$

where the supremum is over neighboring datasets. In the second equality, we denoted i^* the index on which D, D' differ.

Remark that the sensitivity decreases as the number of records in the data increases. This is due to the fact that, when datasets are big, each individual has a smaller contribution to the result. This observation is essential when using differential privacy in practice, as it allows to compute aggregated values both *privately* and *accurately* when the dataset is large enough.

3.1.3 (c) Laplace and Gaussian Mechanism

When a query $f : \mathcal{D} \to \mathbb{R}^p$ has a bounded sensitivity, it is possible to release its (approximate) value in a differentially private way. The two most commonly used mechanisms to do so are the Laplace and Gaussian mechanisms. These mechanisms rely on the addition of Laplace or Gaussian noise, which will allow to conceal individuals' contribution to the value of the query.

We first describe the Laplace mechanism. This mechanism is based on the centered Laplace distribution, which has a probability density function given by

$$\mathrm{Lap}(x|\lambda) = \frac{1}{2\lambda}\exp\Big(-\frac{|x|}{\lambda}\Big) \ , \quad \text{for } x \in \mathbb{R} \ . \tag{3.1.8}$$

We will use the notation $X \sim \mathrm{Lap}(\lambda)^p$ to denote a random variable X taking its values in $\mathbb{R}^p$ whose coordinates are independently sampled from the Laplace distribution with parameter λ. The Laplace mechanism is defined as follows.

Algorithm 3.1.2: $\texttt{LM}_\lambda$: Laplace Mechanism.

Input: query $f : \mathcal{D} \to \mathbb{R}^p$, dataset D, noise scale $\lambda > 0$.

Return: $f(D) + \mathrm{Lap}\,(0, \lambda)^p$.

This mechanism provides an unbiased estimate of the function f with bounded variance. We can show that the Laplace mechanism satisfies pure differential privacy.

Theorem 3.1.2 (Theorem 3.6 in Dwork and Roth, 2014)**.** *Let $\epsilon \geq 0$ and $f : \mathcal{D} \to \mathbb{R}^p$ be a query with ℓ_1-sensitivity $\Delta_1(f)$. The Laplace mechanism $\texttt{LM}_\lambda$ with parameter $\lambda = \frac{\Delta_1(f)}{\epsilon}$ is ϵ-differentially private.*

Proof. Let $D \sim D' \in \mathcal{D}$ be two datasets differing on one element. Let g_D and $g_{D'}$ be the density functions of the output of $\texttt{LM}_\lambda(f; D)$ and $\texttt{LM}_\lambda(f; D')$. For $z \in \mathbb{R}^p$,

$$\frac{g_D(z)}{g_{D'}(z)} = \prod_{j=1}^{p} \frac{\exp\left(-\frac{\epsilon|f(D)_j - z_j|}{\Delta_1(f)}\right)}{\exp\left(-\frac{\epsilon|f(D')_j - z_j|}{\Delta_1(f)}\right)} = \prod_{j=1}^{p} \exp\left(\frac{\epsilon|f(D')_j - z_j| - \epsilon|f(D)_j - z_j|}{\Delta_1(f)}\right) \ ,$$

where $f(D)_j$ is the j-th coordinate of $f(D)$, and similarly for $f(D')$. By the triangle inequality and the properties of the exponential function we obtain

$$\frac{g_D(z)}{g_{D'}(z)} \leq \prod_{j=1}^{p} \exp\left(\frac{\epsilon|f(D)_j - f(D)_j|}{\Delta_1(f)}\right) = \exp\left(\frac{\epsilon\|f(D) - f(D)\|_1}{\Delta_1(f)}\right) \leq \exp(\epsilon) \ ,$$

which proves that $\texttt{LM}_\lambda$ satisfies ϵ-differential privacy. □

Note the importance of the sensitivity: to achieve a fixed ϵ-differential privacy guarantee, the noise has to be proportional to the ℓ_1 sensitivity of the query.

For vector-valued queries, the ℓ_1-sensitivity can be quite large. Such queries often have a much smaller ℓ_2-sensitivity: being able to calibrate the noise to this sensitivity could thus greatly reduce the overall variance of the noise. To do this, we need to change the distribution of the noise from Laplace to Gaussian. The Gaussian distribution has the following density function

$$\mathcal{N}(x|\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{|x|^2}{2\sigma^2}\right) \quad , \quad \text{for } x \in \mathbb{R} \ . \tag{3.1.9}$$

We will denote $X \sim \mathcal{N}(\sigma^2)^p$ a random variable taking its values in $\mathbb{R}^p$ whose coordinates are independently sampled from the Gaussian distribution with parameter σ^2. This leads to defining the *Gaussian mechanism* as follows.

Algorithm 3.1.3: $\texttt{GM}_\sigma$: Gaussian Mechanism.

Input: query $f : \mathcal{D} \to \mathbb{R}^p$, dataset D, noise scale $\sigma^2 > 0$.

Return: $f(D) + \mathcal{N}(0, \sigma^2)^p$.

Like the Laplace mechanism, this mechanism gives an unbiased estimator of f with bounded variance. The Gaussian mechanism also satisfies differential privacy.

Theorem 3.1.3 (Corollary 3 in Mironov, 2017)**.** *Let $\alpha > 1$, $\epsilon > 0$, and $f : \mathcal{D} \to \mathbb{R}^p$ be a query with ℓ_2-sensitivity $\Delta_2(f)$. The Gaussian mechanism $\texttt{GM}_\sigma$ with parameter $\sigma^2 = \frac{\alpha\Delta_2(f)^2}{2\epsilon}$ is (α, ϵ)-Rényi differentially private.*

Remark 3.1.2 (From Theorem 3.22 in Dwork and Roth (2014))**.** *We can also directly give approximate differential privacy for the Gaussian mechanism. It is* (ϵ, δ)*-differentially private as long as* $\sigma^2 \geq \frac{2\log(1.25/\delta)\Delta_2(f)^2}{\epsilon^2}$.

This theorem follows from Proposition 7 of Mironov (2017). Note that the variance of the noise scales with α: this highlights that the Gaussian mechanism cannot guarantee pure differential privacy (as this would require taking $\alpha \to +\infty$, see Section 3.1.2 (b)). However, this limitation is compensated by the fact that the variance of the noise in the Gaussian mechanism depends on the ℓ_2-sensitivity of the query. This sensitivity can be up to $\sqrt{p}$ times lower than the ℓ_1-sensitivity. It is therefore natural to use this mechanism to design differentially private mechanisms.We will see in Section 3.2.3 that the Gaussian mechanism is an important building block in the design of differentially private optimization algorithms.

3.1.3 (d) Report Noisy Max

Sometimes, we do not need to release the complete sensitive query. A notable example is when we aim at computing the index of the maximal element of a sensitive vector. In this case, we can use the Report Noisy Max mechanism (Dwork and Roth, 2014). This mechanism adds Laplace noise to each coordinate of the vector, and release the maximum of the noisy values.

Algorithm 3.1.4: $\texttt{RNM}_\lambda$: Report Noisy Max.

Input: queries $f_k : \mathcal{D} \to \mathbb{R}$ for $k \in [K]$, dataset D, noise scales $\lambda_k > 0$.

For $k = 0$ to K:

Compute $u_k = f_k(D) + \mathrm{Lap}(\lambda)$.

Return: $\arg\max_{k \in [K]} u_k$.

This mechanism preserves pure differential privacy.

Theorem 3.1.4 (Privacy of the Report Noisy Max Mechanism)**.** *Let* $\epsilon > 0$*. For* $k \in [K]$*, let* $f_k : \mathcal{D} \to \mathbb{R}$ *be a query with sensitivity* $\Delta(f_k)$ *(note that since* f_k *is scalar, its* ℓ_1 *and* ℓ_2 *sensitivities coincide). The Report Noisy Max Mechanism* $\texttt{RNM}_\lambda$ *with parameter* $\lambda \in \mathbb{R}^p$ *such that* $\lambda_k = \frac{\Delta(f_k)}{\epsilon}$ *is* ϵ*-differentially private.*

The proof can be found in Claim 3.9 from Dwork and Roth (2014). There, the fundamental observation is that the noise we add in each f_k only depends on its sensitivity,

rather than on the complete ℓ_1-sensitivity of the query $D \mapsto (f_1(D), \dots, f_K(D))$, which would be much larger. This mechanism will be at the core of the differential privacy greedy coordinate descent algorithm, which will be the object of Chapter 5.

3.1.4 Building More Complex Mechanisms

The basic mechanisms that we described above can be used to build more complex mechanisms. This is at the root of all differentially private optimization algorithms. The two essential ideas are that (i) releasing the output of multiple differentially private algorithms *on the same dataset* still guarantee differential privacy (although with looser guarantees), and (ii) sampling a fraction of the dataset and running a differentially private algorithm on this sub-sample amplifies privacy guarantees.

3.1.4 (a) Composition

Each time a database is queried, more information gets leaked. This can be accounted for by tracking the evolution of the privacy budget as the same database gets queried. We aim at quantifying the differential privacy guarantees of $K > 0$ algorithms $\mathcal{A}_1, \dots, \mathcal{A}_K$ on the same dataset. Importantly, we also allow the output of each mechanism to depend on the output of the previous ones. Formally, we aim at quantifying the differential privacy guarantee satisfied by the mechanism recursively defined by,

$$\mathcal{A}_{\text{comp}}^{(k)} : D \mapsto \mathcal{A}_K(D; a_1, \dots, a_{k-1}) \ , \quad \text{for } k \in [K] \ , \tag{3.1.10}$$

where, for $i \leq k$, a_i is the output of the i-th mechanism $\mathcal{A}_{\text{comp}}^{(i)}(D)$. We first state the following theorem, that gives the privacy guarantees of the composition of (ϵ, δ)-differentially private algorithms.

Theorem 3.1.5 (Theorem 3.20 and Corollary 3.21 in Dwork and Roth, 2014)**.** *Let $\epsilon > 0$, $\delta, \delta_0 \in [0, 1]$, and $\mathcal{A}_1 : \mathcal{D} \to \mathcal{E}_1$, $\mathcal{A}_2 : \mathcal{D} \times \mathcal{E}_1 \to \mathcal{E}_2$, …, $\mathcal{A}_K : \mathcal{D} \times \mathcal{E}_1 \times \cdots \times \mathcal{E}_{K-1} \to \mathcal{E}_K$ be a sequence of (ϵ, δ)-differentially private algorithms. Then the algorithm $\mathcal{A}_{comp}^{(K)}$ as defined in (3.1.10) satisfies (ϵ', δ')-differential privacy with*

$$\epsilon' = \sqrt{2K \log(1/\delta_0)}\epsilon + K\epsilon(\exp(\epsilon - 1)) < \ , \quad \text{and} \quad \delta' = k\delta + \delta_0 \ . \tag{3.1.11}$$

When the target ϵ' is smaller than 1, and $\delta' > 0$, we can simplify this expression as $\epsilon' = \sqrt{2K \log(1/\delta_0)}\epsilon + K\epsilon^2$. Consequently, it suffices to set $\epsilon = \frac{\epsilon'}{\sqrt{8K \log(1/\delta_0)}}$ to obtain the desired target value for ϵ'.

This theorem is particularly helpful when composing pure ϵ-differentially private mechanisms, where it yields reasonably tight guarantees. We will notably use this

result to compose Laplace $\texttt{LM}_\lambda$ and Report Noisy Max $\texttt{RNM}_\lambda$ mechanisms in Chapter 5 when designing our differentially private greedy coordinate descent algorithm.

When composing approximate (ϵ, δ)-differentially private mechanisms (*e.g.,* the Gaussian mechanism $\texttt{GM}_\sigma$), it is often tighter to use the composition theorem of Rényi differential privacy. The result can then be converted back to the usual differential privacy using Proposition 3.1.1. These mechanisms can be composed as follows.

Theorem 3.1.6 (Proposition 1 in Mironov, 2017)**.** *Let $\alpha > 1$ and $\mathcal{A}_1 : \mathcal{D} \to \mathcal{E}_1$, $\mathcal{A}_2 : \mathcal{D} \times \mathcal{E}_1 \to \mathcal{E}_2$, ..., $\mathcal{A}_K : \mathcal{D} \times \mathcal{E}_1 \times \cdots \times \mathcal{E}_{K-1} \to \mathcal{E}_K$ be a sequence of (α, ϵ_k)-Rényi differentially private algorithms. Then the algorithm $\mathcal{A}^{(K)}_{comp}$ as defined in (3.1.10) satisfies (α, ϵ)-Rényi differentially private with parameter $\epsilon = \sum_{k=1}^{K} \epsilon_k$.*

3.1.4 (b) Privacy Amplification by Sampling

The last building block that we need is a privacy amplification result. This allows to characterize the privacy guarantees achieved when a differentially private algorithm is run on a random sub-sample of dataset. Formally, we define the $\texttt{Sample}_q$ mechanism, for $q \in [0, 1]$, as follows.

> **Algorithm 3.1.5:** $\texttt{Sample}_q$: sampling mechanism.
>
> **Input:** dataset D of n records, fraction of the samples q.
>
> **Return:** sample of $\lfloor qn \rfloor$ records uniformly sampled from D.

Composing the $\texttt{Sample}_q$ mechanism with a database query can greatly affect its sensitivity. For instance, consider the averaging query from Example 3.1.1, composed with $\texttt{Sample}_q$. It returns $f \circ \texttt{Sample}_q(D) = \frac{1}{qn} \sum_{i \in S_q} d_i$, where S_q is the subset of size qn sampled by $\texttt{Sample}_q$. The sensitivity of the composed mechanism is increased by a factor $1/q$, that is $\Delta_2(f \circ \texttt{Sample}_q) = \frac{1}{qn}$.

Fortunately, when releasing $f \circ \texttt{Sample}_q$ privately, the mere fact of sampling a subset of the dataset improves privacy guarantees. This phenomenon was notably studied by Li et al. (2012), Balle et al. (2018), Balle et al. (2020), and Steinke (2022) who proved the following theorem.

Theorem 3.1.7 (Theorem 9, Balle et al., 2020)**.** *Let $q \in [0, 1]$, $\epsilon \geq 0$, $\delta \in [0, 1]$, and $\mathcal{A} : \mathcal{D} \to \mathcal{E}$ be a (ϵ, δ)-differentially private algorithm Then $\mathcal{A} \circ \texttt{Sample}_q$ is $(\epsilon', q\delta)$-differentially private with $\epsilon' = \log(1 + q(\exp(\epsilon) - 1))$.*

For small values of ϵ, we have $\epsilon' = \log(1 + q(\exp(\epsilon) - 1)) \approx q\epsilon$. Therefore, Theorem 3.1.7 essentially means that privacy guarantees are amplified by a factor q. In practice, this compensates for the increase in sensitivity mentioned above.

Similar results exist for Rényi differential privacy (Mironov et al., 2019; Wang et al., 2019a). For the sampled Gaussian mechanism, one can derive the following amplification result, that follows from the instantiation of the results of Bun et al. (2018) to the Gaussian mechanism.

Theorem 3.1.8 (Table 1 in Mironov et al., 2019, adapted from Bun et al., 2018)**.** *Let $q \in [0, 1/10]$, $\sigma > \sqrt{5}$ and $1 < \alpha \leq \frac{1}{2}\sigma^2 \log(1/q)$. Then the sampled Gaussian mechanism $\mathtt{GM}_\sigma \circ \mathtt{Sample}_q$ with parameter σ^2 is $(\alpha, \frac{6q^2\alpha}{\sigma^2})$-Rényi differentially private.*

Note that Mironov et al. (2019) and Wang et al. (2019a) also give tighter results without conditions on α, σ, although without a closed form expression. In Section 3.2.3 (b) these results will allow us to assess the privacy guarantees of the differentially private stochastic gradient descent algorithm.

3.2 Differentially Private Machine Learning

One may think that, as machine learning works with aggregated quantities on possibly large datasets, it does not contain confidential information. However, as any computation done on a set of data, they are subject to the rule stated in Section 3.1.2: any useful computation done on a dataset leaks information on this dataset. Therefore, special care has to be taken to train these models while preserving data privacy.

In this section, we give in Section 3.2.1 a quick tour of existing attacks on machine learning models. These attacks highlight the reality of privacy leaks, and thus, the necessity of addressing them. To this end, we introduce the differentially private empirical risk minimization problem (DP-ERM) (Chaudhuri et al., 2011), which aims at training machine learning models under differential privacy. Then, we describe some classical approaches for solving it in Section 3.2.3, and discuss their usability in practice. Finally, we describe utility lower bounds in Section 3.2.4. Under the usual assumptions on the objective, these worst-case lower bounds are nearly matched by the methods proposed in Section 3.2.3. Nonetheless, we will see in Chapter 4 that these lower bounds can be refined when regularity is measured in a coordinate-wise manner.

3.2.1 Privacy Leaks in Machine Learning

Although trained machine learning models aim at finding general patterns that apply to the complete population, they still tend to leak some information on their training data. This has been demonstrated in practice through inference attacks, that try to reconstruct (part of) the training data. We now describe two types of such attacks: membership inference attacks, and reconstruction attacks.

Membership Inference Attacks. Membership inference attacks aim at inferring, from the result of a query, whether an individual was present in the data or not. These attacks correspond to the point of view of the attacker that differential privacy tries to protect against. They were introduced by Shokri et al. (2017), who proposed to attack a model as a black box. In this setting, the adversary has a black-box access to the model (*i.e.,* they can query the model with arbitrary, and obtain the values predicted by the model). Their attack works in three steps: (i) generate synthetic data that is somewhat similar to the training data, then (ii) train multiple models on parts of this data, and (iii) train another classifier on the *prediction of the models from previous step*, using as label the membership of the record in the training data of the model. The resulting model is expected to be able to distinguish between a point used in training a model and another one. This kind of approach have then notably been studied by Yeom et al. (2018), Truex et al. (2019), and Ye et al. (2022). Other settings, where adversary can do more than simply querying the model have also been studied (Nasr et al., 2019; Sablayrolles et al., 2019; Melis et al., 2019).

We note that most of the aforementioned works assume that the attacker possesses a set of candidate records that contains a large fraction of true training records. This setting is in line with the guarantees of differential privacy (where the adversary may *know all records but one* and still be unable to re-identify anyone), but may not be overly realistic. The works of Jayaraman et al. (2021) and Carlini et al. (2021a) studied a harder (but more realistic) setting where few of the records from the candidate set are actual training records.

We refer to Hu et al. (2022a) for a detailed overview of membership inference attacks.

Reconstruction Attacks. Attribute inference attacks aim to reconstruct (part of) the training data from a trained model or from intermediate computations like gradients. This threatens individuals' privacy since all their personal information (present in a private dataset) could be reconstructed by malicious parties.

Multiple works have shown that such attacks are possible from gradients. For instance, consider the federated learning setting, where multiple agents learn a model collectively using `FedAvg` with one local step (McMahan et al., 2017). The server asks an agent to compute a gradient $\nabla f(w; d)$, for some function f that depends on the agent's local data d and some parameters $w \in \mathbb{R}^p$. In such setting, Phong et al. (2017), Wang et al. (2019b), and Zhu et al. (2019) showed that one can reconstruct the data by solving a problem similar to this:

$$\min_{d'} \|\nabla f(w; d') - g\|^2 \ , \qquad \text{where } g = \nabla f(w; d) \ .$$

This can notably be done by the server that orchestrates the training, who knows everything but the data record. These approaches were extended to mini-batch `FedAvg`

by Geiping et al. (2020) and Wen et al. (2022). Similarly, Fowl et al. (2022) proposed to modify models during training so that the data is obtained completely without having to solve such a problem.

Finally, some works have studied reconstruction attacks from trained models. This was notably studied by Guo et al. (2022) and Balle et al. (2022) in a general setting. More specific works considered high-dimensional linear models (Paige et al., 2021), generative models (Wang et al., 2009; Carlini et al., 2023) and language models Carlini et al. (2019) and Carlini et al. (2021b).

In the remainder of this Chapter, we give an overview of the methods that can be set up to limit the possibility for an adversary to perform the attacks we just described.

3.2.2 Differentially Private Empirical Risk Minimization

A general method for training machine learning models in a differentially private way is differentially private empirical risk minimization (Chaudhuri et al., 2011). Let $\mathcal{X}$ be a feature space and $\mathcal{Y}$ a label space. Suppose that a trusted data curator has access to a data set $D = \{d_1, \dots, d_n\} \subseteq (\mathcal{X} \times \mathcal{Y})^n$ of n records. To train a model privately, one aims at designing an (ϵ, δ)-differentially private algorithm that computes an approximation w^{priv} of

$$w^* \in \arg\min_{w \in \mathcal{W}} \left\{ F(w) := f(w) + \psi(w) \right\} \ , \quad \text{with } f(w) = \frac{1}{n} \sum_{i=1}^{n} \ell(w; d_i) \ , \qquad (\star')$$

which is a special instance of $(\star)$ with $f_i(w) = \ell(w; d_i)$, for some loss function $\ell : \mathcal{W} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. We assume that $\mathcal{W} \subseteq \mathbb{R}^p$ is a closed convex set, $\ell(\cdot; d)$ is smooth and proper convex for all $d \in D$, and ψ is proper convex. The function f is the only function that depends on the data, and ψ is a regularizer that controls the model's complexity and structure. In the following, we call *utility* the expected suboptimality gap. Specifically, if an algorithm outputs w^{priv}, we measure its utility as $\mathbb{E}(F(w^{\text{priv}})) - F(w^*)$, where the expectation is over the randomness of the algorithm.

3.2.3 Solving Differentially Private Empirical Risk Minimization

Multiple approaches have been proposed for solving $(\star')$ using differentially private algorithms. In this section, we describe two these methods: output perturbation and differentially private stochastic gradient descent (`DP-SGD`). We then discuss some practical considerations that are essential for the real-world use of these mechanisms.

3.2.3 (a) Output Perturbation

One very simple way of finding an (ϵ, δ)-differentially private solution to $(\star')$ is to compute a solution w^* and release it using the Gaussian mechanism (see Section 3.1.3 (c)). This approach coined *output perturbation* was studied by Chaudhuri and Monteleoni (2008) and Chaudhuri et al. (2011) and later by Lowy and Razaviyayn (2021).

Algorithm 3.2.1: Output Pertubation.

Input: dataset D, noise scale $\sigma > 0$.

Compute $w^* \in \arg\min_{w \in \mathcal{W}} \{F(w)\}$.

Return: $w^{\text{priv}} = w^* + \mathcal{N}(0, \sigma^2)^p$.

To assess the privacy guarantees of this mechanism, we need to compute the sensitivity of the function that maps a dataset to a solution of $(\star')$. To do so, we need the solution to be unique, and to derive a bound on how much it can change when one element of the dataset changes.

Theorem 3.2.1. *Let $\epsilon \geq 0$, $\delta \in [0, 1]$. Assume f is differentiable, L-Lipschitz, μ-strongly convex, and has a finite minimum. Then, Algorithm 3.2.1 with $\sigma^2 = \frac{2\log(1.25/\delta)L^2}{\mu^2 n^2 \epsilon^2}$ is (ϵ, δ)-differentially private.*

Proof. To prove this theorem, we study the sensitivity of the function

$$g : D \mapsto \arg\min_{w \in \mathcal{W}} F(w) \ .$$

First, since F is μ-strongly convex and has a finite minimum, it has a unique minimizer, and g is well-defined. We now let D and D' be two datasets differing on their first element. Let w_1^* and w_2^* be the minimizers of $F(\cdot; D)$ and $F(\cdot; D')$ respectively. Since f is differentiable and strongly convex, we can use (2.1.15) to obtain

$$F(w_1^*; D) \leq F(w_2^*; D) - \tfrac{\mu}{2}\|w_1^* - w_2^*\|^2 \tag{3.2.1}$$
$$F(w_2^*; D) \leq F(w_1^*; D) + \langle \nabla F(w_2^*; D), w_2^* - w_1^* \rangle - \tfrac{\mu}{2}\|w_1^* - w_2^*\|^2 \ . \tag{3.2.2}$$

Now we remark that $F(w_2^*; D) = F(w_2^*; D') + \frac{1}{n}(\ell(w_2^*; d_1') - \ell(w_2^*; d_2')$. Therefore, since w_1^* and w_2^* are the minimum of $F(\cdot; D)$ and $F(\cdot; D')$, we have $\nabla F(w_2^*; D') = 0$ and

$$\nabla F(w_2^*; D) = \tfrac{1}{n}(\nabla \ell(w_2^*; d_1') - \nabla \ell(w_2^*; d_2')) \ . \tag{3.2.3}$$

Summing (3.2.1) and (3.2.2) and replacing the value of $\nabla F(w_2^*; D)$, we obtain

$$\mu\|w_1^* - w_2^*\|^2 \leq \tfrac{1}{n}\langle \nabla \ell(w_2^*; d_1') - \nabla \ell(w_2^*; d_1), w_2^* - w_1^* \rangle \leq \tfrac{2L}{n}\|w_2^* - w_1^*\| \ , \tag{3.2.4}$$

which implies that the sensitivity of g is $\Delta_2(g) \leq \frac{2L}{\mu n}$. The theorem follows from the differential privacy guarantees of the Gaussian mechanism stated in Remark 3.1.2. □

Theorem 3.2.1 proves that the output perturbation mechanism is differentially private. Nonetheless, we stress that for the theorem to hold, the computation of the minimizer of the ($\star'$) problem has to be *computed exactly*. In general, it may be difficult to guarantee the exactitude of this computation. In the next section, we discuss an algorithm that directly computes a differentially private value without relying on the exact computation of the solution.

Remark 3.2.1. *A very related method is* objective perturbation *(Chaudhuri et al., 2011). Instead of finding the true value, then perturbing it, the objective is augmented with an additive noise term, which guarantees the privacy of the solution. These type of algorithms have been studied by Kifer et al. (2012) on sparse problems, and Neel et al. (2020) studied it under various sets of assumptions.*

3.2.3 (b) Differentially Private Stochastic Gradient Descent

In this section, we describe the most widely used algorithm for solving the $\star'$ problem: *differentially private stochastic gradient* (`DP-SGD`). `DP-SGD` is a variant of the `SGD` algorithm that we described in Section 2.2.2. Contrary to output perturbation, it can solve ($\star'$) even on non-strongly-convex losses.

`DP-SGD` was initially proposed by Song et al. (2013) and Jain et al. (2012). Then, Bassily et al. (2014b) proved the optimality of `DP-SGD`'s utility, and Wang et al. (2017) studied variance-reduced variants of `DP-SGD` to improve the efficiency of the algorithm (although for the same utility). `DP-SGD` has also been widely studied as a minimizer of the population risk, see Duchi et al. (2013), Bassily et al. (2019), and Feldman et al. (2020). We describe the proximal variant of this algorithm, that can handle non-smooth regularizers.

Algorithm 3.2.2: `DP-SGD`: Differentially Private Proximal `SGD`.

Input: initial point w^0, step sizes $\gamma_0, \ldots, \gamma_T > 0$, noise scale $\sigma > 0$.

For $t = 0$ to $T - 1$:

Sample index i uniformly at random in $[n]$

Update $w^{t+1} = \mathrm{prox}_{\gamma_t \psi}\left(w^t - \gamma_t(\nabla f_i(w^t) + \eta^t\right)$, where $\eta^t \sim \mathcal{N}(0, \sigma^2)$.

Return: w^T.

At each iteration of DP-SGD, we sample some record from the data, and add Gaussian noise. This allows to give differential privacy guarantees for DP-SGD.

Theorem 3.2.2 (Adapted from Theorem II.2 in Bassily et al. (2014b))**.** *Let $\epsilon \geq 0$ and $\delta \in [0,1]$. There exist a value of $\sigma^2 = O(\frac{TL^2\log(1/\delta)}{n^2\epsilon^2})$, that can easily be computed numerically, such that DP-SGD with parameter σ^2 is (ϵ,δ)-differentially private.*

Proof. To prove the theorem, we study the Rényi differential privacy of DP-SGD, and convert them back to the usual differential privacy. Let $\alpha > 0$. From Theorem 3.1.3, each iteration of DP-SGD is $(\alpha, \frac{\alpha L^2}{2\sigma^2})$-Rényi differentially private. Since each iteration operates on a sample of size $\frac{1}{n}$ of the data, this guarantee is amplified by a factor $O(1/n^2)$ (see Theorem 3.1.8). Using the composition result from (3.1.10), we obtain that DP-SGD is $(\alpha, \frac{\alpha TL^2}{2n^2\sigma^2})$-Rényi differentially private. Converting back to (ϵ,δ)-differential privacy using Proposition 3.1.1 gives the result. □

Remark 3.2.2. *In Theorem 3.2.2, we write $\sigma^2 = O(\frac{TL^2\log(1/\delta)}{n^2\epsilon^2})$ uniquely to give an intuition on the scale of the noise required for DP-SGD to satisfy differential privacy. In practice, this value is computed numerically by tuning the parameter α of Rényi differential privacy properly.*

The utility of DP-SGD has first been formally studied by Bassily et al. (2014b) under the assumption that f is differentiable (and not necessarily smooth), L-Lipschitz and convex. They use an indicator function, $\psi = \iota_{\mathcal{W}}$, so that the iterates are projected on the convex set $\mathcal{W}$ at each iteration.

Theorem 3.2.3 (Theorem II.4 in Bassily et al., 2014b)**.** *Let f be differentiable, L-Lipschitz and convex, and take $\psi = \iota_{\mathcal{W}}$ so that $\mathrm{prox}_{\gamma_t\psi}$ is the projection on the set $\mathcal{W}$. Define $w^* \in \arg\min_{w\in\mathcal{W}}$ a minimizer of f, and denote $\|\mathcal{W}\|_2$ the diameter of $\mathcal{W}$. Set the number of iterations to $T = n^2$, and $\sigma^2 = O(\frac{L^2T\log(1/\delta)}{n^2\epsilon^2})$.*

- *If F is convex, set the step size $\gamma_t = \frac{\|\mathcal{W}\|_2}{\sqrt{t(n^2L^2+p\sigma^2)}}$. Then the output of DP-SGD achieves the utility*

$$\mathbb{E}(F(w^{priv})) - F(w^*) = O\left(\frac{L\|\mathcal{W}\|_2\sqrt{p\log(1/\delta)}}{n\epsilon}\right) .$$

- *If F is μ-strongly-convex with respect to the ℓ_2-norm, set $\gamma_t = \frac{1}{\mu nt}$. Then the output of DP-SGD achieves the utility*

$$\mathbb{E}(F(w^{priv})) - F(w^*) = O\left(\frac{L^2p\log(1/\delta)}{\mu n^2\epsilon^2}\right) .$$

Proof. The proof relies on the following bounds on the gradients expected square norm, for any $w \in \mathcal{W}$, and $i \sim \mathcal{U}([n])$,

$$\begin{aligned}\mathbb{E}(\|\nabla \ell(w; d_i) + \eta^t\|^2) &= \mathbb{E}(\|\nabla \ell(w; d_i)\|^2) + 2\,\mathbb{E}(\langle \nabla \ell(w; d_i), \eta^t \rangle) + \mathbb{E}(\|\eta^t\|^2) \\ &\leq L^2 + p\sigma^2 \ ,\end{aligned}$$

where the inequality comes from the uniform bound $\|\nabla \ell(w; d_i)\|^2 \leq L$, the fact that $\mathbb{E}(\eta^t) = 0$ and the fact that σ^2 is the variance of each coordinate of η^t.

The bounds then follow from the analysis of projected `SGD` from Shamir and Zhang (2013), that relies on the availability of a bound on the squared norm of stochastic gradients. With our choice of step sizes, their results state that, for convex functions (see their Theorem 2 with $c = \|\mathcal{W}\|_2 \epsilon / L\sqrt{p \log(1/\delta)}$), that

$$\begin{aligned}\mathbb{E}(F(w^T) - F(w^*)) &\leq \frac{8\sqrt{p \log(1/\delta)} L \|\mathcal{W}\|_2 \log(T)}{\epsilon \sqrt{T}} + \frac{4\sqrt{p}\epsilon \|\mathcal{W}\|_2 \sigma^2 \log(T)}{L\sqrt{T}} \\ &= \frac{8\sqrt{p \log(1/\delta)} L \|\mathcal{W}\|_2 \log(T)}{\epsilon \sqrt{T}} + O\left(\frac{4\sqrt{pT \log(1/\delta)} L \|\mathcal{W}\|_2 \log(T)}{n^2 \epsilon}\right) \ ,\end{aligned}$$

and for μ-strongly-convex functions (see their Theorem 1), that

$$\mathbb{E}(F(w^T) - F(w^*)) \leq \frac{34L^2 \log(T)}{\lambda T} + \frac{34p\sigma^2 \log(T)}{\mu T} = \frac{34L^2 \log(T)}{\lambda T} + O\left(\frac{L^2 p \log(T)}{\mu n^2 \epsilon^2}\right) \ .$$

The result follows from setting $T = n^2$ in these two inequalities, which balances the two terms. □

This theorem gives a theoretical guarantee on the utility of `DP-SGD`. The proof of these utility results highlights the tension between the optimization error, which decreases with the number of iterations, and the noise due to privacy, which increases with the number of iterations. We stress that the latter increases, not because of noise accumulation, but because of the composition of multiple queries over the same database: this requires increasing the variance of the noise to keep a constant privacy budget. We will observe the same phenomenon in the utility analyses we carry in Chapters 4 and 5 for the `DP-CD` and `DP-GCD` algorithms.

We note that these utility upper bounds grow polynomially with the dimension. This is due because, at each iteration, we add noise on each of the gradient's coordinates. We will see in Chapter 5 that, in some cases, doing coordinate-wise updates on properly chosen coordinates can reduce this dependence from polynomial to logarithmic. Nonetheless, under the general assumptions of Theorem 3.2.3, it is not possible to achieve better utility (see Section 3.2.4). Therefore, *under the assumptions of Theorem 3.2.3*, `DP-SGD` optimally solves the $\star'$ problem.

Remark 3.2.3. *A result similar to Theorem 3.2.3 can be obtained for* `DP-SGD` *with general ψ and fixed step size, based on the convergence result of Proximal* `SGD` *from Khaled et al. (2020). This requires special attention, but the arguments are the same as in the proof of Theorem 3.2.3.*

3.2.3 (c) Differentially Private Machine Learning in Practice

When trying to train a model privately, several practical challenges arise. First, the sensitivity estimated from the Lipschitz constant of the loss typically overestimate the actual norm of the gradient, preventing algorithms like `DP-SGD` from finding a meaningful model. Second, optimization algorithms often crucially depend on the choice of their hyperparameters, which can be difficult to choose privately. We give a brief overview of methods that have been proposed to address these issues in practice.

Gradient Clipping. In gradient-based algorithms like `DP-SGD`, we compute a differentially private approximation of the gradient using, for instance, the Gaussian mechanism. To do so, we need a bound on the sensitivity of this gradient, which can be obtained through the Lipschitz assumption, as we discussed in Section 2.1.4 (a)). Since this bound needs to hold uniformly for all gradients of the loss, on all data records, it can be very high compared to the actual value of the gradients. In some problems (*e.g.,* deep neural networks), it may also be difficult to compute this constant tightly to begin with. To mitigate these issues, practical implementations of `DP-SGD` often use gradient clipping, as described by Abadi et al. (2016a). We set a threshold $C > 0$, and clip the gradients whose norm is higher than this threshold as follows:

$$\mathrm{clip}(\nabla \ell(w; d); C) = \min\left(1, \frac{C}{\|\nabla \ell(w; d)\|}\right) \nabla \ell(w; d). \tag{3.2.5}$$

This guarantees that the clipped gradient is bounded by C. This has two important consequences in terms of privacy: (i) it guarantees that *for any record, the gradient will be bounded by C* (even if it has an unexpectedly high value), and (ii) it reduces the sensitivity of the gradient from $2L$ to $2C$, which can be much lower.

In practice, clipping is indispensable to obtain reasonable utility while guaranteeing privacy. It is notably used in all implementations of `DP-SGD` (see *e.g.,* PyTorch Opacus (Yousefpour et al., 2022), and TensorFlow Privacy (Abadi et al., 2016b)).

Unfortunately, clipping introduces bias in the gradient, as not all individual gradients are clipped the same. This can be interpreted as a bias-variance trade-off (Amin et al., 2019): low clipping induces large bias, but small variance, whereas high clipping induces little to no bias, but large variance. Nonetheless, this bias is not always a problem. For instance, Chen et al. (2020) highlighted that when gradients follow a

symmetric distribution, clipping does not introduce that much bias. It may also be possible to reduce this bias by setting the clipping threshold adaptively (Pichapati et al., 2019; Andrew et al., 2021), although this is difficult to do using only private information on the gradients. Finally, we note that the recent work of Yang et al. (2022) and Koloskova et al. (2023) analyzed clipping in `DP-SGD` under a relative smoothness assumption, highlighting the fact that the choice of the threshold C indeed introduces bias.

Hyperparameter Tuning. The utility of differentially private optimization algorithms like `DP-SGD` is highly dependent on the choice of their hyperparameters. Classical methods for hyperparameter tuning (*e.g.,* grid-search) require running the algorithm multiple times. Adapting them to the differentially private setting is thus a challenging task. A naive solution is to run the algorithm with different sets of hyperparameters, and use composition results to guarantee privacy of the complete procedure. While this preserves differential privacy, it generally destroys utility.

In general, when selecting hyperparameters, we are only interested in finding the best ones: it should thus be possible to improve the naive solution by not releasing the outputs of runs that gave poor results. This idea was first explored by Chaudhuri and Vinterbo (2013), who used a stability assumption to reduce the overall budget of the tuning. Later on, Liu and Talwar (2019) proposed a more general method (*i.e.,* without the stability assumption), based on private selection methods, that are similar to the report noisy max mechanism (see Section 3.1.3 (d)). Their work was further extended to Rényi differential privacy by Papernot and Steinke (2022). Other approaches have also been proposed, based on adaptive algorithms (Mohapatra et al., 2022; Priyanshu et al., 2021), or on running algorithms on subsets of the data to reduce the privacy loss (Koskela and Kulkarni, 2023).

3.2.4 Utility Lower Bounds

For a given privacy budget the problem ($\star'$) can not be solved up to arbitrary precision. This is due to the fact that solving a problem too precisely could allow to infer the presence of some individuals in the training data. For (ϵ, δ)-differential privacy, the following theorem states lower bounds on utility.

Theorem 3.2.4 (Utility Lower Bounds for DP-ERM, see Theorems V.3 and V.5 in Bassily et al., 2014b)**.** *Let $n, p > 0$, $\epsilon > 0$ and $\delta = o(1/n)$, and assume that $\mathcal{W}$ is bounded with diameter $\|\mathcal{W}\|_2$. For every (ϵ, δ)-differentially private algorithm $\mathcal{A}$, there exists a dataset D such that:*

If F is convex, with probability at least $1/2$*,*

$$F(\mathcal{A}(D)) - F(w^*) = \Omega\left(L\|\mathcal{W}\|_2 \min\left(1, \frac{\sqrt{p}}{n\epsilon}\right)\right) \quad . \tag{3.2.6}$$

*If F is μ-strongly convex (*w.r.t., *ℓ_2-norm), with probability at least* $1/3$*,*

$$F(\mathcal{A}(D)) - F(w^*) = \Omega\left(\frac{L^2}{\mu} \min\left(1, \frac{p}{n^2\epsilon^2}\right)\right) \quad . \tag{3.2.7}$$

These results are based on the results of Bun et al. (2014), who studied counting queries under (ϵ, δ)-differential privacy. Their results are based on the work of Boneh and Shaw (1998) and Tardos (2008) about fingerprinting codes. These codes were originally designed to protect against piracy on proprietary software: each copy is equipped with a hidden serial number, and pirates aim at making up fake serial numbers. Since pirates do not know the location of all digits of this serial number, they can only change some of them: examining the fake number, they could trace it back to the original copies of the pirates. The idea behind Bun et al. (2014)'s lower bounds is that counting queries can be used as a way of finding back these pirates (leading to reidentification). Bassily et al. (2014b) then further reduced the DP-ERM problem to computing counting queries. In Section 4.4, we provide a refined version of these lower bounds, where Lipschitzness of the objective is measured in a coordinate-wise manner rather than on the full function.

Note that Bassily et al. (2014b) give similar results for pure ϵ-differential privacy, based on the work of Hardt and Talwar (2010). We do not state these results since all our results will be stated in terms of approximate differential privacy.

Finally, we note that Talwar et al. (2016) proved that the lower bound from Theorem 3.2.4 on convex objective function does not hold when the ℓ_1 diameter of $\mathcal{W}$, $\|\mathcal{W}\|_1$ is independent of the dimension. In this case, the lower bound can be refined to $F(w^{\text{priv}}) - F(w^*) = \Omega(\frac{1}{n^{2/3}})$. Notably, this lower bound is matched (up to logarithmic factors) by differentially private variants of the Frank-Wolfe algorithm (Jaggi, 2013; Frank and Wolfe, 1956). In Chapter 5, we propose an algorithm that nearly matches this lower bound even when $\mathcal{W}$ is unbounded (or has a ℓ_1-diameter that depends on the dimension).

Chapter 4

Private Randomized Coordinate Descent

Chapter Abstract

We propose differentially private proximal coordinate descent (DP-CD), a new differentially private method to solve composite empirical risk minimization (DP-ERM). We derive utility guarantees through a novel theoretical analysis of inexact coordinate descent. Our results show that, thanks to larger step sizes, DP-CD can exploit imbalance in gradient coordinates to outperform DP-SGD. We also prove new lower bounds for composite DP-ERM under coordinate-wise regularity assumptions, that are nearly matched by DP-CD. For practical implementations, we propose to clip gradients using coordinate-wise thresholds that emerge from our theory, avoiding costly hyperparameter tuning.

This Chapter is mostly based on the paper: *"Differentially Private Coordinate Descent for Composite Empirical Risk Minimization"* (Mangold, Bellet, Salmon, and Tommasi, 2022), published at ICML 2022.

The code corresponding to this Chapter is available at https://gitlab.inria.fr/pmangold1/private-coordinate-descent/.

4.1 Introduction

In this Chapter, we propose the differentially private proximal coordinate descent algorithm (DP-CD). This algorithm is based on the proximal coordinate descent method, that we described in Section 2.2.3. Like DP-SGD (see Section 3.2.3 (b)), DP-CD preserves differential privacy by performing updates based on perturbed gradients. At

each iteration, it does a proximal coordinate update using a coordinate-wise gradient, that was computed under differential privacy using the Gaussian mechanism.

We propose DP-CD and analyze its theoretical and empirical convergence properties as a differentially private solver for the composite empirical risk minimization problem. Let $\mathcal{X}$ be a feature space and $\mathcal{Y}$ a label space, and suppose that we have a dataset $D = \{d_1, \dots, d_n\} \subseteq (\mathcal{X} \times \mathcal{Y})^n$ of n records. We study the following unconstrained composite problem:

$$w^* \in \operatorname*{arg\,min}_{w \in \mathbb{R}^p} \left\{ F(w) := f(w) + \psi(w) \right\} \ , \quad \text{with } f(w) = \frac{1}{n} \sum_{i=1}^{n} \ell(w; d_i) \ , \qquad (\star')$$

where $\ell : \mathbb{R}^p \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is a loss function which is convex and coordinate-wise smooth in its first parameter, and $\psi : \mathbb{R}^p \to \mathbb{R}$ is a separable convex regularizer (*i.e.,* $\psi(w) = \sum_{j=1}^{p} \psi_j(w_j)$) that is typically nonsmooth (*e.g.,* ℓ_1-norm).

We start by showing that the proposed algorithm satisfies differential privacy. Importantly, we note that the coordinates of the gradient generally have a much lower sensitivity than the full gradient. This allows DP-CD to compensate its larger number of iterations (which induces larger noise) with smaller sensitivity.[1] We then theoretically analyze the properties of DP-CD by developing a novel analysis of proximal coordinate descent with perturbed (but unbiased) gradients. This allows to derive upper bounds on the privacy-utility trade-off achieved by DP-CD. To this end, we prove a recursion on distances of DP-CD's iterates to an optimal point. Our analysis keeps track of coordinate-wise regularity constants all along, which tightly captures the importance of using large constant step sizes to obtain high utility. Our results highlight the fact that DP-CD can exploit imbalanced gradient coordinates to outperform DP-SGD. We assess the optimality of DP-CD by deriving lower bounds that capture coordinate-wise Lipschitz regularity measures, and show that DP-CD matches those bounds up to logarithmic factors. Our lower bounds also suggest interesting perspectives for future work on DP-CD algorithms.

Our theoretical results also have important consequences for practical implementations, which heavily rely on gradient clipping to achieve good utility. In contrast to DP-SGD, DP-CD requires to set *coordinate-wise* clipping thresholds, which can lead to impractical coordinate-wise hyperparameter tuning. We instead propose a simple rule for adapting these thresholds from a single hyperparameter. We also show how the coordinate-wise smoothness constants used by DP-CD can be estimated privately. We validate our theory with numerical experiments on real and synthetic datasets. These experiments further show that even in balanced problems, DP-CD can still improve

[1] Contrarily to DP-SGD, DP-CD does not rely on privacy amplification by sampling (see Section 3.1.4 (b)), which is not applicable in this setting.

over DP-SGD, confirming the relevance of DP-CD for DP-ERM.

The main contributions of this Chapter can be summarized as follows:

1. We propose the first differentially private proximal coordinate descent method for composite DP-ERM, formally prove its utility, and highlight regimes where it outperforms DP-SGD.
2. We show matching lower bounds under coordinate-wise regularity assumptions.
3. We give practical guidelines to use DP-CD, and show its relevance through numerical experiments.

The rest of this Chapter is organized as follows. We briefly describe some related work in Section 4.2. In Section 4.3, we present our DP-CD algorithm, show that it satisfies differential privacy, establish utility guarantees, and compare these guarantees with those of DP-SGD. In Section 4.4, we derive lower bounds under coordinate-wise regularity assumptions, and show that DP-CD can match them. Section 4.5 discusses practical questions related to gradient clipping and the private estimation of smoothness constants. Section 4.6 presents our numerical experiments, comparing DP-CD and DP-SGD on LASSO and ℓ_2-regularized logistic regression problems.

4.2 Related Work

Prior to our work, only few works have mentioned the idea of a differentially private coordinate descent method. Damaskinos et al. (2021) introduced a coordinate descent method to privately solve the dual problem associated with generalized linear models with ℓ_2 regularization. Dual coordinate descent is tightly related to SGD, as each coordinate in the dual is associated with one data point. The authors briefly mention the possibility of performing primal coordinate descent but discard it on account of the seemingly large sensitivity of its updates. We show that primal DP-CD is in fact quite effective, and can be used to solve more general problems than considered by Damaskinos et al. (2021).

Primal coordinate descent was also successfully used by Bellet et al. (2018) to privately learn personalized models from decentralized datasets. For the smooth objective they consider, each coordinate depends only on a subset of the full dataset, which directly yields low coordinate-wise sensitivity updates. In contrast, we introduce a general algorithm for composite DP-ERM, for which a novel utility analysis is required.

For a general overview of related work on non-private coordinate descent, we refer to the discussions in Section 2.2.3.

4.3 Differentially Private Coordinate Descent

In this section, we introduce the differentially private proximal coordinate descent (DP-CD) algorithm to solve problem ($\star'$) under (ϵ, δ)-differential privacy constraints. We first describe our algorithm, show how to parameterize it to satisfy the desired privacy constraint, and prove corresponding utility results. Finally, we compare these utility guarantees with DP-SGD.

4.3.1 Private Proximal Coordinate Descent

Let $D = \{d_1, \dots, d_n\} \in \mathcal{X}^n$ be a dataset. We denote by $f(w) = \frac{1}{n}\sum_{i=1}^n \ell(w; d_i)$ the M-coordinate-smooth part of ($\star'$), by $\psi(w) = \sum_{j=1}^p \psi_j(w_j)$ its separable part, and let $F(w) = f(w) + \psi(w)$. Proximal coordinate descent methods Richtárik and Takáč, 2014 solve problem ($\star'$) through iterative proximal gradient steps along each coordinate of F. Formally, given $w \in \mathbb{R}^p$ and $j \in [p]$, the j-th coordinate of w is updated as follows:

$$w_j^+ = \mathrm{prox}_{\gamma_j \psi_j}\left(w_j - \gamma_j \nabla_j f(w_t)\right) \ , \tag{4.3.1}$$

where $\gamma_j > 0$ is the step size and $\mathrm{prox}_{\gamma_j\psi_j}(w) = \arg\min_{v\in\mathbb{R}^p}\left\{\frac{1}{2}\|v - w\|_2^2 + \gamma_j\psi_j(v)\right\}$ is the proximal operator associated with ψ_j (Parikh and Boyd, 2014).

Algorithm 4.3.1: DP-CD: Differentially Private Proximal Coordinate Gradient Descent.

Input: initial point w^0, noise scales $\sigma_1, \dots, \sigma_p > 0$, step sizes $\gamma_1, \dots, \gamma_p > 0$, number of iteration $T, K > 0$.

For $t = 0$ to $T - 1$:

Set $\theta^0 = w^t$

For $k = 0$ to $K - 1$:

Pick $j \sim \mathcal{U}([p])$

Set $\theta^{k+1} = \theta^k$

Update $\theta_j^{k+1} = \mathrm{prox}_{\gamma_j\psi_j}(\theta_j^k - \gamma_j(\nabla_j f(\theta^k) + \eta_j^t))$, with $\eta_j^t \sim \mathcal{N}(0, \sigma_j^2)$

Set $w^{t+1} = \frac{1}{K}\sum_{k=1}^K \theta^k$

Return: w^T.

Update (4.3.1) only requires the computation of the j-th entry of the gradient. To satisfy differential privacy, we perturb this gradient entry with additive Gaussian noise of variance σ_j^2. The complete `DP-CD` procedure is shown in Algorithm 4.3.1. At each iteration, we pick a coordinate uniformly at random and update according to (4.3.1), albeit with noise addition. For technical reasons related to our analysis, we use a periodic averaging scheme. This scheme is similar to `DP-SVRG` (Johnson and Zhang, 2013), although no variance reduction is required since `DP-CD` computes coordinate gradients over the whole dataset.

4.3.2 Privacy Guarantees

For Algorithm 4.3.1 to satisfy (ϵ, δ)-differential privacy, the noise scales $\sigma = (\sigma_1, \ldots, \sigma_p)$ can be calibrated as given in Theorem 4.3.1.

Theorem 4.3.1. *Assume $\ell(\cdot; d)$ is L-coordinate-Lipschitz $\forall d \in \mathcal{X}$. Let $\epsilon \leq 1$ and $\delta < 1/3$. If $\sigma_j^2 = \frac{12 L_j^2 T K \log(1/\delta)}{n^2 \epsilon^2}$ for all $j \in [p]$, then Algorithm 4.3.1 satisfies (ϵ, δ)-differential privacy.*

Proof Sketch. (Complete proof in Appendix A.2)

We track the privacy loss using Rényi differential privacy. The j-th entry of ∇f has sensitivity $\Delta(\nabla_j f) = \Delta(\nabla_j \ell)/n \leq 2L_j/n$. By the Gaussian mechanism each iteration of `DP-CD` is $(\alpha, \frac{2L_j^2 \alpha}{n^2 \sigma_j^2})$-Rényi differential privacy for all $\alpha > 1$. The composition theorem for Rényi differential privacy gives a global guarantee for `DP-CD`, that we convert to (ϵ, δ)-differential privacy using Proposition 3 of Mironov (2017).

Choosing α carefully finally proves the result. □

The dependence of the noise scales on ϵ, δ, n and TK (the number of updates) in Theorem 4.3.1 is standard in DP-ERM. However, the noise is calibrated to the loss function's *coordinate*-Lipschitz constants. These can be much lower their global counterpart, the latter being used to calibrate the noise in `DP-SGD` algorithms. This will be crucial for `DP-CD` to achieve better utility than `DP-SGD` in some regimes. We also note that, unlike `DP-SGD`, `DP-CD` does not rely on privacy amplification by subsampling (Balle et al., 2018; Mironov et al., 2019), and thereby avoids the approximations required by these schemes to bound the privacy loss.

Remark 4.3.1. *Theorem 4.3.1 assumes $\epsilon \in (0, 1]$ to give a simple closed form for the noise scales. In practice we compute tighter values numerically using Rényi differential privacy formulas directly (see Eq. A.2.5 in Appendix A.2), removing this assumption.*

4.3.3 Utility Guarantees

We now state our central result on the utility of DP-CD for the composite DP-ERM problem. As done in previous work, we use the asymptotic notation $\widetilde{O}$ to hide non-significant logarithmic factors. Non-asymptotic utility bounds can be found in Appendix B.2.

Theorem 4.3.2. *Let $\ell(\cdot; d)$ be a convex and L-coordinate-Lipschitz loss function for all $d \in \mathcal{X}$, and f be convex and M-coordinate-smooth. Let $\psi : \mathbb{R}^p \to \mathbb{R}$ be a convex and separable function. Let $\epsilon \leq 1, \delta < 1/3$ be the privacy budget. Let w^* be a minimizer of F and $F^* = F(w^*)$. Let $w_{priv} \in \mathbb{R}^p$ be the output of Algorithm 4.3.1 with step sizes $\gamma_j = 1/M_j$, and noise scales $\sigma_1, \ldots, \sigma_p$ set as in Theorem 4.3.1 (with T and K chosen below) to ensure (ϵ, δ)-differential privacy. Then, the following holds:*

1. *For F convex, $K = O\left(\frac{R_M \sqrt{p} n \epsilon}{\|L\|_{M^{-1}}}\right)$, and $T = 1$, then:*

$$\mathbb{E}[F(w_{priv}) - F^*] = \widetilde{O}\left(\frac{\sqrt{p \log(1/\delta)}}{n\epsilon} \|L\|_{M^{-1}} R_M\right) ,$$

   where $R_M = \max(\sqrt{F(w^0) - F(w^*)}, \|w^0 - w^*\|_M)$ and more simply $R_M = \|w^0 - w^*\|_M$ when $\psi = 0$.

2. *For F μ_M-strongly convex w.r.t. $\|\cdot\|_M$, $K = O\left(p/\mu_M\right)$, and $T = O\left(\log(\frac{n\epsilon\mu_M}{p\|L\|_{M^{-1}}})\right)$, then:*

$$\mathbb{E}[F(w_{priv}) - F^*] = \widetilde{O}\left(\frac{p \log(1/\delta)}{n^2\epsilon^2} \frac{\|L\|^2_{M^{-1}}}{\mu_M}\right) .$$

Expectations are over the randomness of the algorithm.

Proof Sketch. (Complete proof in Appendix B.2)

Existing analyses of CD fail to track the noise tightly across coordinates when adapted to the private setting. Contrary to these classical analyses, we prove a recursion on $\mathbb{E}\|\theta^k - w^*\|^2_M$, rather than on $\mathbb{E}\big[F(\theta^k) - F(w^*)\big]$. Our key technical result is a descent lemma (Lemma A.3.2) allowing us to obtain

$$\begin{aligned} &\mathbb{E}\big[F(\theta^{k+1}) - F^*\big] - \tfrac{p-1}{p}\mathbb{E}\big[F(\theta^k) - F^*\big] \\ &\quad \leq \mathbb{E}\|\theta^k - w^*\|^2_M - \mathbb{E}\|\theta^{k+1} - w^*\|^2_M + \tfrac{1}{p}\|\sigma\|^2_M \ . \end{aligned} \tag{4.3.2}$$

The above inequality shows that coordinate-wise updates leave a fraction $\frac{p-1}{p}$ of the function "unchanged", while the remaining part decreases (up to additive noise). Importantly, all quantities are measured in M-norm. When summing (4.3.2) for

$k = 0, \dots, K-1$, its left hand side simplifies and its right hand side is simplified as a telescoping sum:

$$\tfrac{1}{p}\sum_{k=1}^{K}\mathbb{E}\big[F(\theta^k) - F^*\big] \leq \mathbb{E}\big[F(w^t) - F^*\big] + \mathbb{E}\|w^t - w^*\|_M^2 + \tfrac{K}{p}\|\sigma\|_{M^{-1}}^2 \ ,$$

where w^t comes from $\theta^0 = w^t$. As $w^{t+1} = \frac{1}{K}\sum_{k=1}^{K}\theta^k$ and F is convex, we have

$$F(w^{t+1}) - F^* \leq \tfrac{1}{K}\sum_{k=1}^{K} F(\theta^k) - F^* \ .$$

This proves the sub-linear convergence (up to an additive noise term) of the inner loop. The result in the convex case follows directly (since $T = 1$, only one inner loop is run). For strongly convex F, it further holds that $\mathbb{E}\|w^t - w^*\|_M^2 \leq \frac{2}{\mu_M}\mathbb{E}[F(w^t) - F(w^*)]$. Replacing in (4.5.1) with large enough K gives

$$\mathbb{E}\big[F(w^{t+1}) - F^*\big] \leq \tfrac{1}{2}\mathbb{E}\big[F(w^t) - F^*\big] + \|\sigma\|_{M^{-1}}^2 \ ,$$

and linear convergence (up to an additive noise term) follows. Finally, K and T are chosen to balance the "optimization" and the "privacy" errors. □

Remark 4.3.2. *Our novel convergence proof of CD is also useful in the non-private setting. In particular, we improve upon known convergence rates for inexact CD methods with additive error (Tappenden et al., 2016), under the additional assumptions that gradients are unbiased. In their formalism, we have $\alpha = 0$ and $\beta = \|\sigma\|_{M^{-1}}^2/p$. With our analysis, the algorithm requires $2pR_M^2/(\xi - p\beta)$ (resp. $4p/\mu_M \log((F(w^0) - F^*)/(\xi - p\beta))$) iterations to achieve expected precision $\xi > p\beta$ when F is convex (resp. μ_M-strongly-convex* w.r.t., $\|\cdot\|_M$*), improving upon Tappenden et al. (2016)'s results by a factor $\sqrt{p\beta/2R_M^2}$ (resp. $\mu_M/2$). See Appendix A.3.3 for details. Moreover, unlike this prior work, our analysis does not require the objective to decrease at each iteration, which is essential to guarantee differential privacy.*

Our utility guarantees stated in Theorem 4.3.2 directly depend on precise coordinate-wise regularity measures of the objective function. In particular, the initial distance to optimal, the strong convexity parameter and the overall sensitivity of the loss function are measured in the norms $\|\cdot\|_M$ and $\|\cdot\|_{M^{-1}}$ (*i.e.,* weighted by coordinate-wise smoothness constants or their inverse). In the remainder of this section, we thoroughly compare our utility results with existing ones for `DP-SGD`. We will show the optimality of our utility guarantees in Section 4.4.

Table 4.1: Utility guarantees for `DP-CD`, `DP-SGD`, and `DP-SVRG` for L-coordinate-Lipschitz, Λ-Lipschitz loss.

	Convex	Strongly-convex
`DP-CD`	$\widetilde{O}\left(\frac{\sqrt{p\log(1/\delta)}}{n\epsilon}\|L\|_{M^{-1}}R_M\right)$	$\widetilde{O}\left(\frac{p\log(1/\delta)}{n^2\epsilon^2}\frac{\|L\|^2_{M^{-1}}}{\mu_M}\right)$
`DP-SGD` `DP-SVRG`	$\widetilde{O}\left(\frac{\sqrt{p\log(1/\delta)}}{n\epsilon}\Lambda R_I\right)$	$\widetilde{O}\left(\frac{p\log(1/\delta)}{n^2\epsilon^2}\frac{\Lambda^2}{\mu_I}\right)$

4.3.4 Comparison with `DP-SGD` and `DP-SVRG`

We now compare `DP-CD` with `DP-SGD` and `DP-SVRG`, for which Bassily et al. (2014b) and Wang et al. (2017) proved utility guarantees. In this section, we assume that the loss function ℓ satisfies the hypotheses of Theorem 4.3.2, and is Λ-Lipschitz. We denote by μ_I the strong convexity parameter of $\ell(\cdot, d)$ *w.r.t.*, $\|\cdot\|_2$ and R_I the equivalent of R_M when M is the identity matrix I. As can be seen from Table 4.1, comparing `DP-CD` and `DP-SGD` boils down to comparing $\|L\|_{M^{-1}}R_M$ with ΛR_I for convex functions and $\|L\|^2_{M^{-1}}/\mu_M$ with Λ^2/μ_I for strongly-convex functions. We compare these terms in two scenarios, depending on the distribution of coordinate-wise smoothness constants. To ease the comparison, we assume that $R_M = \|w^0 - w^*\|_M$ and $R_I = \|w^0 - w^*\|_I$ (which is notably the case when $\psi = 0$), and that F has a unique minimizer w^*.

4.3.4 (a) Balanced Setting

When the smoothness constants M are all equal, we have $\|L\|_{M^{-1}}R_M = \|L\|_2 R_I$, and $\|L\|^2_{M^{-1}}/\mu_M = \|L\|^2_2/\mu_I$. This boils down to comparing $\|L\|_2$ to Λ. As $\Lambda \leq \|L\|_2 \leq \sqrt{p}\Lambda$, `DP-CD` can be up to p times worse than `DP-SGD`. This can only happen when features are extremely correlated, which is generally not the case in machine learning. We show empirically in Section 4.6.2 that, even in balanced regimes, `DP-CD` can still significantly outperform `DP-SGD`.

4.3.4 (b) Unbalanced Setting

More favorable regimes exist when smoothness constants are imbalanced. To illustrate this, consider the case where the first coordinate of the loss function ℓ dominates others. There, $M_{\max} = M_1 \gg M_{\min} = M_j$ and $L_{\max} = L_1 \gg L_{\min} = L_j$ for all $j \neq 1$, so that L_1^2/M_1 dominates the other terms of $\|L\|^2_{M^{-1}}$. This yields $\|L\|^2_{M^{-1}} \approx L_1^2/M_1 \approx \Lambda/M_{\max}$, and $\mu_M = \mu_I M_{\min}$. Moreover, if the first coordinate of w^* is already well estimated by w^0 (which is common for sparse models), then $R_M \approx M_{\min}R_I$. We obtain that $\|L\|_{M^{-1}}R_M \approx \sqrt{M_{\min}/M_{\max}}\Lambda R_I$ for convex losses and $\frac{\|L\|_{M^{-1}}}{\mu_M} \approx \frac{M_{\min}}{M_{\max}}\frac{\Lambda^2}{\mu_I}$

for strongly-convex ones. In both cases, `DP-CD` can perform arbitrarily better than `DP-SGD`, depending on the ratio between the smallest and largest coordinate-wise smoothness constants of the loss function. This is due to the inability of `DP-SGD` to adapt its step size to each coordinate. `DP-CD` thus converges quicker than `DP-SGD` on coordinates with smaller-scale gradients, requiring fewer accesses to the dataset, and in turn less noise addition. We give more details on this comparison in Appendix A.4, and complement it with an empirical evaluation on synthetic and real-world data in Section 4.6.

4.4 Lower Bounds

We now prove a new lower bound on the error achievable for composite DP-ERM with L-coordinate-Lipschitz loss functions. While our proof borrows some ideas from the lower bounds known for constrained DP-ERM with Λ-Lipschitz losses (Bassily et al., 2014b), deriving our lower bounds requires to address a number of specific challenges. First, we cannot use an ℓ_2 norm constraint as in Bassily et al. (2014b) in the design of the worst-case problem instances: we can only rely on *separable* regularizers. Second, imbalanced coordinate-wise Lipschitz constants prevent lower-bounding the distance between an arbitrary point and the solution. This leads us to revisit the construction of a "reidentifiable dataset" from Bun et al. (2014) so that we have L-coordinate-Lipschitzness while the sum of each column is large enough, which is crucial in our proof. The full proof is given in Appendix A.5.

Theorem 4.4.1. *Let $n, p > 0$, $\epsilon > 0$, $\delta = o(\frac{1}{n})$, $L_1, \dots, L_p > 0$, such that for all $\mathcal{J} \subseteq [p]$ of size at least $\lceil \frac{p}{75} \rceil$, $\sum_{j \in \mathcal{J}} L_j^2 = \Omega(\|L\|_2^2)$. Let $\mathcal{X} = \prod_{j=1}^{p} \{\pm L_j\}$ and consider any (ϵ, δ)-differentially private algorithm that outputs w^{priv}. In each of the two following cases there exists a dataset $D \in \mathcal{X}^n$, a L-coordinate-Lipschitz loss $\ell(\cdot, d)$ for all $d \in D$ and a regularizer ψ so that, with F the objective of $(\star')$ minimal at $w^* \in \mathbb{R}^p$:*

1. *If F is convex:*

$$\mathbb{E}\big[F(w^{priv}; D) - F(w^*)\big] = \Omega\left(\frac{\sqrt{p}\|L\|_2\|w^*\|_2}{n\epsilon}\right) .$$

2. *If F is μ_I-strongly-convex* w.r.t., $\|\cdot\|_2$*:*

$$\mathbb{E}\big[F(w^{priv}; D) - F(w^*)\big] = \Omega\left(\frac{p\|L\|_2^2}{\mu_I n^2 \epsilon^2}\right) .$$

We recover the lower bounds of Bassily et al. (2014b) for Λ-Lipschitz losses as a special case of ours by setting $L_1 = \cdots = L_p = \Lambda/\sqrt{p}$. In this case, the loss function

used in our proof is indeed $(\sum_{j=1}^{p} L_j^2)^{1/2} = \Lambda$-Lipschitz. To relate these lower bounds to the performance of `DP-CD`, consider a suboptimal version of our algorithm where the step sizes are set to $\gamma_1 = \cdots = \gamma_p = (\max_j M_j)^{-1}$. In this setting, results from Theorem 4.3.2 still hold, and match the lower bounds from Theorem 4.4.1 up to logarithmic factors. We leave open the question of the optimality of `DP-CD` under the additional hypothesis of smoothness.

We note that the assumption on the sum of the L_j's over a set of indices $\mathcal{J}$ in Theorem 4.4.1 can be eliminated at the cost of an additional factor of $L_{\min}/L_{\max}$ for convex losses and $(L_{\min}/L_{\max})^2$ for strongly-convex losses, making the bound looser. Although the aforementioned assumption may seem solely technical, we conjecture that better utility is possible when a few coordinate-wise Lipschitz constants dominate the others. We discuss this further in Section 5.5.

4.5 `DP-CD` in Practice

We now discuss practical questions related to `DP-CD`. First, we show how to implement coordinate-wise gradient clipping using a single hyperparameter. Second, we explain how to privately estimate the smoothness constants. Finally, we discuss the possibility of standardizing the features and how this relates to estimating smoothness constants for the important problem of fitting generalized linear models.

4.5.1 Coordinate-wise Gradient Clipping

To bound the sensitivity of coordinate-wise gradients, our analysis of Section 4.3 relies on the knowledge of Lipschitz constants for the loss function $\ell(\cdot; d)$ that must hold for all possible data points $d \in \mathcal{X}$. This is classic in the analysis of DP optimization algorithms (see e.g., Bassily et al., 2014b; Wang et al., 2017). In practice however, these Lipschitz constants can be difficult to bound tightly and often give largely pessimistic estimates of sensitivities, thereby making gradients overly noisy. To overcome this problem, the common practice in concrete deployments of `DP-SGD` algorithms is to *clip per-sample gradients* so that their norm does not exceed a fixed threshold parameter $C > 0$ (Abadi et al., 2016a):

$$\operatorname{clip}(\nabla \ell(w), C) = \min\left(1, \frac{C}{\|\nabla \ell(w)\|_2}\right) \nabla \ell(w) \ . \tag{4.5.1}$$

This effectively ensures that the sensitivity $\Delta(\operatorname{clip}(\nabla \ell, C))$ of the clipped gradient is bounded by $2C$.

In `DP-CD`, gradients are released one coordinate at a time and should thus be clipped in a coordinate-wise fashion. Using the same threshold for each coordinate would ruin

the ability of DP-CD to account for imbalance across gradient coordinates, whereas tuning coordinate-wise thresholds as p individual hyperparameters $\{C_j\}_{j=1}^p$ is impractical.

Instead, we leverage the results of Theorem 4.3.2 to adapt them from a single hyperparameter. We first remark that our utility guarantees are invariant to the scale of the matrix M. After rescaling M to $\widetilde{M} = \frac{p}{\mathrm{tr}(M)} M$ so that $\mathrm{tr}(\widetilde{M}) = \mathrm{tr}(I) = p$, as proposed by Richtárik and Takáč (2014), the key quantity $\|L\|_{M^{-1}}$ in our our utility bounds is replaced by $\|L\|_{\widetilde{M}^{-1}}$. This suggests to parameterize the j-th threshold as $C_j = \sqrt{M_j / \mathrm{tr}(M)} C$ for some $C > 0$, ensuring that $\|\{C_j\}_{j=1}^p\|_{\widetilde{M}^{-1}} \leq 2C$. The parameter C thus controls the overall sensitivity, allowing clipped DP-CD to perform p iterations for the same privacy budget as one iteration of clipped DP-SGD.

4.5.2 Private Smoothness Constants

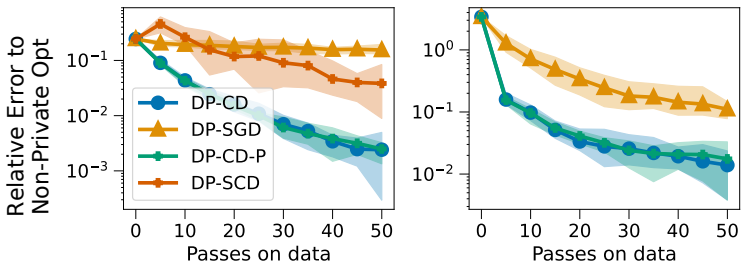
DP-CD requires the knowledge of the coordinate-wise smoothness constants $M_1, \ldots, M_p$ of f to set appropriate step sizes (see Theorem 4.3.2) and clipping thresholds (see above).[2] In most problems, the M_j's depend on the dataset D and must thus be estimated privately using a fraction ϵ' of the overall privacy budget ϵ. Recall that f is the average loss over the dataset D (see the definition of $(\star')$). We can thus denote by $M_j^{(i)}$ the j-th coordinate-smoothness constant of $\ell(\cdot, d_i)$, where d_i is the i-th point in D. The j-th smoothness constant of the function f is thus the average of all these constants: $M_j = \frac{1}{n} \sum_{i=1}^n M_j^{(i)}$.

Assuming that the practitioner knows an approximate upper bound b_j over the $M_j^{(i)}$'s, they can enforce it by clipping $M_j^{(i)}$ to b_j for each $i \in [n]$. The sensitivity of the average of the clipped $M_j^{(i)}$'s is thus $2b_j/n$. One can then compute an estimate of $M_1, \ldots, M_p$ under ϵ-DP using the Laplace mechanism as follows:

$$M_j^{priv} = \frac{1}{n} \sum_{i=1}^n \mathrm{clip}(M_j^{(i)}, b_j) + \mathrm{Lap}\left(\frac{2b_j p}{n\epsilon'}\right), \quad \text{for each } j \in [p] \ , \tag{4.5.2}$$

where the factor p in noise scale comes from using the simple composition theorem Dwork and Roth, 2014, and $\mathrm{Lap}(\lambda)$ is a sample drawn in a Laplace distribution of mean zero and scale λ. The computed constant can then directly be used in DP-CD, allocating the remaining budget $\epsilon - \epsilon'$ to the optimization procedure. We show numerically in Section 4.6 that dedicating 10% of the total budget ϵ to this strategy allows DP-CD to effectively exploit the imbalance across gradients' coordinates.

[2] In fact, only $M_j / \sum_{j'} M_{j'}$ is needed, as we tune the clipping threshold and scaling factor for the step sizes. See Section 4.6.



(a) Electricity Logistic, $\lambda = 1/n$.

(b) California LASSO, $\lambda = 1.0$.

Figure 4.5.1: Relative error to non-private optimal for `DP-CD` (blue), `DP-CD` with privately estimated coordinate-wise smoothness constants (green), `DP-SGD` (orange) and `DP-SCD` (red, only applicable to the smooth case) on two *imbalanced* problems. The number of passes is tuned separately for each algorithm to achieve lowest error. We report min/mean/max values over 10 runs.

4.5.3 Feature Standardization

CD algorithms are very popular to solve generalized linear models (Friedman et al., 2010) and their regularized version (*e.g.,* LASSO, logistic regression). For these problems, the coordinate-wise smoothness constants are $M_j \propto \frac{1}{n}\|X_{:,j}\|_2^2$, where $X_{:,j} \in \mathbb{R}^n$ is the vector containing the value of the j-th feature. Therefore, standardizing the features to have zero mean and unit variance (a standard preprocessing step) makes coordinate-wise smoothness constants equal. However, this requires to compute the mean and variance of each feature in D, which is more costly than the smoothness constants to estimate privately.[3] Moreover, while our theory suggests that `DP-CD` may not be superior to `DP-SGD` when smoothness constants are all equal (see Section 4.3.4), the numerical results of Section 4.6 show that `DP-CD` often outperforms `DP-SGD` even when features are standardized.

Finally, we emphasize that standardization is not always possible. This notably happens when solving the problem at hand is a subroutine of another algorithm. For instance, the Iteratively Reweighted Least Squares (IRLS) algorithm (Holland and Welsch, 1977) finds the maximum likelihood estimate of a generalized linear model by solving a sequence of linear regression problems with reweighted features, proscribing standardization. Similar situations happen when using reweighted ℓ_1 methods for non-convex sparse regression (Candès et al., 2008), relying on convex (LASSO) solvers for the inner loop. `DP-CD` is thus a method of choice to serve as subroutine in private versions of these algorithms.

[3] We note that the privacy cost of standardization is rarely accounted for in practical evaluations.

4.6 Numerical Experiments

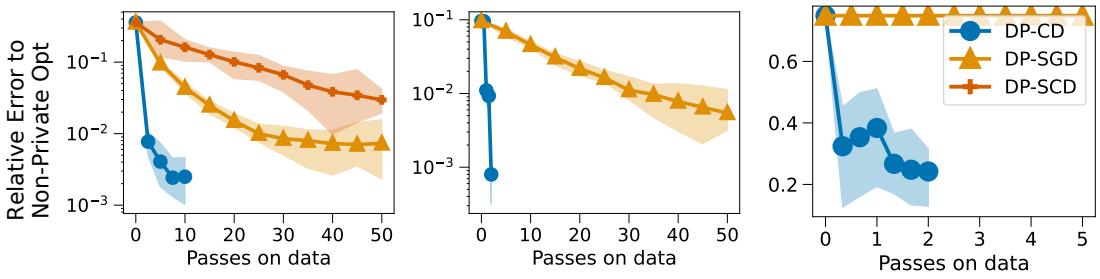
In this section, we assess the practical performance of DP-CD against (proximal) DP-SGD on LASSO[4] and ℓ_2-regularized logistic regression[5]. On the latter problem, we also consider the dual private coordinate descent algorithm of Damaskinos et al. (2021) (DP-SCD). For LASSO, we use the California dataset (Kelley Pace and Barry, 1997), with $n = 20,640$ records and $p = 8$ features as well as a synthetic dataset (coined "Sparse LASSO") with $n = 1,000$ records and $p = 1,000$ independent features that follow a standard normal distribution. The labels are then computed as a noisy sparse linear combination of a subset of 10 active features. For logistic regression, we consider the Electricity dataset (*Electricity Dataset* 2022) with $45,312$ records and 8 features. On California and Electricity, we set $\epsilon = 1$ and $\delta = 1/n^2$, which is generally seen as a rather high privacy regime. The Sparse LASSO dataset corresponds to a challenging setting for privacy ($n = p$), so we consider a low privacy regime with $\epsilon = 10$, $\delta = 1/n^2$. Privacy accounting for DP-SGD is done by numerically evaluating the Rényi DP formula given by the sampled Gaussian mechanism (Mironov et al., 2019). Similarly for DP-CD, we do not use the closed-form formula of Theorem 4.3.1 but rather numerically evaluate the tighter Rényi DP formula given in Appendix A.2.

For DP-SGD, we use constant step sizes and standard gradient clipping. For DP-CD, we adapt the coordinate-wise clipping thresholds from one hyperparameter, as described in Section 4.5.1. Similarly, coordinate-wise step sizes are set to $\gamma_j = \gamma/M_j$, where γ is a hyperparameter. When the coordinate-wise smoothness constants are not all equal, we also consider DP-CD with privately computed M_j's, as described in Section 4.5.2. For each dataset and each algorithm, we simultaneously tune the clipping threshold, the number of passes over the dataset and, for DP-CD and DP-SGD, the step sizes. After tuning these parameters, we report the relative error to the (non-private) optimal objective value. The complete tuning procedure is described in Appendix D.1.1, where we also give the best error for various numbers of passes for each algorithm and dataset. The code used to obtain all our results is available in a public repository[6] and in the supplementary material.

4.6.1 Imbalanced Datasets

In the Electricity and California datasets, features are naturally imbalanced. DP-CD can exploit this through the use of coordinate-wise smoothness constants. We also consider a variant of DP-CD (DP-CD-P) which dedicates 10% of the privacy budget ϵ to estimate these constants (see Section 4.5.2) from a crude upper bound on each feature

[4] *i.e.*, $\ell(w,(x,y)) = (w^\top x - y)^2$, $\psi(w) = \lambda\|w\|_1$.
[5] *i.e.*, $\ell(w,(x,y)) = \log(1+\exp(-yw^\top x))$, $\psi(w) = \frac{\lambda}{2}\|w\|_2^2$.
[6] https://gitlab.inria.fr/pmangold1/private-coordinate-descent/



(a) Electricity Logistic, $\lambda = 1/n$.

(b) California LASSO, $\lambda = 0.1$.

(c) Sparse LASSO, $\lambda = 30$.

Figure 4.6.1: Relative error to non-private optimal for `DP-CD` (blue), `DP-SGD` (orange) and `DP-SCD` (red, only applicable to the smooth case) on three *balanced* problems. The number of passes is tuned separately for each algorithm to achieve lowest error. We report min/mean/max values over 10 runs.

(twice their maximal absolute value). It then uses the resulting private smoothness constants in step sizes and clipping thresholds. Figure 4.5.1 shows that `DP-CD` outperforms `DP-SGD` and `DP-SCD` by an order of magnitude on both datasets, even when the smoothness constants are estimated privately.

4.6.2 Balanced Datasets

To assess the performance of `DP-CD` when coordinate-wise smoothness constants are balanced, we standardize the Electricity and California datasets (see Section 4.5.3). As standardization is done for all algorithms, we do not account for it in the privacy budget. On standardized datasets, coordinate-wise smoothness constants are all equal, removing the need of estimating them privately. We report the results in Figure 4.6.1. Although our theory suggests that `DP-CD` may do worse than `DP-SGD` in balanced regimes, we observe that it still improves over `DP-SGD` (and `DP-SCD`) in practice. Similar observations hold in our challenging Sparse LASSO problem, where `DP-SGD` is barely able to make any progress. We believe these results are in part due to the beneficial effect of clipping in `DP-CD`, and the fact that `DP-SGD` relies on amplification by subsampling, for which privacy accounting is not perfectly tight. Additionally, CD methods are known to perform well on fitting linear models: our results show that this transfers well to private optimization.

4.6.3 Running Time

The results above showed that `DP-CD` yields better utility than `DP-SGD`. We also observe that `DP-CD` tends to reach these results in up to 10 times fewer passes on

the data than DP-SGD (see Appendix D.1.1 for detailed results). Additionally, when accounting for running time, DP-CD significantly outperforms DP-SGD: we refer to Appendix D.1.2 for the counterparts of Figure 4.5.1 and 4.6.1 as a function of the running time instead of the number of passes.

4.7 Conclusion and Discussion

In this Chapter, we presented the first differentially private proximal coordinate descent algorithm for composite DP-ERM. We derived optimal upper bounds on the privacy-utility trade-off achieved by DP-CD. We also proved new lower bounds under a coordinate-Lipschitzness assumption, and showed that DP-CD matches these bounds. Our results demonstrate that DP-CD strongly outperforms DP-SGD when gradients' coordinates are imbalanced, and numerical experiments show that DP-CD can also perform very well in balanced regimes. The choice of coordinate-wise clipping thresholds is crucial for DP-CD to achieve good utility in practice, and we provided a simple rule of thumb for setting them.

Although DP-CD already achieves good utility when most coordinates have small sensitivity, our lower bounds suggest that even better utility could be achieved by dynamically allocating more privacy budget to coordinates with largest sensitivities. A promising direction is to design DP-CD algorithms that leverage active set methods (Yuan et al., 2010; Lewis and Wright, 2016; Nutini et al., 2017; De Santis et al., 2016; Massias et al., 2018), which could provide practical alternatives to recent DP-SGD approaches that use a subspace assumption (Zhou et al., 2021; Kairouz et al., 2021). We also believe that adaptive clipping techniques (Pichapati et al., 2019; Andrew et al., 2021) may help to further improve the practical performance of DP-CD when coordinate-wise smoothness constants are more balanced. Finally, we remark that utility could also be improved further by changing the way coordinates are selected for updates. In the next Chapter, we study another variant of differentially private coordinate descent, where coordinates are chosen using a greedy selection rule.

Chapter 5

Differentially Private Greedy Coordinate Descent

Chapter Abstract

In high dimension, it is common for some model's parameters to carry more information than others. To exploit this, we propose a differentially private greedy coordinate descent (DP-GCD) algorithm. At each iteration, DP-GCD privately performs a coordinate-wise gradient step along the gradients' (approximately) greatest entry. We show theoretically that DP-GCD can achieve a logarithmic dependence on the dimension for a wide range of problems by naturally exploiting their structural properties (such as quasi-sparse solutions). We illustrate this behavior numerically, both on synthetic and real datasets.

This Chapter is mostly based on the paper: *"High-Dimensional Private Empirical Risk Minimization by Greedy Coordinate Descent"* (Mangold, Bellet, Salmon, and Tommasi, 2023a), published at AISTATS 2023.

The code corresponding to this Chapter is available at https://gitlab.inria.fr/pmangold1/greedy-coordinate-descent.

5.1 Introduction

In this Chapter, we propose the differentially private greedy coordinate descent algorithm (DP-GCD). This algorithm extends the GCD algorithm that we presented in Section 2.2.4 to the differentially private setting. Similarly to DP-CD, DP-GCD updates one coordinate at a time, but instead of choosing the coordinate randomly, it chooses

the one with the largest gradient entry. We describe `DP-GCD` and analyze it both theoretically and empirically as a solver of the unconstrained smooth DP-ERM problem. We recall that $\mathcal{X}$ denotes a feature space and $\mathcal{Y}$ a label space, and that we have a dataset $D = \{d_1, \dots, d_n\} \subseteq (\mathcal{X} \times \mathcal{Y})^n$ of n records. We aim at solving the following smooth empirical risk minimization problem:

$$w^* \in \arg\min_{w \in \mathbb{R}^p} \big\{ f(w) \big\} \ , \quad \text{with } f(w) = \frac{1}{n} \sum_{i=1}^{n} \ell(w; d_i) \ , \tag{$\star'$}$$

where p can be large, $\ell : \mathcal{W} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is a loss function which is convex and coordinate-wise smooth in its first parameter.

As we discussed in Chapter 3, solving ($\star'$) using a differentially private algorithm necessarily decreases the utility of the trained model. Specifically, existing lower bounds on utility for a fixed privacy budget (see Section 3.2.4) show that utility decreases polynomially with the dimension p. Since machine learning models are often high-dimensional (*e.g.,* $n \approx p$ or even $n \ll p$), this is a massive drawback for the practical use of differentially private empirical risk minimization. To go beyond this negative result, one can leverage the fact that high-dimensional problems often exhibit some *structure*. In particular, some parameters are typically more significant than others: it is notably (but not only) the case when models are sparse, which is often desired in high dimension (Tibshirani, 1996). Private learning algorithms could thus be designed to exploit this by focusing on the most significant parameters of the problem. Several works have tried to exploit such high-dimensional problems' structure to reduce the dependence on the dimension, *e.g.,* from polynomial to logarithmic. Talwar et al. (2015), Bassily et al. (2021), and Asi et al. (2021) proposed a DP Frank-Wolfe algorithm (`DP-FW`) that exploits the solution's sparsity. However, their algorithm only works on ℓ_1-constrained DP-ERM, restricting its range of application. For sparse linear regression, Kifer et al. (2012) proposed to first identify some support and then solve the DP-ERM problem on the restricted support. Unfortunately, their approach requires implicit knowledge of the solution's sparsity. Finally, Kairouz et al. (2021) and Zhou et al. (2021) used public data to estimate lower-dimensional subspaces, where the gradient can be computed at a reduced privacy cost. A key limitation is that such public data set, from the same domain as the private data, is typically not available in learning scenarios involving sensitive data.

The differentially private greedy coordinate descent algorithm (`DP-GCD`), that we propose in this Chapter, does not have these pitfalls. At each iteration, `DP-GCD` privately determines the gradient's greatest coordinate, and performs a gradient step in its direction. It can thus focus on the most useful parameters, avoiding to waste privacy budget on updating non-significant ones. Formally, we show that `DP-GCD` reduces the dependence on the dimension from polynomial to logarithmic for a wide range of unconstrained problems. This is the first algorithm to obtain such gains without

relying on ℓ_1 or ℓ_0 constraints. In fact, DP-GCD's utility naturally depends on ℓ_1-norm quantities (*i.e.,* distance from initialization to optimal or strong-convexity parameter) and spans two different regimes. When these ℓ_1-norm quantities are $O(1)$ as assumed in DP-FW, DP-GCD attains $O(\log(p)/n^{2/3}\epsilon^{2/3})$ and $O(\log(p)/n^2\epsilon^2)$ utility on convex and strongly-convex problems respectively, outperforming existing DP-FW algorithms without solving a constrained problem. In the second regime, when the ℓ_2-norm counterpart of the above quantities are $O(1)$ as assumed for DP-SGD and its variants, we show that DP-GCD adapts to the problem's underlying structure. Specifically, it is able to *interpolate between logarithmic and polynomial dependence on the dimension.* In addition to these general utility results, we prove that for strongly convex problems with quasi-sparse solutions (including but not limited to sparse problems), DP-GCD converges to a good approximate solution in few iterations. This improves utility in the ℓ_2-norm setting, replacing the polynomial dependence on the ambient space's dimension by the quasi-sparsity level of the solution. We evaluate both our algorithms numerically on real and synthetic datasets, validating our theoretical observations.

The contributions of this Chapter can be summarized as follows:

1. We propose differentially private greedy coordinate descent (DP-GCD), a method that performs updates along the (approximately) greatest entry of the gradient. We formally establish its privacy guarantees, and derive high probability utility upper bounds.

2. We prove that DP-GCD exploits structural properties of the problem (*e.g.,* quasi-sparse solutions) to improve utility. Importantly, DP-GCD does not require prior knowledge of this structure to exploit it.

3. We empirically validate our theoretical results on a variety of synthetic and real datasets, showing that DP-GCD outperforms existing private algorithms on high-dimensional problems with quasi-sparse solutions.

The rest of the Chapter is organized as follows. First, we discuss related work in more details in Section 5.2. Section 5.3 then introduces DP-GCD, and formally analyzes its privacy and utility. We validate our theoretical results numerically in Section 5.4. Finally, we conclude and discuss the limitations of our results in Section 5.5.

5.2 Related Work

Differentially Private Machine Learning in High Dimension. Several approaches have been explored to reduce the dependence on the dimension. One option is to consider ℓ_1-constrained problems. For DP-ERM, Talwar et al. (2015) and Talwar et al. (2016) used a differentially private Frank-Wolfe algorithm (DP-FW) (Frank

and Wolfe, 1956; Jaggi, 2013) to achieve utility that scales logarithmically with the dimension. Asi et al. (2021) and Bassily et al. (2021) proposed stochastic `DP-FW` algorithms. For more general domains (*e.g.,* polytopes), Kasiviswanathan and Jin (2016) randomly project the data on a smaller-dimensional space, and lift the result back onto the original space. The dependence in the dimension is encoded by the Gaussian width of the domain, again leading to $O(\log p)$ error for the ℓ_1 ball or the simplex. Wang et al. (2017) derived a faster mirror descent algorithm for DP-ERM whose utility also depends on the Gaussian width of the domain. Our approach matches the $O(\log p)$ dependence of the above methods when key quantities are bounded in ℓ_1 norm, but can also achieve such gains for more general problems, *e.g.,* when the problem has a quasi-sparse solution. Kifer et al. (2012) previously leveraged the solution sparsity for the specific problem of sparse linear regression: they first identify some support, and then solve DP-ERM on this restricted support. Similarly, Wang and Gu (2019) and Hu et al. (2022b) proposed hard thresholding-based algorithms for DP-ERM under sparsity (ℓ_0 norm) constraints. Both approaches achieve an error of $O(\log p)$ but rely either on prior knowledge on the solution's sparsity, or on the tuning of an additional hyperparameter. In contrast, our approach automatically adapts to the sparsity and works also when solutions are only quasi-sparse. Finally, Kairouz et al. (2021) and Zhou et al. (2021) estimate lower-dimensional gradient subspaces using public data. This reduces noise addition, but in practice, public data is only rarely available.

Private Coordinate Descent. In the previous Chapter, we proposed differentially private coordinate descent (`DP-CD`), analyzed its utility and derived corresponding lower bounds. We showed that `DP-CD` can exploit coordinate-wise regularity assumptions to use larger step-sizes, outperforming `DP-SGD` when gradient coordinates are imbalanced. Interestingly, `DP-GCD` also shares this property. Other work on differentially private coordinate descent, that we already discussed in Section 4.2, all rely on random selection of the updated coordinate. This rule fails to exploit key problem's properties such as sparsity of the solution, and thus suffer a polynomial dependence on the dimension p. In contrast, our private greedy selection rule focuses on the most useful coordinates, thereby reducing the dependence on p to only logarithmic in such settings.

5.3 Private Greedy Coordinate Descent

In this section, we present the contribution of this Chapter: the differentially private greedy coordinate descent algorithm (`DP-GCD`). As described in Section 5.3.1, `DP-GCD` updates only one coordinate per iteration, which is selected greedily as the (approximately) largest entry of the gradient so as to maximize the improvement in utility at

each iteration. We establish privacy (Section 6.5) and utility (Section 5.3.3) guarantees for `DP-GCD`. We further show in Section 5.3.4 that `DP-GCD` enjoys improved utility for high-dimensional problems with a *quasi-sparse* solution (*i.e.,* with a fraction of the parameters dominating the others). We then provide a proximal extension of `DP-GCD` to non-smooth problems (Section 5.3.5) and conclude with a discussion of `DP-GCD`'s computational complexity in Section 5.3.6.

5.3.1 The Algorithm

At each iteration, `DP-GCD` (Algorithm 5.3.1) updates the parameter with the greatest gradient value (rescaled by the inverse square root of the coordinate-wise smoothness constant). This corresponds to the Gauss-Southwell-Lipschitz rule (Nutini et al., 2015). We describe this algorithm in Algorithm 5.3.1.

Algorithm 5.3.1: `DP-GCD`: Differentially Private Greedy Coordinate Descent.

Input: initial point w^0, noise scales $\lambda_1, \dots, \lambda_p, > 0$, $\lambda'_1, \dots, \lambda'_p, > 0$, step sizes $\gamma_1, \dots, \gamma_p > 0$, number of iteration $T > 0$.

For $t = 0$ to $T - 1$:

Select $j = \underset{j' \in [p]}{\arg\max} \frac{|\nabla_{j'} f(w^t) + \chi^t_{j'}|}{\sqrt{M_{j'}}}$, with $\chi^t_{j'} \sim \text{Lap}(\lambda'_{j'})$

Set $w^{t+1} = w^t$

Update $w^{t+1}_j = w^t_j - \gamma_j(\nabla_j f(w^t) + \eta^t_j)$, with $\eta^t_j \sim \text{Lap}(\lambda_j)$

Return: w^T.

To guarantee privacy, this selection is done using the report-noisy-max mechanism (Dwork and Roth, 2014) with noise scales λ'_j along j-th entry ($j \in [p]$). `DP-GCD` then performs a gradient step with step size $\gamma_j > 0$ along this direction. The gradient is privatized using the Laplace mechanism (Dwork and Roth, 2014) with scale λ_j.

Remark 5.3.1 (Sparsity of iterates)**.** *When initialized at $w^0 = 0$, `DP-GCD` generates sparse iterates. Since it chooses its updates greedily, this gives a screening ability to the algorithm (Fang et al., 2020). We discuss the implications of this property in Section 5.3.4, where we show that `DP-GCD`'s utility is improved when the problem's solution is (quasi-)sparse.*

5.3.2 Privacy Guarantees

The privacy guarantees of DP-GCD depends on the noise scales λ_j and λ'_j. In Theorem 5.3.1, we describe how to set these values so as to ensure that DP-GCD is (ϵ, δ)-differentially private.

Theorem 5.3.1. *Let $\epsilon, \delta \in (0, 1]$. Algorithm 5.3.1 with $\lambda_j = \lambda'_j = \frac{8L_j}{n\epsilon}\sqrt{T\log(1/\delta)}$ is (ϵ, δ)-differentially private.*

> *Proof Sketch.* (Detailed proof in Appendix B.1) Let $\epsilon' = \epsilon/\sqrt{16T\log(1/\delta)}$. At an iteration t, data is accessed twice. First, to compute the index j of the coordinate to update. It is obtained as the index of the largest noisy entry of f's gradient, with noise $\text{Lap}(\lambda'_j)$. By the report-noisy-argmax mechanism, j is ϵ'-DP. Second, to compute the gradient's j's entry, which is released with noise $\text{Lap}(\lambda_j)$.The Laplace mechanism ensures that this computation is also ϵ'-DP. Algorithm 5.3.1 is thus the $2T$-fold composition of ϵ'-DP mechanisms, and the result follows from DP's advanced composition theorem (Dwork and Roth, 2014). □

Remark 5.3.2. *The assumption $\epsilon \in (0, 1]$ is only used to give a closed-form expression for the noise scales λ, λ's. In practice, we tune them numerically to obtain the desired value of $\epsilon > 0$ by the advanced composition theorem (see eq. (B.1.1) in Appendix B.1), removing the assumption $\epsilon \leq 1$.*

Computing the greedy update requires injecting Laplace noise that scales with the coordinate-wise Lipschitz constants $L_1, \ldots, L_p$ of the loss. These constants are typically smaller than their global counterpart. This allows DP-GCD to inject less noise on smaller-scaled coordinates.

5.3.3 Utility Guarantees

We now prove utility upper bounds for DP-GCD. We show that in favorable settings (see discussion below), DP-GCD decreases the dependence on the dimension from polynomial to logarithmic. Theorem 5.3.2 gives asymptotic utility upper bounds, where $\widetilde{O}$ ignores non-significant logarithmic terms. Complete non-asymptotic results can be found in Appendix B.2.

Theorem 5.3.2. *Let $\epsilon, \delta \in (0, 1]$. Assume $\ell(\cdot; d)$ is a convex and L-coordinate-Lipschitz loss function for all $d \in \mathcal{X}$, and f is M-coordinate-smooth. Define $\mathcal{W}^*$ the set of minimizers of f, and f^* the minimum of f. Let $w_{priv} \in \mathbb{R}^p$ be the output of Algorithm 5.3.1 with step sizes $\gamma_j = 1/M_j$, and noise scales $\lambda_1, \ldots, \lambda_p$, $\lambda'_1, \ldots, \lambda'_p$ set as in Theorem 5.3.1 (with T chosen below) to ensure (ϵ, δ)-DP. Then, the following holds for any $\zeta \in (0, 1]$:*

1. *When f is convex, let* $R_{M,1} = \max_{w\in\mathbb{R}^p} \max_{w^*\in\mathcal{W}^*}\{\|w - w^*\|_{M,1} \,|\, f(w) \leq f(w^0)\}$. *Assume the initial optimality gap is* $f(w^0) - f^* \geq \frac{16L_{\max}\sqrt{T\log(1/\delta)\log(2Tp/\zeta)}}{M_{\min} n\epsilon}$, *and set* $T = O(n^{2/3}\epsilon^{2/3}R_{M,1}^{2/3}M_{\min}^{1/3}/L_{\max}^{2/3}\log(1/\delta)^{1/3})$. *Then with probability at least* $1-\zeta$,

$$f(w_{priv}) - f^* = \widetilde{O}\left(\frac{R_{M,1}^{4/3}L_{\max}^{2/3}\log(1/\delta)\log(p/\zeta)}{n^{2/3}\epsilon^{2/3}M_{\min}^{1/3}}\right) .$$

2. *When f is $\mu_{M,1}$-strongly convex w.r.t. $\|\cdot\|_{M,1}$, set the number of iterations to* $T = O\left(\frac{1}{\mu_{M,1}}\log(\frac{M_{\min}\mu_{M,1}n\epsilon(f(w^0)-f(w^*))}{L_{\max}\log(1/\delta)\log(2p/\zeta)})\right)$. *Then with probability at least* $1-\zeta$,

$$f(w_{priv}) - f^* = \widetilde{O}\left(\frac{L_{\max}^2\log(1/\delta)\log(2p/\mu_M\zeta)}{M_{\min}\mu_{M,1}^2 n^2\epsilon^2}\right) .$$

Proof Sketch. (Detailed proof in Appendix B.2). We start by proving a noisy "descent lemma". Since f is smooth, we have $f(w^{t+1}) \leq f(w^t) - \frac{1}{2M_j}\nabla_j f(w^t)^2 + \frac{1}{2M_j}(\eta_j^t)^2$. The greedy selection of j gives that $-\frac{1}{M_j}(\nabla_j f(w^t)+\chi_j)^2 \leq -\|\nabla f(w^t)+\chi\|_{M^{-1},\infty}^2$. We then use the inequality $(a+b)^2 \leq 2a^2+2b^2$ for $a, b \in \mathbb{R}$, and convexity arguments to prove the lemma. When f is convex, we have

$$\begin{aligned} f(w^{t+1}) - f(w^*) &\leq f(w^t) - f(w^*) \\ &- \frac{(f(w^t)-f(w^*))^2}{8\|w^t - w^*\|_{M,1}^2} + \frac{|\eta_j^t|^2}{2M_j} + \frac{|\chi_j^t|^2}{2M_j} + \frac{|\chi_{j^*}^t|^2}{4M_{j^*}} . \end{aligned}$$

There, we observe that, at each iteration, either (i) w^t is far enough from the optimum, and the value of the objective decreases with high probability, either (ii) w^t is close to the optimum, then all future iterates remain in a ball whose radius depends on the scale of the noise. We prove this key property rigorously in Appendix B.2.3 (b).

When f is $\mu_{M,1}$-strongly-convex *w.r.t.*, $\|\cdot\|_{M,1}$, we obtain

$$\begin{aligned} f(w^{t+1}) - f(w^*) \leq \left(1 - \frac{\mu_{M,1}}{4}\right)(f(w^t) - f(w^*)) \\ + \frac{|\eta_j^t|^2}{2M_j} + \frac{|\chi_j^t|^2}{2M_j} + \frac{|\chi_{j^*}^t|^2}{4M_{j^*}} , \end{aligned}$$

and the result follows by induction. In both settings, we use Chernoff bounds to obtain a high-probability result. □

Remark 5.3.3. *The lower bound on $f(w^0) - f^*$ in Theorem 5.3.2 is a standard assumption in the analysis of inexact coordinate descent methods: it ensures that*

sufficient decrease is possible despite the noise. A similar assumption is made by Tappenden et al. (2016), see Theorem 5.1 therein.

Discussion of the utility bounds One of the key properties of DP-GCD is that its utility is dictated by ℓ_1-norm quantities ($R_{M,1}$ and $\mu_{M,1}$). Remarkably, this arises without enforcing any ℓ_1 constraint in the problem, which is in stark contrast with private Frank-Wolfe algorithms (DP-FW) that require such constraints (Talwar et al., 2015; Asi et al., 2021; Bassily et al., 2021). To better grasp the implications of this, we discuss our results in two regimes considered in previous work (see Section 5.2): (i) when these ℓ_1-norm quantities are bounded (similarly to DP-FW algorithms), and (ii) when their ℓ_2-norm counterparts are bounded (similarly to DP-SGD-style algorithms).

Bounded in ℓ_1-norm. When $R_{M,1}$ and $\mu_{M,1}$ are $O(1)$, as assumed in prior work on DP-FW (Talwar et al., 2015; Asi et al., 2021; Bassily et al., 2021), DP-GCD's dependence on the dimension is *logarithmic*. For convex objectives, its utility is $O(\log(p)/n^{2/3}\epsilon^{2/3})$, matching that of DP-FW and known lower bounds (Talwar et al., 2015). For strongly-convex problems, DP-GCD is the first algorithm to achieve a $O(\log(p)/n^2\epsilon^2)$ utility. Indeed, the only competing result in this setting, due to Asi et al. (2021), obtains a worse utility of $O(\log(p)^{4/3}/n^{4/3}\epsilon^{4/3})$ by using an impractical reduction of DP-FW to the convex case. DP-GCD outperforms this prior result without suffering the extra complexity due to the reduction.

Bounded in ℓ_2-norm. Consider $R_{M,2}$ and $\mu_{M,2}$, the ℓ_2-norm counterparts of $R_{M,1}$ and $\mu_{M,1}$. Assume that $R_{M,2}$ and $\mu_{M,2}$ are both $O(1)$, as considered in DP-SGD and its variants (Bassily et al., 2014a; Wang et al., 2017). We compare these quantities using the following inequalities (see Stich et al., 2017; Nutini et al., 2015):

$$R_{M,2}^2 \leq R_{M,1}^2 \leq pR_{M,2}^2 \ , \qquad \tfrac{1}{p}\mu_{M,2} \leq \mu_{M,1} \leq \mu_{M,2} \ .$$

In the best case of these inequalities, the $O(\log p)$ utility bounds of the bounded ℓ_1 norm regime are preserved in the bounded ℓ_2 scenario. In the worst case, the utility of DP-GCD becomes $\widetilde{O}(p^{2/3}/n^{2/3}\epsilon^{2/3})$ and $\widetilde{O}(p^2/n^2\epsilon^2)$ for convex and strongly-convex objectives respectively. These worst-case results match DP-FW's utility in the convex setting (see *e.g.,* Asi et al. (2021)), but they do not match DP-SGD's utility. However, this sheds light on an interesting phenomenon: DP-GCD *interpolates between ℓ_1- and ℓ_2-norm regimes.* Indeed, it lies somewhere between the two extreme cases we just described, depending on how the ℓ_1- and ℓ_2-norm constants compare. Most interestingly, it does so without *a priori* knowledge of the problem or explicit constraint on the domain. Whether there exists an algorithm that yields optimal utility in all regimes is an interesting open question.

Coordinate-wise regularity Due to its use of coordinate-wise step sizes, DP-GCD can adapt to coordinate-wise imbalance of the objective in the same way as its

randomized counterpart, DP-CD, where coordinates are chosen uniformly at random (Mangold et al., 2021). This adaptivity notably appears in Theorem 5.3.2 through the measurement of $R_{M,1}$ and $\mu_{M,1}$ relatively to the scaled norm $\|\cdot\|_{M,1}$ (as defined in Section 6.3). We refer to (Mangold et al., 2021) for detailed discussion of these quantities and the associated gains compared to full gradient methods like DP-SGD.

5.3.4 Better Utility on Quasi-Sparse Problems

In addition to the general utility results presented above, we now exhibit a specific setting where DP-GCD performs especially well, namely strongly-convex problems whose solutions are dominated by a few parameters. We call such vectors quasi-sparse.

Definition 5.3.1 ((α,τ)-quasi-sparsity)**.** *A vector $w \in \mathbb{R}^p$ is (α,τ)-quasi-sparse if it has at most τ entries superior to α (in modulus). When $\alpha = 0$, the vector is called τ-sparse.*

Note that any vector in $\mathbb{R}^p$ is $(0,p)$-quasi-sparse, and for any τ there exists $\alpha > 0$ such that the vector is (α,τ)-quasi-sparse. In fact, α and τ are linked, and $\tau(\alpha)$ can be seen as a function of α. Of course, quasi-sparsity will only yield meaningful improvements when α and τ are small simultaneously.

We now state the main result of this section, which shows that DP-GCD (initialized with $w^0 = 0$) converges to a good approximate solution in few iterations for problems with quasi-sparse solutions.

Theorem 5.3.3 (Proof in Appendix B.2.4 (c))**.** *Consider f satisfying the hypotheses of Theorem 5.3.2, with Algorithm 5.3.1 initialized at $w^0 = 0$. We denote its output w^T, and assume that its iterates remain s-sparse for some $s \leq p$. Assume that f is $\mu_{M,2}$-strongly-convex* w.r.t., *$\|\cdot\|_{M,2}$, and that the (unique) solution of problem ($\star'$) is (α,τ)-quasi-sparse for some $\alpha,\tau \geq 0$. Let $0 \leq T \leq p - \tau$ and $\zeta \in [0,1]$. Then with probability at least $1-\zeta$:*

$$f(w^T) - f^* \leq \prod_{t=1}^{T}\Big(1 - \tfrac{\mu_{M,2}}{4(\tau+\min(t,s))}\Big)(f(w^0) - f^*) + \widetilde{O}\Big((T+\tau)(p-\tau)\alpha^2 + \tfrac{L_{\max}^2 T(T+\tau)}{M_{\min}\mu_{M,2}n^2\epsilon^2}\Big) \ .$$

We assume that $\alpha^2 = O\left(L_{\max}^2(s+\tau)/M_{\min}\mu_{M,2}^2 p n^2\epsilon^2\right)$, and set the number of iterations to $T = \frac{s+\tau}{\mu_{M,2}}\log((f(w^0)-f^*)M_{\min}\mu_{M,2}n^2\epsilon^2/L^2)$. Then, we have that, with probability at least $1-\zeta$,

$$f(w^T) - f^* = \widetilde{O}\left(\frac{L_{\max}^2}{M_{\min}}\frac{(s+\tau)^2\log(2p/\zeta)}{\mu_{M,2}n^2\epsilon^2}\right) \ .$$

Here, strong convexity is measured in ℓ_2 norm but the dependence on the dimension is reduced from p, the ambient space dimension, to $(s+\tau)^2$, the *effective dimension of the space where the optimization actually takes place*. For high-dimensional sparse problems, the latter is typically much smaller and yields a large improvement in utility. Note that it is not necessary for the solution to be perfectly sparse: it suffices that most of its mass is concentrated in a fraction of the coordinates. Notably, when $\alpha^2 = O(L_{\max}^2 T / M_{\min} \mu_{M,2} p n^2 \epsilon^2)$, the lack of sparsity is smaller than the noise, and does not affect the rate. It generalizes the results by Fang et al. (2020) for non-private and sparse settings, that we recover when $\alpha = 0$ and $\epsilon \to +\infty$.

In practice, the assumption over the iterates' sparsity is often met with $s \ll p$. In the non-private setting, greedy coordinate descent is known to focus on coordinates that are non-zero in the solution (Massias et al., 2017): this keeps iterates' sparsity close to the one of the solution. Furthermore, due to privacy constraints, `DP-GCD` will often run for $T \ll p$ iterations. This is especially true in high-dimensional problems, where the amount of noise required to guarantee privacy does not allow many iterations (*cf.* experiments in Section 5.4).

5.3.5 Proximal `DP-GCD`

In Section 5.3.4, we proved that `DP-GCD`'s utility is improved when problem's solution is (quasi-)sparse. This motivates us to consider problems with sparsity-inducing regularization (*i.e.,* when $\psi \neq 0$ in $(\star')$), such as the ℓ_1 norm of w (Tibshirani, 1996). To tackle such non-smooth terms, we propose a proximal version of `DP-GCD` (for which the same privacy guarantees hold), building upon the multiple greedy rules that have been proposed for the nonsmooth setting (see *e.g.,* Tseng and Yun, 2009; Nutini et al., 2015). We describe this algorithm in Algorithm 5.3.2.

Algorithm 5.3.2: Proximal `DP-GCD`: Differentially Private Proximal Greedy Coordinate Descent.

Input: initial point w^0, noise scales $\lambda_1, \dots, \lambda_p, > 0$, $\lambda'_1, \dots, \lambda'_p, > 0$, step sizes $\gamma_1, \dots, \gamma_p > 0$, number of iteration $T > 0$.

For $t = 0$ to $T - 1$:

Select j by the noisy `GS-s`, `GS-r` or `GS-q` rule with noise scales $\lambda'_1, \dots \lambda'_p$

Set $w^{t+1} = w^t$

Update $w_j^{t+1} = \operatorname{prox} \gamma_j \psi_j (w^t - \gamma_j (\nabla_j f(w^t) + \eta_j^t))$, with $\eta_j^t \sim \operatorname{Lap}(\lambda_j)$

Return: w^T.

The same privacy guarantees as for the smooth `DP-GCD` algorithm hold since, privacy-wise, the proximal step is a post-processing step. We also adapt the greedy selection rule to incorporate the non-smooth term. We can use one of the following three rules

$$j = \arg\max_{j \in [p]} \min_{\xi_j \in \partial \psi_j(w_j)} \frac{1}{\sqrt{M_j}} |\nabla_j f(w^t) + \eta_j^t + \xi_j| \ , \tag{GS-s}$$

$$j = \arg\max_{j \in [p]} \sqrt{M_j} |\operatorname{prox}_{\frac{1}{M_j}\psi_j}(w_j^t - \frac{1}{M_j}(\nabla_j f(w^t) + \eta_j^t) - w_j^t| \ , \tag{GS-r}$$

$$j = \arg\max_{j \in [p]} \min_{\alpha \in \mathbb{R}} \nabla_j f(w^t)\alpha + \frac{M_j}{2}\alpha^2 + \psi_j(w_j^t + \alpha) - \psi_j(w_j^t) \ . \tag{GS-q}$$

These rules are commonly considered in the non-private GCD literature (see *e.g.,* Tseng and Yun, 2009; Shi et al., 2017; Karimireddy et al., 2019), except for the noise η_j^t and the rescaling in the `GS-s` and `GS-r` rules.

5.3.6 Computational Cost

Each iteration of `DP-GCD` requires computing a full gradient, but only uses one of its coordinates. In non-private optimization, one would generally be better off performing the full update to avoid wasting computation. This is not the case when gradients are private. Indeed, using the full gradient requires privatizing p coordinates, even when only a few of them may be needed. Conversely, the report noisy max mechanism (Dwork and Roth, 2014) allows to select these entries *without paying the full privacy cost of dimension*. Hence, the greedy updates of `DP-GCD` reduce the noise needed at the cost of more computation.

In practice, the higher computational cost of each iteration may not always translate in a significantly larger cost overall: as shown by our theoretical results, `DP-GCD` is able to exploit the *quasi-sparsity* of the solution to progress fast and only a handful of iterations may be needed to reach a good private solution. In contrast, most updates of classic private optimization algorithms (like `DP-SGD`) may not be worth doing, and lead to unnecessary injection of noise. We illustrate this phenomenon numerically in Section 5.4.

5.4 Experiments

In this section, we evaluate the practical performance of `DP-GCD` on linear models using the logistic and squared loss with ℓ_1 and ℓ_2 regularization. We compare `DP-GCD` to two competitors: differentially private stochastic gradient descent (`DP-SGD`) with batch size 1 (Bassily et al., 2014a; Abadi et al., 2016a), and differentially private

Table 5.1: Number of records and features in each dataset.

	`log1`, `log2`	`square`	`mtp`	`dorothea`	`california`	`madelon`
Records	1,000	1,000	4,450	800	20,640	2,600
Features	100	1,000	202	88,119	8	501

randomized coordinate descent (`DP-CD`) (Mangold et al., 2021). The code is available online[1] and in the supplementary.

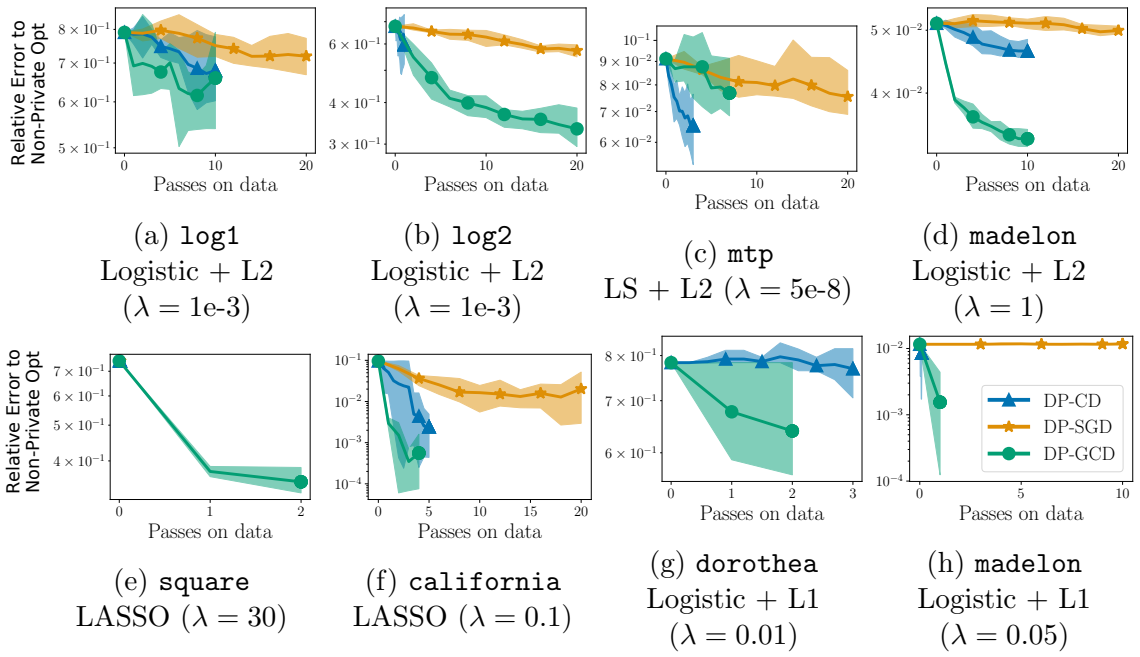
Datasets. The first two datasets, coined `log1` and `log2`, are synthetic. We generate a design matrix $X \in \mathbb{R}^{1,000\times 100}$ with unit-variance, normally-distributed columns. Labels are computed as $y = Xw^{(true)} + \varepsilon$, where ε is normally-distributed noise and $w^{(true)}$ is drawn from a log-normal distribution of parameters $\mu = 0$ and $\sigma = 1$ or 2 respectively. This makes $w^{(true)}$ quasi-sparse. The `square` dataset is generated similarly, with $X \in \mathbb{R}^{1,000\times 1,000}$ and $w^{(true)}$ having only 10 non-zero values. The `california` dataset can be downloaded from `scikit-learn` (Pedregosa et al., 2011) while `mtp`, `madelon` and `dorothea` are available in the `OpenML` repository (Vanschoren et al., 2014); see summary in Table 5.1. We discuss the levels of (quasi)-sparsity of each problem's solution in Appendix D.2.

Algorithmic setup. *(Privacy.)* For each algorithm, the tightest noise scales are computed numerically to guarantee a suitable privacy level of $(1, 1/n^2)$-DP, where n is the number of records in the dataset. For `DP-CD` and `DP-SGD`, we privatize the gradients with the Gaussian mechanism (Dwork and Roth, 2014), and account for privacy tightly using Rényi differential privacy (RDP) (Mironov, 2017). For `DP-SGD`, we use RDP amplification for the subsampled Gaussian mechanism (Mironov et al., 2019).

(Hyperparameters.) For `DP-SGD`, we use constant step sizes and standard gradient clipping (Abadi et al., 2016a). For `DP-GCD` and `DP-CD`, we set the step sizes to $\eta_j = \frac{\gamma}{M_j}$, and adapt the coordinate-wise clipping thresholds from one hyperparameter, as proposed by Mangold et al. (2021). For each algorithm, we thus tune two hyperparameters: one step-size and one clipping threshold; see also Appendix D.2.

(Plots.) In all experiments, we plot the relative error to the *non-private* optimal objective value for the best set of hyperparameters (averaged over 5 runs), as a function of the number of passes on the data. Each pass corresponds to p iterations of `DP-CD`, n iterations of `DP-SGD` and 1 iteration of `DP-GCD`. This guarantees the same amount of computation for each algorithm, for each x-axis tick.

[1] https://gitlab.inria.fr/pmangold1/greedy-coordinate-descent



(a) `log1` Logistic + L2 (λ = 1e-3)
(b) `log2` Logistic + L2 (λ = 1e-3)
(c) `mtp` LS + L2 (λ = 5e-8)
(d) `madelon` Logistic + L2 (λ = 1)
(e) `square` LASSO (λ = 30)
(f) `california` LASSO (λ = 0.1)
(g) `dorothea` Logistic + L1 (λ = 0.01)
(h) `madelon` Logistic + L1 (λ = 0.05)

Figure 5.4.1: Relative error (min/mean/max over 5 runs) to non-private optimal for `DP-GCD` (our approach) versus `DP-CD` and `DP-SGD`. On the x-axis, 1 tick represents a full access to the data: p iterations of `DP-CD`, n iterations of `DP-SGD` and 1 iteration of `DP-GCD`. Number of iterations, clipping thresholds and step sizes are tuned simultaneously for each algorithm.

`DP-GCD` exploits problem structure. In the higher-dimensional datasets `square` and `dorothea`, where $p \geq n$, `DP-GCD` is the only algorithm that manages to do multiple iterations and to decrease the objective value (see Figures 5.4.1e and 5.4.1g). In both problems, solutions are sparse due to the ℓ_1 regularization. This shows that `DP-GCD`'s greedy selection of updates can exploit this property to find relevant non-zero coefficients (see Table D.4 in Appendix D.2), even when this selection is noisy. The lower-dimensional datasets `log1`, `log2` and `madelon` (where $p < n$) are still too high dimensional (relatively to n) for `DP-SGD` and `DP-CD` to make significant progress. In contrast, `DP-GCD` exploits the fact that solutions are quasi-sparse to find good approximate solutions quickly (see Figures 5.4.1a, 5.4.1b, 5.4.1d, 5.4.1e, 5.4.1g and 5.4.1h). On the low-dimensional dataset `california`, `DP-GCD` is roughly on par with `DP-SGD` and `DP-CD` (see Figure 5.4.1f). This is due to the additional noise term introduced by the greedy selection rule: in such setting, the lower number of iterations does not compensate for this as much as in higher-dimensional problems. A similar phenomenon arise in `mtp` (Figure 5.4.1c), whose solution is not imbalanced enough for `DP-GCD` to be superior to its competitors.

Computational complexity. As discussed in Section 5.3.6, one iteration of `DP-GCD` requires a full pass on the data. This is as costly as p iterations of `DP-CD` or n iterations of `DP-SGD`. Nonetheless, on many problems, `DP-GCD` requires just as many passes on the data as `DP-CD` and `DP-SGD` (Figures 5.4.1a and 5.4.1c to 5.4.1f). When more computation is required, it also provides significantly better solutions than `DP-CD` and `DP-SGD` (Figure 5.4.1b). This is in line with our theoretical results from Section 5.3.4.

5.5 Conclusion and Discussion

We proposed `DP-GCD`, a greedy coordinate descent algorithm for DP-ERM. In favorable settings, `DP-GCD` achieves utility guarantees of $O(\frac{\log(p)}{n^{2/3}\epsilon^{2/3}})$ and $O(\frac{\log(p)}{n^2\epsilon^2})$ for convex and strongly-convex objectives. It is the first algorithm to achieve such rates without solving an ℓ_1-constrained problem. Instead, we show that `DP-GCD` depends on ℓ_1-norm quantities and automatically adapts to the structure of the problem. Specifically, `DP-GCD` interpolates between logarithmic and polynomial dependence on the dimension, depending on the problem. Thus, `DP-GCD` constitutes a step towards the design of an algorithm that adjusts to the appropriate ℓ_p structure of a problem (see Bassily et al., 2021; Asi et al., 2021).

We also showed that `DP-GCD` adapts to the quasi-sparsity of the problem, without requiring *a priori* knowledge about it. In such problems, it converges to a good approximate solution in few iterations. This improves utility, and reduces the polynomial dependence on the dimension to a polynomial dependence on the (much smaller) quasi-sparsity level of the solution.

We also proposed and evaluated a proximal variant of `DP-GCD`, allowing non-smooth, sparsity-inducing regularization. While it is not covered by our utility guarantees, we note that the only existing analysis of such variants in the non-private setting is the one of Karimireddy et al. (2019) for ℓ_1 and box constraints. Their proof relies on an alternation between `good` (that provably progress) and `bad` steps (that do not increase the objective), which does not transfer to the private setting. Extending such results to DP-ERM is an exciting direction for future work.

Chapter 6

Quantifying the Impact of Privacy on Fairness and Accuracy

Chapter Abstract

We theoretically study the impact of differential privacy on fairness in classification. We prove that, given a class of models, popular group fairness measures are pointwise Lipschitz-continuous with respect to the parameters of the model. This result is a consequence of a more general statement on accuracy conditioned on an arbitrary event (such as membership to a sensitive group), which may be of independent interest. We use this Lipschitz property to prove a non-asymptotic bound showing that, as the number of samples increases, the fairness level of private models gets closer to the one of their non-private counterparts. This bound also highlights the importance of the confidence margin of a model on the disparate impact of differential privacy.

This Chapter is mostly based on the paper: *"Differential Privacy Has Bounded Impact on Fairness in Classification"* (Mangold, Perrot, Bellet, and Tommasi, 2023b), published at ICML 2023.

The code corresponding to this Chapter is available at https://github.com/pmangold/fairness-privacy.

6.1 Introduction

Until now, we have mostly discussed the privacy issues linked with the training of machine learning models. But other ethical concerns, like fairness of the model's predictions, have also attracted a lot of interest in the past few years. Fairness requires

models not to unjustly discriminate against specific individuals or subgroups of the population, while privacy preserves individual-level information about the training data from being inferred from the model. These two notions have been extensively studied in isolation: there exists numerous approaches to learn fair models (Caton and Haas, 2020; Mehrabi et al., 2021), or to preserve privacy (Dwork and Roth, 2014; Liu et al., 2021). However, only few works studied the interplay between privacy and fairness. In this Chapter, we take a step forward in this direction, proposing a new theoretical bound on the relative impact of privacy on fairness in classification.

Fairness takes various forms (depending on the task and context), and several definitions exist. On the one hand, the goal may be to ensure that similar individuals are treated similarly. This is captured by individual fairness (Dwork et al., 2012) and counterfactual fairness (Kusner et al., 2017). On the other hand, group fairness requires that decisions made by machine learning models do not unjustly discriminate against subgroups of the population. In this thesis, we focus on group fairness and consider four popular definitions, namely Equalized Odds (Hardt et al., 2016), Equality of Opportunity (Hardt et al., 2016), Accuracy Parity (Zafar et al., 2017), and Demographic Parity (Calders et al., 2009).

In this Chapter, we study the interplay between differential privacy and fairness in machine learning. We quantify the difference in fairness levels between private and non-private models in multi-class classification. We derive high probability bounds showing that this difference shrinks at a rate of $\widetilde{O}(\sqrt{p}/n)$. To obtain this result, we first prove that the accuracy of a model conditioned on an arbitrary event (such as membership to a sensitive group), is pointwise Lipschitz continuous with respect to the model parameters. This property is inherited by many popular group fairness notions, such as Equalized Odds, Equality of Opportunity, Accuracy Parity and Demographic Parity. Consequently, two sufficiently close models will have similar fairness levels. We then upper-bound the distance between the optimal non-private model and the private models obtained with privacy preserving mechanisms like output perturbation (Chaudhuri et al., 2011; Lowy and Razaviyayn, 2021) or `DP-SGD` (Song et al., 2013; Bassily et al., 2014b). These bounds hold for strongly convex empirical risk minimization formulations, potentially allowing explicit fairness-promoting convex regularization terms (Bechavod and Ligett, 2018; Huang and Vishnoi, 2019; Lohaus et al., 2020; Tran et al., 2021a). Combining these two results, we derive high probability bounds on the fairness loss due to privacy. They show that, with enough training examples, (i) given an optimal non-private model, enforcing privacy will not harm fairness too much, and (ii) given a private model, the corresponding (unknown) non-private optimal model cannot be vastly fairer. Our results also highlight the role of the *confidence margin* of models in the disparate impact of differential privacy: notably, if the non-private model has high per-group confidence, then our bound on the loss in fairness due to privacy will be smaller.

The contributions of this Chapter can be summarized as follows:

1. We prove that group fairness is pointwise Lipschitz, with a smaller constant for models with large margins.
2. We bound the distance between private and optimal models, and show that the difference in their fairness levels decreases in $\widetilde{O}(\sqrt{p}/n)$.
3. We show that this bound can be computed even when the optimal model is unknown, and numerically demonstrate that we obtain non-trivial guarantees.

6.2 Related work

The joint study of fairness and privacy in machine learning only goes back a few years, and has been the focus of a recent survey Fioretto et al., 2022. One may identify three main research directions. First, it has been empirically observed that privacy can exacerbate unfairness (Bagdasaryan et al., 2019; Pujol et al., 2020; Farrand et al., 2020; Uniyal et al., 2022) and, conversely, that enforcing fairness can lead to more privacy leakage for the unprivileged group (Chang and Shokri, 2021). These empirical results suggest that some properties of the dataset (such as group sizes and groupwise input norms) and the choice of the private training method may affect the extent of these disparate impacts. Unfortunately, these observations are not supported by theoretical results, and it is not clear why and when disparate impact occurs. Second, a few approaches have been proposed to learn models that are both fair and privacy preserving. However, these works either have limited theoretical guarantees on their performance (Kilbertus et al., 2018; Xu et al., 2019; Xu et al., 2020; Tran et al., 2021b), or learn stochastic models which might not be usable in contexts where deterministic decisions are expected (Jagielski et al., 2019; Mozannar et al., 2020). Finally, a few works have shown that fairness and privacy are incompatible in some settings, in the sense that there exists data distributions where enforcing one prevents the other from being satisfied (Sanyal et al., 2022), or where enforcing both implies trivial utility (Cummings et al., 2019; Agarwal, 2020). While appealing at first glance, these results usually consider unrealistic cases that are hardly encountered in practice. In this Chapter, e also study fairness and privacy jointly but rather than studying whether they may be achieved simultaneously, we investigate the relative difference in fairness level between private and non-private models.

To the best of our knowledge, the work closest to ours is the one of Tran et al. (2021a). They analyze the impact of privacy on fairness in Empirical Risk Minimization, where their notion of fairness is defined as the difference between the excess risk computed on the overall population and the excess risk computed on a subgroup of the population. They study the expected behavior over the possible private models while our

results are model-specific. In line with the results of this Chapter, their results suggest that the distance to the decision boundary plays a key role in the disparate impact of differential privacy. However, the quantity appearing in their result is based on a second-order Taylor approximations which is loose for popular classification loss functions. In contrast, the quantity appearing in our bounds is precisely the confidence margin considered in prior work on multi-class margin-based classification (Cortes et al., 2013). Finally and most importantly, loss-based fairness does not necessarily imply that the actual decisions taken by the model are fair with respect to standard group-fairness notions (Lohaus et al., 2020). In contrast, we provide guarantees in terms of these widely-accepted group fairness definitions.

6.3 Preliminaries

6.3.1 Classification

We consider a multi-class classification setting with a feature space $\mathcal{X}$, a finite set of labels $\mathcal{Y}$, and a finite set $\mathcal{S}$ of values for the sensitive attribute. Let $\mathcal{D}$ be a distribution over $\mathcal{X} \times \mathcal{S} \times \mathcal{Y}$, and $D = \{(x_1, s_1, y_1), \ldots, (x_n, s_n, y_n)\}$ be a training set of n examples drawn i.i.d. from $\mathcal{D}$. Let $\mathcal{H}$ be a space of real-valued functions $h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ equipped with a norm $\|\cdot\|_{\mathcal{H}}$. For an example $x \in \mathcal{X}$, the predicted label is the one with the highest value, that is $H(x) = \arg\max_{y \in \mathcal{Y}} h(x, y)$. In case of a tie, a random label among the most likely ones is predicted. The confidence margin of a model h for an example-label pair (x, y) is defined as $\rho(h, x, y) = h(x, y) - \max_{y' \neq y} h(x, y')$ (Cortes et al., 2013). This confidence margin is positive when the example x is classified as y by h and negative otherwise. We assume that the margin is Lipschitz-continuous in the model h.

Assumption 6.3.1 (Lipschitzness of the margin)**.** *We assume that ρ is Lipschitz-continuous in its first argument, that is for all $h, h' \in \mathcal{H}$ and $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$|\rho(h, x, y) - \rho(h', x, y)| \leq L_{x,y} \|h - h'\|_{\mathcal{H}} \ ,$$

where $L_{x,y} < +\infty$ may depend on the example (x, y).

This assumption is not very restrictive. Typically, it is satisfied by any class of differentiable model with bounded gradients. As an illustration, consider linear models of the form $h(x, y) = W_y^T x$ where W is a real-valued matrix where each line is a vector W_y of label-specific parameters. Define $\|h - h'\|_{\mathcal{H}} = \|W - W'\|_2$.Then, we have $L_{x,y} = 2\|x\|_2$ since $|\rho(h, x, y) - \rho(h', x, y)| \leq |h(x, y) - h'(x, y)| + \max_{y' \neq y} |h(x, y') - h'(x, y')| \leq 2\|x\|_2 \|h - h'\|_{\mathcal{H}}$.

The goal of a learning algorithm $\mathcal{A} : (\mathcal{X} \times \mathcal{S} \times \mathcal{Y})^n \to \mathcal{H}$ is to find the best possible model to solve the task. In this work, the quality of a model h is evaluated

through its accuracy $\mathrm{Acc}(h) = \mathbb{P}(H(X) = Y)$ but also its fairness level (as defined in Section 6.3.2). Furthermore, given a non-private algorithm $\mathcal{A}$, our goal will be to compare the quality of its output to that of a private version $\mathcal{A}^{\mathrm{priv}}$ of $\mathcal{A}$ that guarantees differential privacy.

6.3.2 Fairness

We focus on group fairness. These definitions are based on the idea that a group of individuals should not be discriminated against, compared to the overall population. Usually, these groups are defined by the sensitive attribute from $\mathcal{S}$. However, in some cases, it is necessary to consider more fine grained partitions. This is for example the case in Equalized Odds (Hardt et al., 2016), where a model is fair if its performance is the same on the overall population and on subgroups of individuals that share the same sensitive group and the same label. Thus, for the sake of generality, we assume that the data can be partitioned into K disjoint groups denoted by $D_1, \ldots, D_k, \ldots, D_K$. As in Maheshwari and Perrot (2022), we consider fairness definitions that, for each group k, can be written as:

$$F_k(h, D) = C_k^0 + \sum_{k'=1}^{K} C_k^{k'} \, \mathbb{P}(H(X) = Y \mid D_{k'}) \quad , \tag{6.3.1}$$

where the $C_k^{k'}$'s are group specific values independent of h, that typically depend on the size of the groups. In Appendix C.1, we show that usual group fairness notions such as Demographic Parity (with binary labels) (Calders et al., 2009), Equality of Opportunity (Hardt et al., 2016), Equalized Odds (Hardt et al., 2016), and Accuracy Parity (Zafar et al., 2017) can all be expressed in the form of (6.3.1). By convention, we consider that $F_k(h, D) > 0$ when the group k is advantaged by h compared to the overall population, $F_k(h, D) < 0$ when the group is disadvantaged and $F_k(h, D) = 0$ when h is fair for group k.

In some cases, rather than measuring fairness for each group k independently, it is interesting to summarize the information with an aggregate value. For example, we will use the mean of the absolute fairness level of each group:

$$\mathrm{Fair}(h, D) = \frac{1}{K} \sum_{k=1}^{K} |F_k(h, D)| \quad , \tag{6.3.2}$$

which is 0 when h is fair and positive when it is unfair.

6.4 Pointwise Lipschitzness and Group Fairness

Here, we show that several *group fairness notions are pointwise Lipschitz* with respect to the model. To this end, we first prove a more general result on the pointwise

Lipschitzness of accuracy conditionally on an arbitrary event.

6.4.1 Pointwise Lipschitzness of Conditional Accuracy

We first relate the difference of conditional accuracy of two models to the distance that separates them. This is summarized in the next theorem.

Theorem 6.4.1 (Pointwise Lipschitzness of Conditional Accuracy)**.** *Let $\mathcal{H}$ be a set of real-valued functions with $L_{X,Y}$ the Lipschitz constants defined in Assumption 6.3.1. Let $h, h' \in \mathcal{H}$ be two models, (X, Y, S) be a triple of random variables with distribution $\mathcal{D}$, and E be an arbitrary event. Assume that $\mathbb{E}\left({}^{L_{X,Y}}/_{|\rho(h',X,Y)|} \mid E\right) < +\infty$, then*

$$|\,\mathbb{P}(H(X) = Y \mid E) - \mathbb{P}(H'(X) = Y \mid E)| \leq \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h,X,Y)|} \mid E\right) \|h - h'\|_{\mathcal{H}} \ . \qquad \text{(Lip)}$$

Proof. (Sketch) The proof of this theorem is in two steps. First, we use the Lipschitzness of the margin (Assumption 6.3.1), the triangle inequality, and the union bound to show that

$$|\,\mathbb{P}\left(H(X) = Y \mid E\right) - \mathbb{P}\left(H'(X) = Y \mid E\right)| \leq \mathbb{P}\left(\frac{L_{X,Y}}{|\rho(h,X,Y)|} \geq \frac{1}{\|h-h'\|_{\mathcal{H}}} \;\middle|\; E\right).$$

Then, applying Markov's inequality gives the desired result. The complete proof can be found in Appendix C.2. □

Theorem 6.4.1 shows the pointwise lipschitzness of $h \mapsto \mathbb{P}\left(H(X) = Y \mid E\right)$. Furthermore, it underlies the importance of having a large confidence margin $\rho(h, x, y)$ for a model h predicting label y for an example x. Hence, ${}^{L_{x,y}}/_{|\rho(h,x,y)|}$ is small when the model h is confident in its prediction for the true label y. This implies that, when the probability (given E) that a point has a small margin is small, $\mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h,X,Y)|} \;\middle|\; E\right)$ is also small. This is notably the case for large margin classifiers.

It is worth noting that the bound presented Theorem 6.4.1 can be tightened (at the expense of readability) without affecting the quantities that need to be controlled, that is the margin $|\rho(h, x, y)|$ and the distance $\|h - h'\|_{\mathcal{H}}$. Hence, note that given $(x, y) \in \mathcal{X} \times \mathcal{Y}$, if $|\rho(h, x, y)| \geq L_{x,y}\|h - h'\|_{\mathcal{H}}$, then it means that h's margin is large enough to ensure that h and h' have the same prediction on x. The corresponding term in the expectation may then be accounted for as zero, improving the upper bound (Remark C.2.2). Interestingly, if all the examples are classified with such a large margin, our bound becomes 0, further hinting toward the importance of large margin classifiers. This result may be further tightened by using a Chernoff bound instead of Markov's inequality (remark C.2.1), yielding $|\,\mathbb{P}(H(X)=Y|E)-\mathbb{P}(H'(X)=Y|E)| \leq \beta_{X,Y}(h)$, with

$$\beta_{X,Y}(h) = \inf_{t \geq 0} \left\{ e^{t\|h-h'\|_{\mathcal{H}}}\, \mathbb{E}\left(e^{-\frac{t|\rho(h,X,Y)|}{L_{X,Y}}} \middle| E\right)\right\} \ .$$

In the subsequent theoretical developments, we use the bound derived in Theorem 6.4.1 for the sake of readability. In the numerical experiments (Section 6.6), we use the version of the bound that yields the tightest results by combining both of the aforementioned techniques.

6.4.2 Pointwise Lipschitzness of Group Fairness Notions

We now use Theorem 6.4.1's general result to relate the fairness levels of two classifiers, based on their distance. In Theorem 6.4.2, we show that fairness notions in the form of (6.3.1) are pointwise Lipschitz.

Theorem 6.4.2 (Pointwise Lipschitzness of Fairness)**.** *Let $h, h' \in \mathcal{H}$, and $L_{X,Y}$ defined as in Assumption 6.3.1. For any fairness notion of the form of (6.3.1), we have, for all $k \in [K]$,*

$$|F_k(h, D) - F_k(h', D)| \leq \chi_k(h, D)\|h - h'\|_{\mathcal{H}} \ .$$

with $\chi_k(h, D) = \sum_{k'=1}^{K} |C_k^{k'}| \, \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h,X,Y)|} \,\middle|\, D_{k'}\right)$. Similarly, for the aggregate measure of fairness defined in (6.3.2),

$$|Fair(h, D) - Fair(h', D)| \leq \frac{1}{K}\sum_{k=1}^{K} \chi_k(h, D)\|h - h'\|_{\mathcal{H}} \ .$$

Proof. (Sketch) To prove the first claim, we use the triangle inequality to show that, for each group, the absolute difference in fairness is bounded by a combination of absolute differences between conditional probabilities. We can then apply Theorem 6.4.1. The second claim follows by applying the first one to each group independently. The complete proof is provided in Appendix C.3. □

Theorem 6.4.2 implies that *classifiers that are sufficiently close have similar fairness levels*. This has two major consequences when studying a given model. On the one hand, we have an upper bound on the harm that can be done to fairness: small variations of the model cannot make it much more unfair. On the other hand, we have a lower bound on the distance needed to make a model fair: making the model significantly more fair requires to substantially alter it. In the next corollary, we instantiate Theorem 6.4.2 for various popular group fairness notions, and for accuracy.

Corollary 6.4.1. *Let $h, h' \in \mathcal{H}$, and $L_{X,Y}$ defined as in Assumption 6.3.1. The difference in fairness or accuracy between h and h' can be bounded as follows.*

***Equalized Odds (Hardt et al., 2016)**: the data is divided into $K = |\mathcal{Y} \times \mathcal{S}|$ groups such that for all $(y, r) \in \mathcal{Y} \times \mathcal{S}$,*

$$\chi_{(y,r)}(h, D) = \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|} \,\middle|\, Y = y\right) + \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|} \,\middle|\, Y = y, S = r\right) \quad .$$

***Equality of Opportunity (Hardt et al., 2016)**: we let $\mathcal{Y}' \subseteq \mathcal{Y}$ the set of desirable outcomes. The data is divided into $K = |\mathcal{Y} \times \mathcal{S}|$ such that for all $(y, r) \in \mathcal{Y} \times \mathcal{S}$,*

$$\chi_{(y,r)}(h, D) = \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|} \,\middle|\, Y = y, S = r\right) + \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|} \,\middle|\, Y = y\right) \quad ,$$

if y is a desired outcome, and $\chi_{(y,r)}(h, D) = 0$ otherwise.

***Accuracy Parity (Zafar et al., 2017)**: the data is divided into $K = |\mathcal{S}|$ groups such that for all $r \in \mathcal{S}$,*

$$\chi_{(r)}(h, D) = \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|}\right) + \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|} \,\middle|\, S = r\right) \quad .$$

***Demographic Parity (Binary Labels) (Calders et al., 2009)**: the data is divided into $K = |\mathcal{Y} \times \mathcal{S}|$ groups such that for all $(y, r) \in \mathcal{Y} \times \mathcal{S}$,*

$$\chi_{(y,r)}(h, D) = \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|}\right) + \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|} \,\middle|\, S = r\right) \quad .$$

***Accuracy**: the data is in a single group, such that*

$$\chi(h, D) = \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|}\right) \quad .$$

Proof. This corollary follows from Theorem 6.4.2 by replacing the $C_k^{k'}$'s by their appropriate values (depending on the considered notion). See Appendix C.1 for more details. □

Corollary 6.4.1 shows that our results are applicable to several *group fairness notions*, but also to *accuracy*. Note that the pointwise Lipschitz constant $\chi_k(h, D)$ depends on the considered notion. In Section 6.5, we use these results to quantify the relative fairness level between private and non-private models.

6.5 Bounding the Relative Fairness of Private Models

In this section, we quantify the difference of fairness between a private model and its non-private counterpart. Let $\ell : \mathcal{H} \times \mathcal{X} \times \mathcal{S} \times \mathcal{Y} \to \mathbb{R}$ be a loss function. Assume ℓ

is Λ-Lipschitz, and μ-strongly-convex with respect to its first variable. Assume the norm $\|\cdot\|_{\mathcal{H}}$ is Euclidean, and that $\mathcal{H}$ is convex. We define the optimal model $h^* \in \mathcal{H}$ as

$$h^* = \arg\min_{h \in \mathcal{H}} f(h) = \frac{1}{n} \sum_{i=1}^{n} \ell(h; x_i, s_i, y_i) \ . \tag{6.5.1}$$

Two mechanisms are commonly used to find a differentially private approximation h^{priv} of h^*: output perturbation (Chaudhuri et al., 2011; Lowy and Razaviyayn, 2021), and `DP-SGD` (Bassily et al., 2014b; Abadi et al., 2016a). For both mechanisms, the distance $\|h^{\text{priv}} - h^*\|_{\mathcal{H}}$ can be upper bounded with high probability. In this section, we recall these two mechanisms and the corresponding high probability upper bounds. We then plug these bounds in Theorem 6.4.2 to bound the fairness level of the private solution h^{priv} relatively to the one of the true solution h^*.

6.5.1 Bounding the Distance between Private and Optimal Classifiers

Output perturbation. Output perturbation computes the non-private solution h^* of (6.5.1), and releases a private estimate by the Gaussian mechanism:

$$h^{\text{priv}} = \pi_{\mathcal{H}}(h^* + \mathcal{N}(\sigma^2 \mathbb{I}_p)) \ ,$$

where $\pi_{\mathcal{H}}$ is the projection on $\mathcal{H}$. Let Δ be the sensitivity of the function $D \mapsto \arg\min_{w \in \mathcal{H}} f(w; D)$. In our setting, we have $\Delta = 2\Lambda/\mu n$. Then, given $0 < \epsilon, \delta < 1$, h^{priv} is (ϵ, δ)-differentially private as long as $\sigma^2 \geq 2\Delta^2 \log(1.25/\delta)/\epsilon^2$. We bound the distance between h^{priv} and h^* with high probability in Lemma 6.5.1 (proved in Appendix C.4).

Lemma 6.5.1. *Let h^{priv} be the vector released by output perturbation with noise $\sigma^2 = 8\Lambda^2 \log(1.25/\delta)/\mu^2 n^2 \epsilon^2$, and $0 < \zeta < 1$, then with probability at least $1 - \zeta$,*

$$\|h^{priv} - h^*\|_2^2 \leq \frac{32 p \Lambda^2 \log(1.25/\delta) \log(2/\zeta)}{\mu^2 n^2 \epsilon^2} \ .$$

`DP-SGD`. `DP-SGD` starts from some $h^0 \in \mathcal{H}$ and updates it using stochastic gradients. That is, with $\gamma > 0$, $i \sim \mathcal{U}([n])$, and $\eta^t \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_p)$, we iteratively update

$$h^{t+1} = \pi_{\mathcal{H}}(h^t - \gamma(\nabla \ell(h^t; x_i, y_i) + \eta^t)) \ .$$

After $T > 0$ iterations, we release $h^{\text{priv}} = h^T$. Given $0 < \epsilon, \delta < 1$, h^{priv} is (ϵ, δ)-differentially private when $\sigma^2 \geq 64\Lambda^2 T^2 \log(3T/\delta) \log(2/\delta)/n^2 \epsilon^2$. Assuming the loss function is smooth in its first parameter, we bound the distance between h^{priv} and h^* with high probability in Lemma 6.5.2 (proved in Appendix C.5).

Lemma 6.5.2. *Let h^{priv} be output of* DP-SGD *with $\sigma^2 = \frac{64\Lambda^2 T^2 \log(3T/\delta)\log(2/\delta)}{n^2\epsilon^2}$. Assume that $\sigma_*^2 = \mathbb{E}_{i\sim[n]} \|\nabla \ell(h^*; x_i, y_i)\|^2 \leq \sigma^2$. Let $0 < \zeta < 1$, then with probability at least $1-\zeta$,*

$$\|h^{priv} - h^*\|_2^2 = \widetilde{O}\left(\frac{p\Lambda^2 \log(1/\delta)^2}{\zeta\mu^2 n^2 \epsilon^2}\right) ,$$

where $\widetilde{O}$ ignores logarithmic terms in n (the number of examples) and p (the number of model parameters).

Remark 6.5.1. *For clarity of exposition in Lemma 6.5.2, we did not use minimal assumptions and used the simplest variant of* DP-SGD*. Notably, the assumption on σ_* can be removed by using variance reduction schemes, and tighter bounds on σ can also be obtained using Rényi Differential Privacy (Mironov, 2017). Similarly, the assumption $\epsilon < 1$ is only used to give simple closed-form bounds. Strong convexity and smoothness assumptions can be relaxed as well.*

Table 6.1: Upper bound, with 99% probability, on the difference of fairness between private and non-private models for different fairness measures and accuracy. Privacy budget is $\epsilon = 1$ and $\delta = 1/n^2$ where n is the number of samples in the training data.

Dataset	Equality of Opportunity	Equalized Odds	Demographic Parity	Accuracy Parity	Accuracy
celebA ($n = 182,339$)	0.1044	0.0975	0.0975	0.0975	0.0487
folktables ($n = 1,498,050$)	0.0017	0.0026	0.0026	0.0026	0.0013

6.5.2 Bounding the Fairness of Private Models

We now state our central result (Theorem 6.5.1), where we bound the fairness of h^{priv} relatively to the one of h^*.

Theorem 6.5.1. *Let h^* be the solution of (6.5.1), and h^{priv} its private estimate obtained by output perturbation. Let $h^{\text{ref}} \in \{h^{priv}, h^*\}$, and $0 < \zeta < 1$. Then, the difference of fairness of group $k \in [K]$ satisfies, with probability at least $1-\zeta$,*

$$|F_k(h^{priv}, D) - F_k(h^*, D)| \leq \frac{\chi_k(h^{\text{ref}}, D) L\Lambda\sqrt{32p\log(1.25/\delta)\log(2/\zeta)}}{\mu n\epsilon} .$$

Similarly, if h^{priv} is estimated through DP-SGD*, we have that, with probability at least $1-\zeta$,*

$$|F_k(h^{priv}, D) - F_k(h^*, D)| \leq \widetilde{O}\left(\frac{\chi_k(h^{\text{ref}}, D) L\Lambda\sqrt{p\log(1/\delta)}}{\sqrt{\zeta}\mu n\epsilon}\right) ,$$

where $\widetilde{O}$ ignores logarithmic terms in n (the number of examples) and p (the number of model parameters).

Proof. By Lemma 6.5.1 or Lemma 6.5.2, we control the distance $\|h^{\text{priv}} - h^*\|$. Plugging this bound in Theorem 6.4.2 gives the result. □

This result shows that, when learning a private model, *the unfairness due to privacy vanishes at a $\widetilde{O}(\sqrt{p}/n)$ rate*. To the best of our knowledge, our result is the first to quantify this rate. Importantly, *it highlights the role of the confidence margin* of the classifier on the impact of differential privacy on fairness. This is in line with previous empirical and theoretical work that identified the groupwise distances to the decision boundary as an important factor (Tran et al., 2021a; Tran et al., 2021b). However, our bounds are the first to quantify this impact through a classic notion of confidence margin studied in learning theory (Cortes et al., 2013).

Our result may be interpreted and used in various ways. A first example is the case where the private model is known but its optimal non-private counterpart is not. There, our result guarantees that, given enough examples, the fairness level of the private model is close to the one of the optimal non-private model. This allows the practitioner to give guarantees on the model, that the end user can trust. A second example is the case where the true model h^* is owned by someone who cannot share it, due to privacy concerns. Imagine that the model needs to be audited for fairness. Then, the model owner can compute a private estimate of their model, and send it to the (honest but curious) auditing company. The bound allows to obtain fairness bounds for the true model from the inspection of the private one, and thus acts as a certificate of correctness of the audit done on the private version of the model.

Remark 6.5.2. *The fairness guarantee for the private model given by Theorem 6.5.1 is relative to the fairness of the optimal model h^*, which may itself be quite unfair. A standard approach to promote fair models is to use convex relaxations of fairness as regularizers to the ERM problem (Bechavod and Ligett, 2018; Huang and Vishnoi, 2019; Lohaus et al., 2020). Interestingly, to be able to use output perturbation, we only require the objective function of (6.5.1) to be strongly convex and Lipschitz over $h \in \mathcal{H}$, which is the case for these relaxations when they are combined with a squared ℓ_2-norm. For binary classification with two sensitive groups, Lohaus et al. (2020) proved that, with a proper choice of regularization parameters, this approach can yield a fair h^* (see their Theorem 1 for more details). Combined with our results, this paves the way for the design of algorithms that learn provably private and fair classifiers. However, several crucial challenges remain to make this approach work in practice, such as (i) finding the appropriate regularization parameters privately, and (ii) providing guarantees on the resulting classifiers' accuracy. We leave this for future work.*

6.6 Numerical Experiments

In this section, we numerically illustrate the upper bounds from Section 6.5.2. We use the `celebA` (Liu et al., 2015) and `folktables` (Ding et al., 2021) datasets, which respectively contain $202,599$ and $1,664,500$ samples, with 39 and 10 features (including one sensitive attribute, sex, that is not not used for prediction), and binary labels. For each dataset, we use 90% of the records for training, and the remaining 10% for empirical evaluation of the bounds. We train ℓ_2-regularized logistic regression models, ensuring that the underlying optimization problem is 1-strongly-convex. This allows learning private models by output perturbation, for which the bound from Theorem 6.5.1 holds.

In Section 6.6.1, we show that we obtain non-trivial guarantees on the private model's fairness and accuracy. Then, we study the influence of the number of training samples and of the privacy budget ϵ in Section 6.6.2, and discuss the tightness of our result in Section 6.6.3.

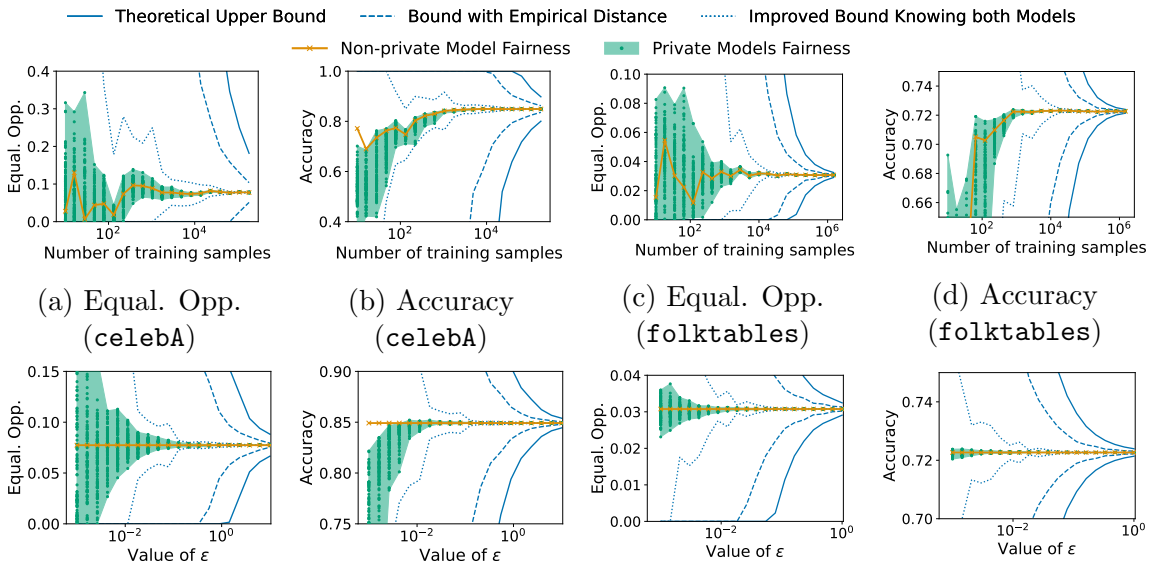
6.6.1 Value of the Upper Bounds

In Table 6.1, we compute the value of Theorem 6.5.1's bounds. We learn a non-private ℓ_2-regularized logistic regression model, and use it to compute the bounds (averaged over the two groups) for multiple fairness and accuracy measures on two datasets. In all cases, our results give non-trivial guarantees on the difference of fairness: it is bounded by at most 0.105 for `celebA` and 0.0026 for `folktables`. This means that any $(1, 1/n^2)$-DP model learned by output perturbation will, with high probability, achieve a fairness level within this margin of that of the non-private model.

6.6.2 Influence of the Training Set Size and Privacy Budget

We now verify numerically the rate at which fairness and accuracy levels decrease when increasing the number of training records or privacy budget. In Figure 6.6.1, we plot the optimal model's equality of opportunity and accuracy, as a function of (i) in the first line, the number of samples n used for training, or (ii) in the second line, the privacy budget ϵ (see Appendix D.3 for results with other fairness measures). For each value of n and ϵ, we plot Theorem 6.5.1's theoretical guarantees (solid blue line). With $\epsilon = 1$, our bounds give meaningful guarantees for $n \geq 100,000$ records on both `celebA` and `folktables` datasets (Figures 6.6.1a to 6.6.1d). When using all records, we obtain meaningful bounds for $\epsilon \geq 1$ for `celebA` and $\epsilon \geq 0.1$ for `folktables` (Figures 6.6.1e to 6.6.1h). Additionally, note that we obtain *both upper and lower bounds* on fairness and accuracy, confirming remarks from Section 6.4.2.

We also report the fairness and accuracy levels of 100 private models computed by



(a) Equal. Opp. (`celebA`)

(b) Accuracy (`celebA`)

(c) Equal. Opp. (`folktables`)

(d) Accuracy (`folktables`)

(e) Equal. Opp. (`celebA`)

(f) Accuracy (`celebA`)

(g) Equal. Opp. (`folktables`)

(h) Accuracy (`folktables`)

Figure 6.6.1: Equality of opportunity (Equal. Opp.) and Accuracy levels for optimal non-private model and random private ones as a function of the number of training records n (first line, with $\epsilon = 1$ and $\delta = 1/n^2$) and of the privacy budget ϵ (second line, using all available training records). For each value of n and ϵ, we sample 100 private models and take their minimum and maximum fairness/accuracy values to mark the area of attainable values. The solid blue line gives the theoretical guarantees from Theorem 6.5.1, while the dashed and dotted line give finer bounds when more information is available (see Section 6.6.3 for details).

output perturbation (in green in Figure 6.6.1). As predicted by our theory, their fairness and accuracy converges towards the ones of their non-private counterparts as n and ϵ increase. Interestingly, our bounds seem to follow the same tendency as what we observe empirically (albeit with a larger multiplicative constant), suggesting that they capture the correct dependence in n and ϵ. We further discuss the tightness of our results in next section.

6.6.3 Tightness of the Bound

We now argue that the two major factors of looseness in our results are (i) the upper bound on $\|h^{\text{priv}} - h^*\|$ and (ii) the looseness of Assumption 6.3.1. While these cannot be improved in general, specific knowledge of h^{priv} and h^* (that is typically not available due to privacy) can lead to tighter bounds. First, when the distance $\|h^{\text{priv}} - h^*\|$ is known, we can use its actual value rather than the upper bounds of

Section 6.5.1 (see dashed blue line in Figure 6.6.1). Second, when both h^{priv} and h^* are known, Assumption 6.3.1 can be substantially refined (see details in Appendix D.3.3). We evaluate this bound for the private model that is the farthest away from the non-private one (see dotted blue line in Figure 6.6.1). The resulting bound appears to be tight up to a small multiplicative constant. These two observations suggest that our bounds cannot be significantly tightened, unless one can obtain such knowledge through either private computation or additional assumptions on the data.

6.7 Conclusion

We proved that the fairness (and accuracy) costs induced by privacy in differentially private classification vanishes at a $\widetilde{O}(\sqrt{p}/n)$ rate, where n is the number of training records, and p the number of parameters. This rate follows from a general statement on group fairness measures' regularity, that we prove to be pointwise Lipschitz with respect to the model. The pointwise Lipschitz constant explicitly depends on the confidence margin of the model. Importantly, our bounds does not require the knowledge of the optimal (non-private) model: they can thus be used in practical privacy-preserving scenarios. We numerically evaluate our bounds on real datasets, and highlight practical settings where non-trivial guarantees can be obtained.

Our results could help build more trustworthy machine learning models, by guaranteeing that their fairness and accuracy approximately match the one of the non-private model. We believe that our results are applicable to privacy-preserving methods beyond output perturbation and `DP-SGD`. Indeed, deriving high-probability bounds on the distance between the private and the non-private model is sufficient to apply them. Note however that our bounds crucially rely on the uniqueness of problem (6.5.1)'s solution, which is guaranteed by strong convexity. Relaxing this hypothesis is challenging, but would greatly broaden the scope of our results.

We stress that our results do not provide fairness guarantees *per se*, but only bound the difference of fairness between models. It is nonetheless a first step towards a more complete understanding of the interplay between privacy, fairness, and accuracy. We believe that our results can guide the design of fairer privacy-preserving machine learning algorithms. A first promising direction in this regard is to combine our bounds with fairness-promoting convex regularizers, as discussed in Remark 6.5.2. Another direction is the design of methods to privately learn models with large-margin guarantees, as recently considered by Bassily et al. (2022a). Our results, which explicitly depend on the confidence margin of the model, suggest that better fairness guarantees could be obtained for these methods.

Chapter 7

Conclusion and Perspectives

7.1 Conclusion

In this thesis, we investigated the role of problem structure in differentially private machine learning. We proposed two new differentially private optimization algorithms for empirical risk minimization. These algorithms can exploit structural properties of the problem to provably achieve a better privacy-utility trade-off than existing algorithms .We also studied how differential privacy impacts fairness in classification problems, and highlighted the role of the confidence of the model across sub-groups of the population.

We proposed in Chapter 4 a differentially private stochastic coordinate descent algorithm. At each iteration of this algorithm, one coordinate is sampled uniformly at random. This coordinate is then updated with a noisy proximal gradient step. Noise addition allows to guarantee differential privacy, but does not prevent from using large coordinate-wise step sizes (like in non-private proximal coordinate descent). Our differentially private coordinate descent algorithm can therefore adapt to the imbalance in gradient's coordinates scales, outperforming existing algorithms in terms of privacy-utility trade-off when the problem at hand is imbalanced. We showed this through a careful analysis of its convergence properties and derived corresponding lower bounds.

We then proposed in Chapter 5 a greedy variant of the differentially private coordinate descent algorithm. This algorithm can further improve utility by choosing the updated coordinates greedily using the report noisy max mechanism. Thanks to these greedy updates, the algorithm can naturally exploit structural properties like the sparsity of the solution. Under favorable structural assumptions, we proved that the dependence of our algorithm's utility on the dimension is reduced from polynomial to logarithmic. We demonstrated that this algorithm can find good (and sparse) parameters in very few iterations, even when the dimension is large.

Finally we investigated in Chapter 6, for classification tasks, the impact of differential privacy on the level of fairness of the learned model. To this end, we derived a bound on the difference of fairness between a private model and its non-private counterpart. This bound follows from the fact that many group fairness notions are pointwise Lipschitz when the decision function is Lipschitz in its parameters. Our results highlight the key role of the confidence margin in this problem, and in particular of its distribution among the different sub-groups of the population.

7.2 Perspectives

Non-uniform Sampling of Coordinates. In Chapter 4, we studied differentially private coordinate descent with *uniform sampling* of the coordinates. In some problems, it may be relevant to sample them non-uniformly (*e.g.,* proportionally to the coordinate-wise smoothness constants, or adaptively). This was studied by Nesterov (2010), Richtárik and Takáč (2014), and Richtárik and Takáč (2016) in the non-private setting. In particular, sampling coordinates with large smoothness constants more often can help find an approximate solution faster (although the algorithm tends to stall after a certain number of iterations). This could be beneficial in differentially private optimization. Due to privacy, we are necessarily finding an approximate solution, yet performing fewer iterations helps reducing the amount of injected noise, possibly improving the precision of this approximation.

Clipping. In practice, the differentially private coordinate descent algorithms we proposed heavily rely on the use of gradient clipping. However, this is not covered by our theory, which makes it difficult to choose the value of this threshold. Recently, Koloskova et al. (2023) proposed a theoretical study of clipped (stochastic) gradient descent under an (L_0, L_1)-smoothness assumption. Using a coordinate-wise variant of this assumption could lead to improve theoretical understanding of these clipped coordinate descent algorithms. Such theoretical analysis could help in defining rules of thumb for setting these clipping thresholds.

Adaptive strategies have also been proposed for setting the clipping thresholds in an adaptive way (Pichapati et al., 2019; Andrew et al., 2021). Developing adaptive clipping strategies for differentially private coordinate methods could also help to alleviate the difficulty of setting the value of these thresholds appropriately.

Hyperparameter-Free Methods. Most of the existing differentially private optimization algorithms heavily rely on one or more hyperparameters (*e.g.,* number of iterations, step size, clipping thresholds, etc.). In particular, the differentially private coordinate descent algorithms we developed in this thesis use multiple parameters per coordinate. Although we have proposed methods to adapt these from one global

hyperparameter (some procedures exist for tuning them privately) it would be more practical to avoid setting them all together. Recently, hyperparameter-free optimization algorithms have regained in popularity, for instance through the works of Defazio and Mishchenko (2023), Mishchenko and Defazio (2023), and Khaled et al. (2023). Extending these ideas to the differentially private setting could yield important improvements in the performance and practical usability of differentially private optimization algorithms.

Screening and Support Recovery. Practical solvers for sparse learning problems are often based on coordinate descent, combined with screening methods (Fercoq et al., 2015; Massias et al., 2017; Massias et al., 2018; Bertrand et al., 2022). These methods aim at identifying coordinates that have already converged, and stop updating them to accelerate the convergence. In some sparse problems, this can result in massive performance gains, which could translate into better utility in differentially private settings.

More generally, (greedy) coordinate descent tend to identify the support of the model fast in the non-private setting (Klopfenstein et al., 2020; Fang et al., 2020). The algorithms developed in this thesis could thus be a promising starting point towards defining differentially private algorithms that can identify the support of a model.

Vertical Federated Learning. In federated learning, multiple agents aim at collaboratively training a model without sharing their data. The vertical flavor of federated learning covers the case where each agent holds a subset of the features. These problems have not been studied very extensively (contrary to other federated learning settings), but some approaches are based on coordinate descent methods (Liu et al., 2020). Differentially private vertical federated learning could therefore be an interesting application of the results we developed in this thesis.

Efficient Greedy Updates. In Chapter 5, we proposed a differentially private greedy coordinate descent algorithm. Although this algorithm can reduce the dependence on the dimension from polynomial to logarithmic, its iterations have an important computational cost. In non-private settings, multiple approaches have been proposed for reducing this cost (Dhillon et al., 2011; Karimireddy et al., 2019). Their approaches cast the greedy selection rule as a nearest neighbors search, and use methods like locality-sensitive hashing to compute an approximation of the greedy rule, reducing the computation cost. In the differentially private greedy coordinate descent algorithm, the greedy rule is always computed approximately (due to the privacy requirement). Therefore, using these approximate greedy selection rules could lead to reducing significantly the computational cost of our algorithm, possibly without altering its convergence properties too much. A promising perspective to solve

this problem is to use the differentially private locality-sensitive methods that were developed by Fernandes et al. (2021).

Proximal greedy coordinate descent. We proposed a proximal variant of differentially private greedy coordinate descent in Section 5.3.5, which achieves good empirical performance. Unfortunately, theoretically analyzing the convergence of this algorithm is very difficult. In the non-private setting, Karimireddy et al. (2019) proposed a modified variant of proximal greedy coordinate descent for ℓ_1-regularized and box-constrained problems, but their approach does not seem to be applicable in the differentially private setting. Developing a proper theory for this algorithm is a challenging open problem.

Logarithmic dependence on dimension: non-greedy algorithms. All differentially private algorithms whose utility can depend logarithmically on the dimension are based on greedy algorithms (*i.e.,* differentially private Frank-Wolfe Talwar et al. (2015), Bassily et al. (2021), and Asi et al. (2021) and our differentially private greedy coordinate descent algorithm). These algorithms all leverage the report noisy max mechanism. They require computing full gradients but only use one coordinate in the final update. To this date, it is not clear whether it is possible to achieve such utility without relying on greedy updates with the report noisy max mechanism.

More generally, developing algorithms that can adapt to the structure of the problem (like differentially private greedy coordinate descent does) without relying on greedy updates is an important research problem. Such algorithms would be interesting to achieve the best possible utility on the problem at hand, without requiring too much computation when greedy updates fail to exploit the structure of the problem.

Achieving fairness and privacy. The theoretical study we proposed in Chapter 6 shows that, in classification problems, differential privacy has a bounded impact on fairness. It does not, however, guarantee that the learned model is fair. This could be achieved by using fairness-promoting regularization strategies like the one proposed by Lohaus et al. (2020). In general, our results provide some insights on the study of fairness, that could guide further developments of differentially private mechanisms that foster fairness. One possible direction would be to guarantee differential privacy with non-uniform noise addition. If done properly, this could bias the models learned this way toward fairer ones.

Large margin classifiers. Our results from Chapter 6 also highlight the key role of the confidence margin on the fairness of the learned models. Therefore, adapting training methods, so that trained models have larger margins, could help in finding more fairness models. Note that large-margin models also tend to achieve better

privacy-utility trade-offs, although few results exist on this question (Bassily et al., 2022a; Bassily et al., 2022b). Nonetheless, this suggests that this direction is promising for training models that achieve good fairness, utility and privacy all at once.

Bibliography

Abadi, M., A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang (Oct. 2016a). "Deep Learning with Differential Privacy". In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. CCS '16. New York, NY, USA: Association for Computing Machinery, pp. 308–318.

Abadi, M., A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng (Mar. 2016b). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*.

Agarwal, S. (2020). "Trade-Offs between Fairness and Privacy in Machine Learning". In.

Amin, K., A. Kulesza, A. Munoz, and S. Vassilvtiskii (May 2019). "Bounding User Contributions: A Bias-Variance Trade-off in Differential Privacy". In: *Proceedings of the 36th International Conference on Machine Learning*. PMLR, pp. 263–271.

Andrew, G., O. Thakkar, B. McMahan, and S. Ramaswamy (2021). "Differentially Private Learning with Adaptive Clipping". In: *Advances in Neural Information Processing Systems*. Vol. 34. Curran Associates, Inc., pp. 17455–17466.

Asi, H., V. Feldman, T. Koren, and K. Talwar (Mar. 2021). "Private Stochastic Convex Optimization: Optimal Rates in ℓ_1 Geometry". In: *arXiv:2103.01516 [cs, math, stat]*.

Atchadé, Y. F., G. Fort, and E. Moulines (Jan. 2017). "On Perturbed Proximal Gradient Algorithms". In: *The Journal of Machine Learning Research* 18.1, pp. 310–342.

Bach, F. and E. Moulines (Dec. 2011). "Non-Asymptotic Analysis of Stochastic Approximation Algorithms for Machine Learning". In: *Proceedings of the 24th Interna-*

tional Conference on Neural Information Processing Systems. NIPS'11. Red Hook, NY, USA: Curran Associates Inc., pp. 451–459.

Bagdasaryan, E., O. Poursaeed, and V. Shmatikov (2019). "Differential Privacy Has Disparate Impact on Model Accuracy". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc.

Balle, B., G. Barthe, and M. Gaboardi (2018). "Privacy Amplification by Subsampling: Tight Analyses via Couplings and Divergences". In: *Advances in Neural Information Processing Systems*. Vol. 31. Curran Associates, Inc.

Balle, B., G. Barthe, and M. Gaboardi (Jan. 2020). "Privacy Profiles and Amplification by Subsampling". In: *Journal of Privacy and Confidentiality* 10.1.

Balle, B., G. Cherubin, and J. Hayes (May 2022). "Reconstructing Training Data with Informed Adversaries". In: *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1138–1156.

Bassily, R., V. Feldman, K. Talwar, and A. Guha Thakurta (2019). "Private Stochastic Convex Optimization with Optimal Rates". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc.

Bassily, R., C. Guzman, and A. Nandi (July 2021). "Non-Euclidean Differentially Private Stochastic Convex Optimization". In: *Proceedings of Thirty Fourth Conference on Learning Theory*. PMLR, pp. 474–499.

Bassily, R., M. Mohri, and A. T. Suresh (Apr. 2022a). *Differentially Private Learning with Margin Guarantees*.

Bassily, R., M. Mohri, and A. T. Suresh (Sept. 2022b). "Open Problem: Better Differentially Private Learning Algorithms with Margin Guarantees". In: *Proceedings of Thirty Fifth Conference on Learning Theory*. PMLR, pp. 5638–5643.

Bassily, R., A. Smith, and A. Thakurta (Oct. 2014a). "Differentially Private Empirical Risk Minimization: Efficient Algorithms and Tight Error Bounds". In: *arXiv:1405.7085 [cs, stat]*.

Bassily, R., A. Smith, and A. Thakurta (Oct. 2014b). "Private Empirical Risk Minimization: Efficient Algorithms and Tight Error Bounds". In: *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. Philadelphia, PA, USA: IEEE, pp. 464–473.

Beaudry, N. J. and R. Renner (Sept. 2012). *An Intuitive Proof of the Data Processing Inequality*.

Bechavod, Y. and K. Ligett (Mar. 2018). *Penalizing Unfairness in Binary Classification*.

Beck, A. (Oct. 2017). *First-Order Methods in Optimization*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Beck, A. and M. Teboulle (2009). "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems". In: p. 20.

Bellet, A., R. Guerraoui, M. Taziki, and M. Tommasi (Mar. 2018). "Personalized and Private Peer-to-Peer Machine Learning". In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 473–481.

Bertrand, Q., Q. Klopfenstein, P.-A. Bannier, G. Gidel, and M. Massias (Dec. 2022). "Beyond L1: Faster and Better Sparse Models with Skglm". In: *Advances in Neural Information Processing Systems* 35, pp. 38950–38965.

Boneh, D. and J. Shaw (1998). "Collusion-Secure Fingerprinting for Digital Data". In: *IEEE Transactions on Information Theory* 44.5, pp. 1897–1905.

Boyd, S. P. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge, UK ; New York: Cambridge University Press.

Bruck, R. E. (Nov. 1977). "On the Weak Convergence of an Ergodic Iteration for the Solution of Variational Inequalities for Monotone Operators in Hilbert Space". In: *Journal of Mathematical Analysis and Applications* 61.1, pp. 159–164.

Bubeck, S. (Nov. 2015). "Convex Optimization: Algorithms and Complexity". In: *Foundations and Trends® in Machine Learning* 8.3-4, pp. 231–357.

Bun, M., C. Dwork, G. N. Rothblum, and T. Steinke (June 2018). "Composable and Versatile Privacy via Truncated CDP". In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2018. New York, NY, USA: Association for Computing Machinery, pp. 74–86.

Bun, M. and T. Steinke (2016). "Concentrated Differential Privacy: Simplifications, Extensions, and Lower Bounds". In: *Theory of Cryptography*. Ed. by M. Hirt and A. Smith. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 635–658.

Bun, M., J. Ullman, and S. Vadhan (2014). "Fingerprinting Codes and the Price of Approximate Differential Privacy". In: p. 10.

Calders, T., F. Kamiran, and M. Pechenizkiy (Dec. 2009). "Building Classifiers with Independency Constraints". In: *2009 IEEE International Conference on Data Mining Workshops*, pp. 13–18.

Candès, E. J., M. B. Wakin, and S. P. Boyd (Dec. 2008). "Enhancing Sparsity by Reweighted L1 Minimization". In: *Journal of Fourier Analysis and Applications* 14.5, pp. 877–905.

Carlini, N., S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer (Dec. 2021a). "Membership Inference Attacks From First Principles". In: *arXiv:2112.03570 [cs]*.

Carlini, N., J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramèr, B. Balle, D. Ippolito, and E. Wallace (Jan. 2023). *Extracting Training Data from Diffusion Models*.

Carlini, N., C. Liu, Ú. Erlingsson, J. Kos, and D. Song (2019). "The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks". In: *28th USENIX Security Symposium (USENIX Security 19)*, pp. 267–284.

Carlini, N., F. Tramèr, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, Ú. Erlingsson, A. Oprea, and C. Raffel (2021b). "Extracting Training Data from Large Language Models". In: *30th USENIX Security Symposium (USENIX Security 21)*, pp. 2633–2650.

Caton, S. and C. Haas (Oct. 2020). "Fairness in Machine Learning: A Survey". In: *arXiv:2010.04053 [cs, stat]*.

Cauchy, A.-L. (1847). "Méthode Générale Pour La Résolution Des Systemes d'équations Simultanées". In: *Comp. Rend. Sci. Paris* 25.1847, pp. 536–538.

Cevher, V. and B. C. Vũ (July 2019). "On the Linear Convergence of the Stochastic Gradient Method with Constant Step-Size". In: *Optimization Letters* 13.5, pp. 1177–1187.

Chang, H. and R. Shokri (Jan. 2021). "On the Privacy Risks of Algorithmic Fairness". In: *arXiv:2011.03731 [cs, stat]*.

Chang, K.-W., C.-J. Hsieh, and C.-J. Lin (June 2008). "Coordinate Descent Method for Large-scale L2-loss Linear Support Vector Machines". In: *The Journal of Machine Learning Research* 9, pp. 1369–1398.

Chaudhuri, K. and C. Monteleoni (2008). "Privacy-Preserving Logistic Regression". In: *Advances in Neural Information Processing Systems*. Vol. 21. Curran Associates, Inc.

Chaudhuri, K., C. Monteleoni, and A. D. Sarwate (2011). "Differentially Private Empirical Risk Minimization". In: *Journal of Machine Learning Research* 12.29, pp. 1069–1109.

Chaudhuri, K. and S. A. Vinterbo (2013). "A Stability-based Validation Procedure for Differentially Private Machine Learning". In: *Advances in Neural Information Processing Systems*. Vol. 26. Curran Associates, Inc.

Chen, X., Z. S. Wu, and M. Hong (June 2020). "Understanding Gradient Clipping in Private SGD: A Geometric Perspective". In: *arXiv:2006.15429 [cs, math, stat]*.

Combettes, P. L. and V. R. Wajs (Jan. 2005). "Signal Recovery by Proximal Forward-Backward Splitting". In: *Multiscale Modeling & Simulation* 4.4, pp. 1168–1200.

Cortes, C., M. Mohri, and A. Rostamizadeh (May 2013). "Multi-Class Classification with Maximum Margin Multiple Kernel". In: *Proceedings of the 30th International Conference on Machine Learning*. PMLR, pp. 46–54.

Cummings, R., V. Gupta, D. Kimpara, and J. Morgenstern (June 2019). "On the Compatibility of Privacy and Fairness". In: *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*. UMAP'19 Adjunct. New York, NY, USA: Association for Computing Machinery, pp. 309–315.

Curry, H. B. (1944). "The Method of Steepest Descent for Non-Linear Minimization Problems". In: *Quarterly of Applied Mathematics* 2.3, pp. 258–261.

Damaskinos, G., C. Mendler-Dünner, R. Guerraoui, N. Papandreou, and T. Parnell (May 2021). "Differentially Private Stochastic Coordinate Descent". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35, pp. 7176–7184.

De Santis, M., S. Lucidi, and F. Rinaldi (Jan. 2016). "A Fast Active Set Block Coordinate Descent Algorithm for ℓ_1-Regularized Least Squares". In: *SIAM Journal on Optimization* 26.1, pp. 781–809.

Defazio, A., F. Bach, and S. Lacoste-Julien (2014). "SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives". In: *Advances in Neural Information Processing Systems*. Vol. 27. Curran Associates, Inc.

Defazio, A. and K. Mishchenko (May 2023). *Learning-Rate-Free Learning by D-Adaptation*.

Dhillon, I., P. Ravikumar, and A. Tewari (2011). "Nearest Neighbor Based Greedy Coordinate Descent". In: *Advances in Neural Information Processing Systems*. Vol. 24. Curran Associates, Inc.

Diffie, W. and M. E. Hellman (Aug. 2022). "New Directions in Cryptography". In: *Democratizing Cryptography: The Work of Whitfield Diffie and Martin Hellman*. 1st ed. Vol. 42. New York, NY, USA: Association for Computing Machinery, pp. 365–390.

Ding, F., M. Hardt, J. Miller, and L. Schmidt (2021). "Retiring Adult: New Datasets for Fair Machine Learning". In: *Advances in Neural Information Processing Systems*. Vol. 34. Curran Associates, Inc., pp. 6478–6490.

Duchi, J. C., M. I. Jordan, and M. J. Wainwright (Oct. 2013). "Local Privacy and Statistical Minimax Rates". In: *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pp. 429–438.

Dwork, C. (2006). "Differential Privacy". In: *Automata, Languages and Programming*. Ed. by M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 1–12.

Dwork, C., M. Hardt, T. Pitassi, O. Reingold, and R. Zemel (Jan. 2012). "Fairness through Awareness". In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. ITCS '12. New York, NY, USA: Association for Computing Machinery, pp. 214–226.

Dwork, C., F. McSherry, K. Nissim, and A. Smith (2006). "Calibrating Noise to Sensitivity in Private Data Analysis". In: *Theory of Cryptography*. Ed. by S. Halevi and T. Rabin. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 265–284.

Dwork, C. and A. Roth (Aug. 2014). "The Algorithmic Foundations of Differential Privacy". In: *Foundations and Trends® in Theoretical Computer Science* 9.3–4, pp. 211–407.

Eckersley, P. (July 2010). "How Unique Is Your Web Browser?" In: *Proceedings of the 10th International Conference on Privacy Enhancing Technologies*. PETS'10. Berlin, Heidelberg: Springer-Verlag, pp. 1–18.

Electricity Dataset (2022).

Erlingsson, Ú., V. Pihur, and A. Korolova (Nov. 2014). "RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response". In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. CCS '14. New York, NY, USA: Association for Computing Machinery, pp. 1054–1067.

van Erven, T. and P. Harremos (July 2014). "Rényi Divergence and Kullback-Leibler Divergence". In: *IEEE Transactions on Information Theory* 60.7, pp. 3797–3820.

Fang, H., Z. Fan, Y. Sun, and M. Friedlander (June 2020). "Greed Meets Sparsity: Understanding and Improving Greedy Coordinate Descent for Sparse Optimization". In: *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 434–444.

Farrand, T., F. Mireshghallah, S. Singh, and A. Trask (Nov. 2020). "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*. PPMLP'20. New York, NY, USA: Association for Computing Machinery, pp. 15–19.

Feldman, V., T. Koren, and K. Talwar (June 2020). "Private Stochastic Convex Optimization: Optimal Rates in Linear Time". In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. New York, NY, USA: Association for Computing Machinery, pp. 439–449.

Fercoq, O., A. Gramfort, and J. Salmon (June 2015). "Mind the Duality Gap: Safer Rules for the Lasso". In: *Proceedings of the 32nd International Conference on Machine Learning*. PMLR, pp. 333–342.

Fercoq, O. and P. Richtárik (Mar. 2014). "Accelerated, Parallel and Proximal Coordinate Descent". In: *arXiv:1312.5799 [cs, math, stat]*.

Fernandes, N., Y. Kawamoto, and T. Murakami (Oct. 2021). "Locality Sensitive Hashing with Extended Differential Privacy". In: *Computer Security – ESORICS 2021: 26th European Symposium on Research in Computer Security, Darmstadt, Germany, October 4–8, 2021, Proceedings, Part II*. Berlin, Heidelberg: Springer-Verlag, pp. 563–583.

Fioretto, F., C. Tran, P. V. Hentenryck, and K. Zhu (July 2022). "Differential Privacy and Fairness in Decisions and Learning Tasks: A Survey". In: *Thirty-First International Joint Conference on Artificial Intelligence*. Vol. 6, pp. 5470–5477.

Fleming, W. (Dec. 2012). *Functions of Several Variables*. Springer Science & Business Media.

Fowl, L. H., J. Geiping, W. Czaja, M. Goldblum, and T. Goldstein (Jan. 2022). "Robbing the Fed: Directly Obtaining Private Data in Federated Learning with Modified Models". In: *International Conference on Learning Representations*.

Frank, M. and P. Wolfe (Mar. 1956). "An Algorithm for Quadratic Programming". In: *Naval Research Logistics Quarterly* 3.1-2, pp. 95–110.

Friedman, J., T. Hastie, and R. Tibshirani (2010). "Regularization Paths for Generalized Linear Models via Coordinate Descent". In: *Journal of statistical software* 33.1, pp. 1–22.

Garling, D. J. H. (2014). *A Course in Mathematical Analysis: Volume 2: Metric and Topological Spaces, Functions of a Vector Variable*. Vol. 2. Cambridge: Cambridge University Press.

Garrigos, G. and R. M. Gower (Feb. 2023). *Handbook of Convergence Theorems for (Stochastic) Gradient Methods*.

GDPR (Apr. 2016). *Règlement (UE) 2016/679 du Parlement européen et du Conseil du 27 avril 2016 relatif à la protection des personnes physiques à l'égard du traitement des données à caractère personnel et à la libre circulation de ces données, et abrogeant la directive 95/46/CE (règlement général sur la protection des données) (Texte présentant de l'intérêt pour l'EEE)*.

Geiping, J., H. Bauermeister, H. Dröge, and M. Moeller (Sept. 2020). "Inverting Gradients – How Easy Is It to Break Privacy in Federated Learning?" In: *arXiv:2003.14053 [cs]*.

Gorbunov, E., F. Hanzely, and P. Richtarik (June 2020). "A Unified Theory of SGD: Variance Reduction, Sampling, Quantization and Coordinate Descent". In: *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 680–690.

Gower, R. M., M. Schmidt, F. Bach, and P. Richtárik (Nov. 2020). "Variance-Reduced Methods for Machine Learning". In: *Proceedings of the IEEE* 108.11, pp. 1968–1983.

Gower, R. M., N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtárik (May 2019). "SGD: General Analysis and Improved Rates". In: *International Conference on Machine Learning*. PMLR, pp. 5200–5209.

Guo, C., B. Karrer, K. Chaudhuri, and L. van der Maaten (June 2022). "Bounding Training Data Reconstruction in Private (Deep) Learning". In: *Proceedings of the 39th International Conference on Machine Learning*. PMLR, pp. 8056–8071.

Hadamard, J. (1908). *Mémoire Sur Le Problème d'analyse Relatif à l'équilibre Des Plaques Élastiques Encastrées*. Mémoires Présentés Par Divers Savants à l'Académie Des Sciences de l'Institut de France, Éxtrait Du Tome XXXIII. Paris: Imprimerie nationale.

Hanzely, F., D. Kovalev, and P. Richtarik (Feb. 2020). "Variance Reduced Coordinate Descent with Acceleration: New Method With a Surprising Application to Finite-Sum Problems". In: *arXiv:2002.04670 [cs, math]*.

Hardt, M., E. Price, E. Price, and N. Srebro (2016). "Equality of Opportunity in Supervised Learning". In: *Advances in Neural Information Processing Systems*. Vol. 29. Curran Associates, Inc.

Hardt, M. and K. Talwar (June 2010). "On the Geometry of Differential Privacy". In: *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*. STOC '10. New York, NY, USA: Association for Computing Machinery, pp. 705–714.

Hawes, M. (2021). *Understanding the 2020 Census Disclosure Avoidance System:*

Hendrikx, H., P. Mangold, and A. Bellet (2023). "The Relative Gaussian Mechanism and its Application to Private Gradient Descent". In: *arXiv preprint arXiv:2308.15250*.

Himmelblau, D. M. (M. (1972). *Applied Nonlinear Programming*. New York: McGraw-Hill.

Holland, P. W. and R. E. Welsch (Jan. 1977). "Robust Regression Using Iteratively Reweighted Least-Squares". In: *Communications in Statistics - Theory and Methods* 6.9, pp. 813–827.

Hu, H., Z. Salcic, L. Sun, G. Dobbie, P. S. Yu, and X. Zhang (Sept. 2022a). "Membership Inference Attacks on Machine Learning: A Survey". In: *ACM Computing Surveys* 54.11s, 235:1–235:37.

Hu, L., S. Ni, H. Xiao, and D. Wang (June 2022b). "High Dimensional Differentially Private Stochastic Optimization with Heavy-tailed Data". In: *Proceedings of the 41st ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*. PODS '22. New York, NY, USA: Association for Computing Machinery, pp. 227–236.

Huang, L. and N. Vishnoi (May 2019). "Stable and Fair Classification". In: *Proceedings of the 36th International Conference on Machine Learning*. PMLR, pp. 2879–2890.

Jaggi, M. (Feb. 2013). "Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization". In: *International Conference on Machine Learning*. PMLR, pp. 427–435.

Jagielski, M., M. Kearns, J. Mao, A. Oprea, A. Roth, S. Sharifi-Malvajerdi, and J. Ullman (May 2019). "Differentially Private Fair Learning". In: *arXiv:1812.02696 [cs, stat]*.

Jain, P., P. Kothari, and A. Thakurta (June 2012). "Differentially Private Online Learning". In: *Proceedings of the 25th Annual Conference on Learning Theory*. JMLR Workshop and Conference Proceedings, pp. 24.1–24.34.

Jayaraman, B., L. Wang, K. Knipmeyer, Q. Gu, and D. Evans (Jan. 2021). "Revisiting Membership Inference Under Realistic Assumptions". In: *arXiv:2005.10881 [cs, stat]*.

Johnson, R. and T. Zhang (2013). "Accelerating Stochastic Gradient Descent Using Predictive Variance Reduction". In: *Advances in Neural Information Processing Systems*. Ed. by C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger. Vol. 26. Curran Associates, Inc.

Kairouz, P., M. R. Diaz, K. Rush, and A. Thakurta (July 2021). "(Nearly) Dimension Independent Private ERM with AdaGrad Rates via Publicly Estimated Subspaces". In: *Proceedings of Thirty Fourth Conference on Learning Theory*. PMLR, pp. 2717–2746.

Kairouz, P., S. Oh, and P. Viswanath (2014). "Extremal Mechanisms for Local Differential Privacy". In: *Advances in Neural Information Processing Systems*. Vol. 27. Curran Associates, Inc.

Kantorovich, L. V. and G. P. Akilov (Jan. 1982). *Functional Analysis*. 2nd edition. Oxford ; New York: Pergamon.

Karimi, H., J. Nutini, and M. Schmidt (2016). "Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Łojasiewicz Condition". In: *Machine Learning and Knowledge Discovery in Databases*. Ed. by P. Frasconi, N. Landwehr, G. Manco, and J. Vreeken. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 795–811.

Karimireddy, S. P., A. Koloskova, S. U. Stich, and M. Jaggi (Apr. 2019). "Efficient Greedy Coordinate Descent for Composite Problems". In: *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 2887–2896.

Kasiviswanathan, S. P. and A. Smith (June 2014). "On the 'Semantics' of Differential Privacy: A Bayesian Formulation". In: *Journal of Privacy and Confidentiality* 6.1.

Kasiviswanathan, S. P. and H. Jin (2016). "Efficient Private Empirical Risk Minimization for High-dimensional Learning". In: p. 10.

Kelley Pace, R. and R. Barry (May 1997). "Sparse Spatial Autoregressions". In: *Statistics & Probability Letters* 33.3, pp. 291–297.

Khaled, A., K. Mishchenko, and C. Jin (May 2023). *DoWG Unleashed: An Efficient Universal Parameter-Free Gradient Descent Method*.

Khaled, A., O. Sebbouh, N. Loizou, R. M. Gower, and P. Richtárik (June 2020). *Unified Analysis of Stochastic Gradient Methods for Composite Convex and Smooth Optimization*.

Kiefer, J. (1953). "Sequential Minimax Search for a Maximum". In: *Proceedings of the American mathematical society* 4.3, pp. 502–506.

Kifer, D., A. Smith, and A. Thakurta (June 2012). "Private Convex Empirical Risk Minimization and High-dimensional Regression". In: *Proceedings of the 25th Annual Conference on Learning Theory*. JMLR Workshop and Conference Proceedings, pp. 25.1–25.40.

Kilbertus, N., A. Gascon, M. Kusner, M. Veale, K. Gummadi, and A. Weller (July 2018). "Blind Justice: Fairness with Encrypted Sensitive Attributes". In: *Proceedings of the 35th International Conference on Machine Learning*. PMLR, pp. 2630–2639.

Klopfenstein, Q., Q. Bertrand, A. Gramfort, J. Salmon, and S. Vaiter (Oct. 2020). *Model Identification and Local Linear Convergence of Coordinate Descent*.

Koloskova, A., H. Hendrikx, and S. U. Stich (May 2023). *Revisiting Gradient Clipping: Stochastic Bias and Tight Convergence Guarantees*.

Koskela, A. and T. Kulkarni (June 2023). *Practical Differentially Private Hyperparameter Tuning with Subsampling*.

Kusner, M. J., J. Loftus, C. Russell, and R. Silva (2017). "Counterfactual Fairness". In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc.

Lamer, A., A. Filiot, Y. Bouillard, P. Mangold, P. Andrey, and J. Schiro (May 2021). "Specifications for the Routine Implementation of Federated Learning in Hospitals Networks". In: *Studies in Health Technology and Informatics*. Vol. 281, pp. 128–132.

Langlois, A., D. Stehlé, and R. Steinfeld (2014). "GGHLite: More Efficient Multilinear Maps from Ideal Lattices". In: *Advances in Cryptology – EUROCRYPT 2014*. Ed. by P. Q. Nguyen and E. Oswald. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 239–256.

Laperdrix, P., W. Rudametkin, and B. Baudry (May 2016). "Beauty and the Beast: Diverting Modern Web Browsers to Build Unique Browser Fingerprints". In: *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 878–894.

Lewis, A. S. and S. J. Wright (July 2016). "A Proximal Method for Composite Minimization". In: *Mathematical Programming* 158.1, pp. 501–546.

Li, N., T. Li, and S. Venkatasubramanian (Apr. 2007). "T-Closeness: Privacy Beyond k-Anonymity and l-Diversity". In: *2007 IEEE 23rd International Conference on Data Engineering*, pp. 106–115.

Li, N., W. Qardaji, and D. Su (May 2012). "On Sampling, Anonymization, and Differential Privacy or, k-Anonymization Meets Differential Privacy". In: *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*. ASIACCS '12. New York, NY, USA: Association for Computing Machinery, pp. 32–33.

Liu, B., M. Ding, S. Shaham, W. Rahayu, F. Farokhi, and Z. Lin (Mar. 2021). "When Machine Learning Meets Privacy: A Survey and Outlook". In: *ACM Computing Surveys* 54.2, 31:1–31:36.

Liu, H., M. Palatucci, and J. Zhang (June 2009). "Blockwise Coordinate Descent Procedures for the Multi-Task Lasso, with Applications to Neural Semantic Basis Discovery". In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ICML '09. New York, NY, USA: Association for Computing Machinery, pp. 649–656.

Liu, J. and K. Talwar (June 2019). "Private Selection from Private Candidates". In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2019. New York, NY, USA: Association for Computing Machinery, pp. 298–309.

Liu, Y., Y. Kang, X. Zhang, L. Li, Y. Cheng, T. Chen, M. Hong, and Q. Yang (July 2020). *A Communication Efficient Collaborative Learning Framework for Distributed Features*.

Liu, Z., P. Luo, X. Wang, and X. Tang (2015). "Deep Learning Face Attributes in the Wild". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3730–3738.

Lohaus, M., M. Perrot, and U. V. Luxburg (Nov. 2020). "Too Relaxed to Be Fair". In: *Proceedings of the 37th International Conference on Machine Learning*. PMLR, pp. 6360–6369.

Lowy, A. and M. Razaviyayn (Feb. 2021). *Output Perturbation for Differentially Private Convex Optimization with Improved Population Loss Bounds, Runtimes and Applications to Private Adversarial Training*.

Luo, Z.-Q. and P. Tseng (Jan. 1992). "On the Convergence of the Coordinate Descent Method for Convex Differentiable Minimization". In: *Journal of Optimization Theory and Applications* 72.1, pp. 7–35.

Machanavajjhala, A., D. Kifer, J. Gehrke, and M. Venkitasubramaniam (Mar. 2007). "L-Diversity: Privacy beyond k-Anonymity". In: *ACM Transactions on Knowledge Discovery from Data* 1.1, 3–es.

Mahalanobis, C. (1936). "On the Generalised Distance in Statistics". In: *Proceedings of the National Institute of Sciences of India* 2.1, pp. 49–55.

Maheshwari, G. and M. Perrot (June 2022). *FairGrad: Fairness Aware Gradient Descent*.

Mangold, P., A. Bellet, J. Salmon, and M. Tommasi (Oct. 2021). "Differentially Private Coordinate Descent for Composite Empirical Risk Minimization". In: *arXiv:2110.11688 [cs, stat]*.

Mangold, P., A. Bellet, J. Salmon, and M. Tommasi (June 2022). "Differentially Private Coordinate Descent for Composite Empirical Risk Minimization". In: *Proceedings of the 39th International Conference on Machine Learning*. PMLR, pp. 14948–14978.

Mangold, P., A. Bellet, J. Salmon, and M. Tommasi (Apr. 2023a). "High-Dimensional Private Empirical Risk Minimization by Greedy Coordinate Descent". In: *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 4894–4916.

Mangold, P., A. Filiot, M. Moussa, V. Sobanski, G. Ficheur, P. Andrey, and A. Lamer (Nov. 2020). "A Decentralized Framework for Biostatistics and Privacy Concerns". In: *Studies in Health Technology and Informatics*. Ed. by A. Värri, J. Delgado, P. Gallos, M. Hägglund, K. Häyrinen, U.-M. Kinnunen, L. B. Pape-Haugaard, L.-M. Peltonen, K. Saranto, and P. Scott. IOS Press.

Mangold, P., M. Perrot, A. Bellet, and M. Tommasi (Jan. 2023b). "Differential Privacy Has Bounded Impact on Fairness in Classification". In: *Proceedings of the 40th International Conference on Machine Learning*. PMLR.

Martinet, B. (1970). "R'egularisation d"in'equations Variationnelles Par Approximations Successives". In.

Martinet, B. (1972). "D{\'e}termination Approch{\'e}e d'un Point Fixe d'une Application Pseudo-Contractante". In: *CR Acad. Sci. Paris* 274.2, pp. 163–165.

Massias, M., A. Gramfort, and J. Salmon (May 2017). *From Safe Screening Rules to Working Sets for Faster Lasso-type Solvers*.

Massias, M., A. Gramfort, and J. Salmon (July 2018). "Celer: A Fast Solver for the Lasso with Dual Extrapolation". In: *Proceedings of the 35th International Conference on Machine Learning*. PMLR, pp. 3315–3324.

McMahan, B., E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas (Apr. 2017). "Communication-Efficient Learning of Deep Networks from Decentralized Data". In: *Artificial Intelligence and Statistics*. PMLR, pp. 1273–1282.

Mehrabi, N., F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan (July 2021). "A Survey on Bias and Fairness in Machine Learning". In: *ACM Computing Surveys* 54.6, 115:1–115:35.

Melis, L., C. Song, E. De Cristofaro, and V. Shmatikov (May 2019). "Exploiting Unintended Feature Leakage in Collaborative Learning". In: *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 691–706.

Miller, F. (1882). *Telegraphic Code to Insure Privacy and Secrecy in the Transmission of Telegrams*.

Mironov, I. (Aug. 2017). "Renyi Differential Privacy". In: *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pp. 263–275.

Mironov, I., K. Talwar, and L. Zhang (Aug. 2019). "Rényi Differential Privacy of the Sampled Gaussian Mechanism". In: *arXiv:1908.10530 [cs, stat]*.

Mishchenko, K. and A. Defazio (June 2023). *Prodigy: An Expeditiously Adaptive Parameter-Free Learner*.

Mohapatra, S., S. Sasy, X. He, G. Kamath, and O. Thakkar (June 2022). "The Role of Adaptive Optimizers for Honest Private Hyperparameter Selection". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 36.7, pp. 7806–7813.

Mozannar, H., M. I. Ohannessian, and N. Srebro (July 2020). "Fair Learning with Private Demographic Data". In: *arXiv:2002.11651 [cs, stat]*.

Narayanan, A. and V. Shmatikov (Nov. 2007). *How To Break Anonymity of the Netflix Prize Dataset*.

Nasr, M., R. Shokri, and A. Houmansadr (May 2019). "Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning". In: *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 739–753.

Needell, D., N. Srebro, and R. Ward (Jan. 2016). "Stochastic Gradient Descent, Weighted Sampling, and the Randomized Kaczmarz Algorithm". In: *Mathematical Programming* 155.1-2, pp. 549–573.

Neel, S., A. Roth, G. Vietri, and S. Wu (Nov. 2020). "Oracle Efficient Private Non-Convex Optimization". In: *Proceedings of the 37th International Conference on Machine Learning*. PMLR, pp. 7243–7252.

Nesterov, Y. (2004). *Introductory Lectures on Convex Programming Volume I: Basic Course*.

Nesterov, Y. (Jan. 2010). "Efficiency of Coordinate Descent Methods on Huge-Scale Optimization Problems". In: *SIAM Journal on Optimization* 22.2, pp. 341–362.

Nesterov, Y. (2018). *Lectures on Convex Optimization*. Vol. 137. Springer Optimization and Its Applications. Cham: Springer International Publishing.

Nesterov, Y. and A. Nemirovskii (Jan. 1994). *Interior-Point Polynomial Algorithms in Convex Programming*. Studies in Applied and Numerical Mathematics. Society for Industrial and Applied Mathematics.

Nitanda, A. (Dec. 2014). "Stochastic Proximal Gradient Descent with Acceleration Techniques". In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*. NIPS'14. Cambridge, MA, USA: MIT Press, pp. 1574–1582.

Nutini, J., I. Laradji, and M. Schmidt (Dec. 2017). "Let's Make Block Coordinate Descent Go Fast: Faster Greedy Rules, Message-Passing, Active-Set Complexity, and Superlinear Convergence". In: *arXiv:1712.08859 [math]*.

Nutini, J., M. Schmidt, I. Laradji, M. Friedlander, and H. Koepke (June 2015). "Coordinate Descent Converges Faster with the Gauss-Southwell Rule Than Random Selection". In: *International Conference on Machine Learning*. PMLR, pp. 1632–1641.

Paige, B., J. Bell, A. Bellet, A. Gascón, and D. Ezer (May 2021). "Reconstructing Genotypes in Private Genomic Databases from Genetic Risk Scores". In: *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology* 28.5, pp. 435–451.

Papernot, N. and T. Steinke (Mar. 2022). *Hyperparameter Tuning with Renyi Differential Privacy*.

Parikh, N. and S. Boyd (Jan. 2014). "Proximal Algorithms". In: *Foundations and Trends in Optimization* 1.3, pp. 127–239.

Passty, G. B. (Dec. 1979). "Ergodic Convergence to a Zero of the Sum of Monotone Operators in Hilbert Space". In: *Journal of Mathematical Analysis and Applications* 72.2, pp. 383–390.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, and D. Cournapeau (2011). "Scikit-Learn: Machine Learning in Python". In: *MACHINE LEARNING IN PYTHON*, p. 6.

Phong, L. T., Y. Aono, T. Hayashi, L. Wang, and S. Moriai (2017). *Privacy-Preserving Deep Learning via Additively Homomorphic Encryption*.

Pichapati, V., A. T. Suresh, F. X. Yu, S. J. Reddi, and S. Kumar (Oct. 2019). "AdaCliP: Adaptive Clipping for Private SGD". In: *arXiv:1908.07643 [cs, stat]*.

Polyak, B. (Jan. 1977). "Subgradient Methods: A Survey of Soviet Research". In.

Polyak, B. T. (1987). *Introduction to Optimization*. 1 ed. Translations Series in Mathematics and Engineering. New York: Optimization Software, Inc.

Priyanshu, A., R. Naidu, F. Mireshghallah, and M. Malekzadeh (Aug. 2021). *Efficient Hyperparameter Optimization for Differentially Private Deep Learning*.

Pub, FIPS (1999). "Data Encryption Standard (Des)". In: *FIPS PUB*.

Pujol, D., R. McKenna, S. Kuppam, M. Hay, A. Machanavajjhala, and G. Miklau (Jan. 2020). "Fair Decision Making Using Privacy-Protected Data". In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. FAT* '20. New York, NY, USA: Association for Computing Machinery, pp. 189–199.

Richtárik, P. and M. Takáč (Apr. 2014). "Iteration Complexity of Randomized Block-Coordinate Descent Methods for Minimizing a Composite Function". In: *Mathematical Programming* 144.1-2, pp. 1–38.

Richtárik, P. and M. Takáč (Aug. 2016). "On Optimal Probabilities in Stochastic Coordinate Descent Methods". In: *Optimization Letters* 10.6, pp. 1233–1243.

Robbins, H. and S. Monro (1951). "A Stochastic Approximation Method". In: *The Annals of Mathematical Statistics* 22.3, pp. 400–407.

Rockafellar, R. T. (1970). *Convex Analysis*. Princeton Mathematical Series 28. Princeton, N.J: Princeton University Press.

Rockafellar, R. T. (Aug. 1976). "Monotone Operators and the Proximal Point Algorithm". In: *SIAM Journal on Control and Optimization* 14.5, pp. 877–898.

Rosasco, L., S. Villa, and B. C. Vũ (Dec. 2020). "Convergence of Stochastic Proximal Gradient Algorithm". In: *Applied Mathematics & Optimization* 82.3, pp. 891–917.

Ryu, E. K. and W. Yin (2022). *Large-Scale Convex Optimization: Algorithm Analysis via Monotone Operators*. Cambridge: Cambridge University Press.

Sablayrolles, A., M. Douze, C. Schmid, Y. Ollivier, and H. Jegou (May 2019). "White-Box vs Black-box: Bayes Optimal Strategies for Membership Inference". In: *Proceedings of the 36th International Conference on Machine Learning*. PMLR, pp. 5558–5567.

Sanyal, A., Y. Hu, and F. Yang (Aug. 2022). "How Unfair Is Private Learning?" In: *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*. PMLR, pp. 1738–1748.

Sardy, S., A. G. Bruce, and P. Tseng (June 2000). "Block Coordinate Relaxation Methods for Nonparametric Wavelet Denoising". In: *Journal of Computational and Graphical Statistics* 9.2, pp. 361–379.

Sason, I. and S. Verdú (Nov. 2016). "F -Divergence Inequalities". In: *IEEE Transactions on Information Theory* 62.11, pp. 5973–6006.

Schmidt, M., N. L. Roux, and F. Bach (Dec. 2011). "Convergence Rates of Inexact Proximal-Gradient Methods for Convex Optimization". In: *arXiv:1109.2415 [cs, math]*.

Shalev-Shwartz, S. (2011). "Online Learning and Online Convex Optimization". In: *Foundations and Trends® in Machine Learning* 4.2, pp. 107–194.

Shamir, A. (Nov. 1979). "How to Share a Secret". In: *Communications of the ACM* 22.11, pp. 612–613.

Shamir, O. and T. Zhang (Feb. 2013). "Stochastic Gradient Descent for Non-smooth Optimization: Convergence Results and Optimal Averaging Schemes". In: *International Conference on Machine Learning*. PMLR, pp. 71–79.

Shannon, C. E. (Oct. 1949). "Communication Theory of Secrecy Systems". In: *The Bell System Technical Journal* 28.4, pp. 656–715.

Shi, H.-J. M., S. Tu, Y. Xu, and W. Yin (Jan. 2017). "A Primer on Coordinate Descent Algorithms". In: *arXiv:1610.00040 [math, stat]*.

Shokri, R., M. Stronati, C. Song, and V. Shmatikov (May 2017). "Membership Inference Attacks Against Machine Learning Models". In: *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18.

Shor, N. Z. (1962). "Application of the Gradient Method for the Solution of Network Transportation Problems". In: *Scientific Seminar on Theory and Application of Cyber- netics and Operations Research, Academy of Sciences*.

Song, S., K. Chaudhuri, and A. D. Sarwate (Dec. 2013). "Stochastic Gradient Descent with Differentially Private Updates". In: *2013 IEEE Global Conference on Signal and Information Processing*. Austin, TX, USA: IEEE, pp. 245–248.

Steinke, T. (Oct. 2022). *Composition of Differential Privacy & Privacy Amplification by Subsampling*.

Stich, S. U., A. Raj, and M. Jaggi (July 2017). "Approximate Steepest Coordinate Descent". In: *Proceedings of the 34th International Conference on Machine Learning*. PMLR, pp. 3251–3259.

Sweeney, L. (2000). "Simple Demographics Often Identify People Uniquely". In: . *Pittsburgh*.

Sweeney, L. (Oct. 2002). "K-Anonymity: A Model for Protecting Privacy". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10.05, pp. 557–570.

Talwar, K., A. Guha Thakurta, and L. Zhang (2015). "Nearly Optimal Private LASSO". In: *Advances in Neural Information Processing Systems* 28.

Talwar, K., A. Thakurta, and L. Zhang (Nov. 2016). "Private Empirical Risk Minimization Beyond the Worst Case: The Effect of the Constraint Set Geometry". In: *arXiv:1411.5417 [cs, stat]*.

Tappenden, R., P. Richtárik, and J. Gondzio (July 2016). "Inexact Coordinate Descent: Complexity and Preconditioning". In: *Journal of Optimization Theory and Applications* 170.1, pp. 144–176.

Tardos, G. (Jan. 2008). "Optimal Probabilistic Fingerprint Codes". In: *Journal of the ACM (JACM)* 55.2, p. 10.

du Terrail, J. O., S.-S. Ayed, E. Cyffers, F. Grimberg, C. He, R. Loeb, P. Mangold, T. Marchand, O. Marfoq, E. Mushtaq, B. Muzellec, C. Philippenko, S. Silva, M. Teleńczuk, S. Albarqouni, S. Avestimehr, A. Bellet, A. Dieuleveut, M. Jaggi, S. P. Karimireddy, M. Lorenzi, G. Neglia, M. Tommasi, and M. Andreux (Oct. 2022). "FLamby: Datasets and Benchmarks for Cross-Silo Federated Learning in Realistic Healthcare Settings". In: *Thirty-Sixth Conference on Neural Information Processing Systems*.

Tibshirani, R. (1996). "Regression Shrinkage and Selection Via the Lasso". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1, pp. 267–288.

Tran, C., M. H. Dinh, and F. Fioretto (June 2021a). *Differentially Private Deep Learning under the Fairness Lens*.

Tran, C., F. Fioretto, and P. V. Hentenryck (May 2021b). "Differentially Private and Fair Deep Learning: A Lagrangian Dual Approach". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 35.11, pp. 9932–9939.

Truex, S., L. Liu, M. E. Gursoy, L. Yu, and W. Wei (2019). "Demystifying Membership Inference Attacks in Machine Learning as a Service". In: *IEEE Transactions on Services Computing*, pp. 1–1.

Tseng, P. (June 2001). "Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization". In: *Journal of Optimization Theory and Applications* 109.3, pp. 475–494.

Tseng, P. and S. Yun (Mar. 2009). "A Coordinate Gradient Descent Method for Nonsmooth Separable Minimization". In: *Mathematical Programming* 117.1, pp. 387–423.

Uniyal, A., R. Naidu, S. Kotti, S. Singh, P. J. Kenfack, F. Mireshghallah, and A. Trask (Mar. 2022). *DP-SGD vs PATE: Which Has Less Disparate Impact on Model Accuracy?*

Vanschoren, J., J. N. van Rijn, B. Bischl, and L. Torgo (June 2014). "OpenML: Networked Science in Machine Learning". In: *ACM SIGKDD Explorations Newsletter* 15.2, pp. 49–60.

Wang, D., M. Ye, and J. Xu (2017). "Differentially Private Empirical Risk Minimization Revisited: Faster and More General". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc.

Wang, L. and Q. Gu (Aug. 2019). "Differentially Private Iterative Gradient Hard Thresholding for Sparse Learning". In: *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. IJCAI'19. Macao, China: AAAI Press, pp. 3740–3747.

Wang, R., Y. F. Li, X. Wang, H. Tang, and X. Zhou (Nov. 2009). "Learning Your Identity and Disease from Research Papers: Information Leaks in Genome Wide Association Study". In: *Proceedings of the 16th ACM Conference on Computer and Communications Security*. CCS '09. New York, NY, USA: Association for Computing Machinery, pp. 534–544.

Wang, Y.-X., B. Balle, and S. P. Kasiviswanathan (Apr. 2019a). "Subsampled Renyi Differential Privacy and Analytical Moments Accountant". In: *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 1226–1235.

Wang, Z., M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi (Apr. 2019b). "Beyond Inferring Class Representatives: User-Level Privacy Leakage From Federated Learning". In: *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 2512–2520.

Warner, S. L. (Mar. 1965). "Randomized Response: A Survey Technique for Eliminating Evasive Answer Bias". In: *Journal of the American Statistical Association* 60.309, pp. 63–69.

Warren, S. D. and L. D. Brandeis (1890). "The Right to Privacy". In: *Harvard Law Review* 4.5, pp. 193–220.

Wen, Y., J. A. Geiping, L. Fowl, M. Goldblum, and T. Goldstein (June 2022). "Fishing for User Data in Large-Batch Federated Learning via Gradient Magnification". In: *Proceedings of the 39th International Conference on Machine Learning*. PMLR, pp. 23668–23684.

Wright, S. J. (June 2015). "Coordinate Descent Algorithms". In: *Mathematical Programming* 151.1, pp. 3–34.

Wright, S. J. and B. Recht (2022). *Optimization for Data Analysis*. Cambridge: Cambridge University Press.

Xiao, L. and T. Zhang (Jan. 2014). "A Proximal Stochastic Gradient Method with Progressive Variance Reduction". In: *SIAM Journal on Optimization* 24.4, pp. 2057–2075.

Xu, D., W. Du, and X. Wu (Sept. 2020). "Removing Disparate Impact of Differentially Private Stochastic Gradient Descent on Model Accuracy". In: *arXiv:2003.03699 [cs, stat]*.

Xu, D., S. Yuan, and X. Wu (May 2019). "Achieving Differential Privacy and Fairness in Logistic Regression". In: *Companion Proceedings of The 2019 World Wide Web Conference*. New York, NY, USA: Association for Computing Machinery, pp. 594–599.

Yang, X., H. Zhang, W. Chen, and T.-Y. Liu (June 2022). *Normalized/Clipped SGD with Perturbation for Differentially Private Non-Convex Optimization*.

Ye, J., A. Maddi, S. K. Murakonda, and R. Shokri (Jan. 2022). *Enhanced Membership Inference Attacks against Machine Learning Models*.

Yeom, S., I. Giacomelli, M. Fredrikson, and S. Jha (July 2018). "Privacy Risk in Machine Learning: Analyzing the Connection to Overfitting". In: *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*, pp. 268–282.

Yousefpour, A., I. Shilov, A. Sablayrolles, D. Testuggine, K. Prasad, M. Malek, J. Nguyen, S. Ghosh, A. Bharadwaj, J. Zhao, G. Cormode, and I. Mironov (Aug. 2022). *Opacus: User-Friendly Differential Privacy Library in PyTorch*.

Yuan, G.-X., K.-W. Chang, C.-J. Hsieh, and C.-J. Lin (Dec. 2010). "A Comparison of Optimization Methods and Software for Large-scale L1-regularized Linear Classification". In: *The Journal of Machine Learning Research* 11, pp. 3183–3234.

Zafar, M. B., I. Valera, M. Gomez Rodriguez, and K. P. Gummadi (Apr. 2017). "Fairness Beyond Disparate Treatment & Disparate Impact: Learning Classification without Disparate Mistreatment". In: *Proceedings of the 26th International Conference on World Wide Web*. WWW '17. Republic and Canton of Geneva,

CHE: International World Wide Web Conferences Steering Committee, pp. 1171–1180.

Zhou, Y., Z. S. Wu, and A. Banerjee (2021). "Bypassing the Ambiant Dimension: Private SGD with Gradient Subspace Identification". In: p. 28.

Zhu, L., Z. Liu, and S. Han (2019). "Deep Leakage from Gradients". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc.

Appendix A

Proofs of Chapter 4

A.1 Lemmas on Sensitivity

In this section, we let $\mathcal{X}$ be the universe where the data is drawn from. To upper bound the sensitivities of a function's gradient, we start by recalling in Lemma A.1.1 that (coordinate) gradients are bounded by (coordinate-wise-)Lipschitz constants. We then link this upper bound with gradients' sensitivities in Lemma A.1.2.

Lemma A.1.1. *Let $\ell : \mathbb{R}^p \times \mathcal{X} \to \mathbb{R}$ be convex and differentiable in its first argument, $\Lambda > 0$ and $L_1, \dots, L_p > 0$.*

1. *If $\ell(\cdot; d)$ is Λ-Lipschitz for all $d \in \mathcal{X}$, then $\|\nabla \ell(w; d)\|_2 \leq \Lambda$ for all $w \in \mathbb{R}^p$ and $d \in \mathcal{X}$.*
2. *If $\ell(\cdot; d)$ is L-coordinate-Lipschitz for all $d \in \mathcal{X}$, then $|\nabla_j \ell(w; d)| \leq L_j$ for all $w \in \mathbb{R}^p$, $d \in \mathcal{X}$ and $j \in [p]$.*

Proof. Let $d \in \mathcal{X}$. We start by proving the first statement. First, if $\nabla \ell(w; d) = 0$, $\|\nabla \ell(w; d)\|_2 = 0 \leq \Lambda$ and the result holds. Second, we focus on the case where $\nabla \ell(w; d) \neq 0$. The convexity of ℓ gives, for $w \in \mathbb{R}^p$, $d \in \mathcal{X}$:

$$\begin{aligned} \ell(w + \nabla \ell(w; d); d) &\geq \ell(w; d) + \langle \nabla \ell(w; d), \nabla \ell(w; d) \rangle \\ &= \ell(w; d) + \|\nabla \ell(w; d)\|_2^2 \ , \end{aligned} \tag{A.1.1}$$

then, reorganizing the terms and using Λ-Lipschitzness of ℓ yields

$$\begin{aligned} \|\nabla \ell(w; d)\|_2^2 &\leq \ell(w + \nabla \ell(w; d); d) - \ell(w; d) \\ &\leq |\ell(w + \nabla \ell(w; d); d) - \ell(w; d)| \\ &\leq \Lambda \|\nabla \ell(w; d)\|_2 \ , \end{aligned} \tag{A.1.2}$$

and the result follows after dividing by $\|\nabla\ell(w;d)\|_2$. To prove the second statement, we set $j \in [p]$, and $w \in \mathbb{R}^p$, and remark that if $\nabla_j\ell(w;d) = 0$, then $|\nabla_j\ell(w;d)| \leq L_j$. When $\nabla_j\ell(w;d) \neq 0$, the convexity of ℓ yields

$$\begin{aligned}\ell(w + \nabla_j\ell(w;d)e_j;d) &\geq \ell(w;d) + \langle\nabla\ell(w;d), \nabla_j\ell(w;d)e_j\rangle \\ &= \ell(w;d) + \nabla_j\ell(w;d)^2 \ .\end{aligned} \tag{A.1.3}$$

Reorganizing the terms and using L-coordinate-Lipschitzness of ℓ gives

$$\begin{aligned}\nabla_j\ell(w;d)^2 &\leq \ell(w + \nabla_j\ell(w;d)e_j;d) - \ell(w;d) \\ &\leq |\ell(w + \nabla_j\ell(w;d)e_j;d) - \ell(w;d)| \\ &\leq L_j|\nabla_j\ell(w;d)| \ ,\end{aligned} \tag{A.1.4}$$

and we get the result after dividing by $|\nabla_j\ell(w;d)|$. □

Lemma A.1.2. *Let $\ell : \mathbb{R}^p \times \mathcal{X} \to \mathbb{R}$ be convex and differentiable in its 1st argument, $\Lambda > 0$ and $L_1, \ldots, L_p > 0$.*

1. *If $\ell(\cdot;d)$ is Λ-Lipschitz for all $d \in \mathcal{X}$, then $\Delta(\nabla\ell) \leq 2\Lambda$.*
2. *If $\ell(\cdot;d)$ is L-coordinate-Lipschitz for all $d \in \mathcal{X}$, then $\Delta(\nabla_j\ell) \leq L_j$ for all $j \in [p]$.*

Proof. We start by proving the first statement. Let $w, w' \in \mathbb{R}^p$, $d, d' \in \mathcal{X}$. From the triangle inequality and Lemma A.1.1, we get the following upper bounds:

$$\|\nabla\ell(w;d) - \nabla\ell(w';d')\|_2 \leq |\nabla\ell(w;d)| + |\nabla\ell(w';d')| \leq 2\Lambda \ , \tag{A.1.5}$$

which is the claim of the first statement. To prove the second statement, we proceed similarly: the triangle inequality and Lemma A.1.1 give the following upper bounds:

$$|\nabla_j\ell(w;d) - \nabla_j\ell(w';d')| \leq |\nabla_j\ell(w;d)| + |\nabla_j\ell(w';d')| \leq 2L_j \ , \tag{A.1.6}$$

which is the desired result. □

We can therefore obtain a bound on the sum of the sensitivities of the functions $d \mapsto 1/M_j\nabla_j\ell(w;d)^2$ for $j \in [p]$, $w \in \mathbb{R}^p$. We denote this sum $\Delta_{M^{-1}}(\nabla\ell)^2$.

Corollary A.1.1. *Let $L_1, \ldots, L_p > 0$. Let $\ell(\cdot;d) : \mathbb{R}^p \to \mathbb{R}$ be a convex, L-coordinate-Lipschitz function for all $d \in \mathcal{X}$. Then*

$$\Delta_{M^{-1}}(\nabla\ell) = \Big(\sum_{j=1}^{p} \frac{1}{M_j}\Delta(\nabla_j\ell)^2\Big)^{\frac{1}{2}} \leq \Big(\sum_{j=1}^{p} \frac{4}{M_j}L_j^2\Big)^{\frac{1}{2}} = 2\|L\|_{M^{-1}} \ . \tag{A.1.7}$$

A.2 Proof of Theorem 4.3.1

To track the privacy loss of an adaptive composition of K Gaussian mechanisms, we use Rényi Differential Privacy (Mironov, 2017, RDP). We note that similar results are obtained with zero Concentrated Differential Privacy (Bun and Steinke, 2016). This flavor of differential privacy, gives tighter privacy guarantees in that setting, as it reduces the noise variance by a multiplicative factor of $\log(K/\delta)$ in comparison to the usual advanced composition theorem of differential privacy (Dwork et al., 2006). Importantly, RDP can be translated back to differential privacy.

In this section, we recall the definition and main properties of zCDP. We denote by $\mathcal{D}$ the set of all datasets over a universe $\mathcal{X}$ and by $\mathcal{F}$ the set of possible outcomes of the randomized algorithms we consider.

A.2.1 Rényi Differential Privacy

We will use the Rényi divergence (Definition A.2.1), which gives a distribution-oriented vision of privacy.

Definition A.2.1 (Rényi divergence, van Erven and Harremos 2014)**.** *For two random variables Y and Z with values in the same domain $\mathcal{C}$, the Rényi divergence is, for $\alpha > 1$,*

$$D_\alpha(Y||Z) = \frac{1}{\alpha - 1} \log \int_{\mathcal{C}} \mathbb{P}\, Y = z^\alpha \, \mathbb{P}\, Z = z^{1-\alpha} dz \ . \tag{A.2.1}$$

We now define RDP in Definition A.2.2. RDP provides a strong privacy guarantee that can be converted to classical differential privacy (Lemma A.2.1 and Corollary A.2.1).

Definition A.2.2 (Rényi Differential Privacy, Mironov 2017)**.** *A randomized algorithm $\mathcal{A} : \mathcal{D} \to \mathcal{F}$ is (α, ϵ)-Rényi-differentially private (RDP) if, for all all datasets $D, D' \in \mathcal{D}$ differing on at most one element,*

$$D_\alpha(\mathcal{A}(D)||\mathcal{A}(D')) \leq \epsilon \ . \tag{A.2.2}$$

Lemma A.2.1 (Mironov 2017, Proposition 3)**.** *If a randomized algorithm $\mathcal{A} : \mathcal{D} \to \mathcal{F}$ is (α, ϵ)-RDP, then it is $(\epsilon + \frac{\log(1/\delta)}{\alpha - 1}, \delta)$-differentially private for all $0 < \delta < 1$.*

Remark A.2.1. *The above (α, ϵ)-RDP guarantees hold for multiple values of α, ϵ. As such, $\epsilon = \epsilon(\alpha)$ can be seen as a function of α, and Lemma A.2.1 ensures that the algorithm is (ϵ', δ)-DP for*

$$\epsilon' = \min_{\alpha > 1} \left\{ \epsilon(\alpha) + \frac{\log(1/\delta)}{\alpha - 1} \right\} \ . \tag{A.2.3}$$

We can now restate in Theorem A.2.1 the composition theorem of RDP, which is key in designing private iterative algorithms.

Theorem A.2.1 (Mironov 2017, Proposition 1)**.** *Let $\mathcal{A}_1, \dots, \mathcal{A}_K : \mathcal{D} \to \mathcal{F}$ be $K > 0$ randomized algorithms, such that for $1 \leq k \leq K$, $\mathcal{A}_k$ is $(\alpha, \epsilon_k(\alpha))$-RDP, where these algorithms can be chosen adaptively (i.e., $\mathcal{A}_k$ can use to the output of $\mathcal{A}_{k'}$ for all $k' < k$). Let $\mathcal{A} : \mathcal{D} \to \mathcal{F}^K$ such that for $D \in \mathcal{D}$, $\mathcal{A}(D) = (\mathcal{A}_1(D), \dots, \mathcal{A}_K(D))$. Then $\mathcal{A}$ is $\left(\alpha, \sum_{k=1}^K \epsilon_k(\alpha)\right)$-RDP.*

Finally, we define the Gaussian mechanism (Definition A.2.3), as used in Algorithm 4.3.1, and restate in Lemma A.2.2 the privacy guarantees that it satisfies in terms of RDP.

Definition A.2.3 (Gaussian mechanism)**.** *Let $f : \mathcal{D} \to \mathbb{R}^p$, $\sigma > 0$, and $D \in \mathcal{D}$. The Gaussian mechanism for answering the query f is defined as:*

$$\mathcal{M}_f^{Gauss}(D; \sigma) = f(D) + \mathcal{N}\left(0, \sigma^2 I_p\right) \quad . \tag{A.2.4}$$

Lemma A.2.2 (Mironov 2017, Corollary 3)**.** *The Gaussian mechanism with noise σ^2 is $(\alpha, \frac{\Delta(f)^2 \alpha}{2\sigma^2})$-RDP, where $\Delta(f) = \sup_{D,D'} \|f(D) - f(D')\|_2$ (for neighboring D, D') is the sensitivity of f.*

Proof. The function $h = \frac{f}{\Delta(f)}$ has sensitivity 1, thus for any $s > 0$, the Gaussian mechanism $\mathcal{M}_h^{Gauss}(\cdot; s)$ is $(\alpha, \frac{\alpha}{2\sigma^2})$-RDP (Mironov, 2017, Corollary 1). As $f = \Delta(f) \times h$, we have $\mathcal{M}_f^{Gauss}(\cdot; \sigma) = \Delta(f) \times \mathcal{M}_h^{Gauss}(\cdot; \frac{\sigma}{\Delta(f)})$. This mechanism is thus $(\alpha, \frac{\Delta(f)^2 \alpha}{2\sigma^2})$-RDP. □

Corollary A.2.1. *Let $0 < \epsilon \leq 1, 0 < \delta < \frac{1}{3}$. If a randomized algorithm $\mathcal{A} : \mathcal{D} \to \mathcal{F}$ is $(\alpha, \frac{\gamma\alpha}{2\sigma^2})$-RDP with $\gamma > 0$ and $\sigma = \frac{\sqrt{3\gamma \log(1/\delta)}}{\epsilon}$ for all $\alpha > 1$, it is also (ϵ, δ)-DP.*

Proof. From Remark A.2.1 it holds that $\mathcal{A}$ is (ϵ', δ)-DP with

$$\epsilon' = \min_{\alpha > 1} \left\{ \frac{\gamma\alpha}{2\sigma^2} + \frac{\log(1/\delta)}{\alpha - 1} \right\} \quad .$$

This minimum is attained when the derivative of the objective is zero, which is the case when $\frac{\gamma}{2\sigma^2} = \frac{\log(1/\delta)}{(\alpha-1)^2}$, resulting in $\alpha = 1 + \sqrt{\frac{2\log(1/\delta)\sigma^2}{\gamma}}$. $\mathcal{A}$ is thus (ϵ', δ)-DP with

$$\epsilon' = \frac{\gamma}{2\sigma^2} + \frac{\sqrt{\gamma \log(1/\delta)}}{\sqrt{2}\sigma} + \frac{\sqrt{\gamma \log(1/\delta)}}{\sqrt{2}\sigma} = \frac{\gamma}{2\sigma^2} + \frac{\sqrt{2\gamma \log(1/\delta)}}{\sigma} \quad . \tag{A.2.5}$$

Choosing $\sigma = \frac{\sqrt{3\gamma \log(1/\delta)}}{\epsilon}$ now gives

$$\epsilon' = \frac{\epsilon^2}{6\log(1/\delta)} + \sqrt{2/3}\epsilon \leq (1/6 + \sqrt{2/3})\epsilon \leq \epsilon \ , \tag{A.2.6}$$

where the first inequality comes from $\epsilon \leq 1$, thus $\epsilon^2 \leq \epsilon$ and $\delta < 1/3$ thus $\frac{1}{\log(1/\delta)} \leq 1$. The second inequality follows from $1/6 + \sqrt{2/3} \approx 0.983 < 1$. □

A.2.2 Proof of Theorem 4.3.1

We are now ready to prove Theorem 4.3.1. From the privacy perspective, Algorithm 4.3.1 adaptively releases and post-processes a series of gradient coordinates protected by the Gaussian mechanism. We thus start by proving Lemma A.2.3, which gives an (ϵ, δ)-differential privacy guarantee for the adaptive composition of K Gaussian mechanisms.

Lemma A.2.3. *Let $0 < \epsilon \leq 1$, $\delta < 1/3$, $K > 0$, $p > 0$, and $\{f_k : \mathbb{R}^p \to \mathbb{R}\}_{k=1}^{k=K}$ a family of K functions. The adaptive composition of K Gaussian mechanisms, with the k-th mechanism releasing f_k with noise scale $\sigma_k = \frac{\Delta(f_k)\sqrt{3K\log(1/\delta)}}{\epsilon}$ is (ϵ, δ)-differentially private.*

Proof. Let $\sigma > 0$. Lemma A.2.2 guarantees that the k-th Gaussian mechanism with noise scale $\sigma_k = \Delta(f_k)\sigma > 0$ is $(\alpha, \frac{\alpha}{2\sigma^2})$-RDP. Then, the composition of these K mechanisms is, according to Theorem A.2.1, $(\alpha, \frac{k\alpha}{2\sigma^2})$-RDP. This can be converted to (ϵ, δ)-DP via Corollary A.2.1 with $\gamma = K$, which gives $\sigma_k = \frac{\Delta(f_k)\sqrt{3k\log(1/\delta)}}{\epsilon}$ for $k \in [K]$. □

We now restate Theorem 4.3.1 and prove it.

Theorem A.2.2. *4.3.1 Assume $\ell(\cdot; d)$ is L-coordinate-Lipschitz $\forall d \in \mathcal{X}$. Let $\epsilon < 1$ and $\delta < 1/3$. If $\sigma_j^2 = \frac{12L_j^2 TK\log(1/\delta)}{n^2\epsilon^2}$ for all $j \in [p]$, then Algorithm 4.3.1 satisfies (ϵ, δ)-DP.*

Proof. For $j \in [1, p]$, $\nabla_j f$ in Algorithm 4.3.1 is released using the Gaussian mechanism with noise variance σ_j^2. The sensitivity of $\nabla_j f$ is $\Delta(\nabla_j f) = \frac{\Delta(\nabla_j \ell)}{n} \leq \frac{2L_j}{n}$. Note that TK gradients are released, and

$$\sigma_j^2 = \frac{12L_j^2 TK\log(1/\delta)}{n^2\epsilon^2} \text{ for } j \in [1, p] \ ,$$

thus by Lemma A.2.3 and the post-processing property of DP, Algorithm 4.3.1 is (ϵ, δ)-differentially private. □

A.3 Proof of Utility (Theorem 4.3.2)

A.3.1 Problem Statement

Let $D \in \mathcal{X}^n$ be a dataset of n elements drawn from a universe $\mathcal{X}$. Recall that we consider the following composite empirical risk minimization problem:

$$w^* \in \arg\min_{w \in \mathbb{R}^p} \left\{ F(w; D) = \underbrace{\frac{1}{n} \sum_{i=1}^{n} \ell(w; d_i)}_{=: f(w; D)} + \psi(w) \right\} , \tag{A.3.1}$$

where $\ell(\cdot, d)$ is convex, L-coordinate-Lipschitz, and M-coordinate-smooth for all $d \in \mathcal{X}$, and $\psi(w) = \sum_{j=1}^{p} \psi_j(w_j)$ is convex and separable. We denote by F the complete objective function, and by f its smooth part. For readability, we omit the dependence on their second argument (*i.e.,* the data) in the rest of this section.

A.3.2 Proof of Theorem 4.3.2

In this section, we prove our central theorem that guarantees the utility of the DP-CD algorithm. To this end, we start by proving a lemma that upper bounds the expected value of $F(\theta^{k+1})$ in Algorithm 4.3.1. Using this lemma, we prove sub-linear convergence for the inner loop of DP-CD. This gives the sub-linear convergence of our algorithm for convex losses. Under the additional hypothesis that F is strongly convex, we show that iterates of the outer loop of DP-CD converge linearly towards the (unique) minimum of F.

We recall that in Algorithm 4.3.1, iterates of the inner loop are denoted by $\theta_1, \ldots, \theta_K$, and those of the outer loop by $\bar{w}_1, \ldots, \bar{w}_T$, with $\bar{w}_t = \frac{1}{K} \sum_{k=1}^{K} \theta^k$ for $t > 0$. Algorithm 4.3.1 is randomized in two ways: when choosing the coordinate to update and when drawing noise. For convenience, we denote by $\mathbb{E}_j[\cdot]$ the expectation *w.r.t.,* the choice of coordinate, by $\mathbb{E}_\eta[\cdot]$ the one *w.r.t.,* the noise, and by $\mathbb{E}_{j,\eta}[\cdot]$ the expectation *w.r.t.,* both. When no subscript is used, the expectation is taken over all random variables. We will also use the notation $\mathbb{E}_{j,\eta}[\cdot|\theta_k]$ for the conditional expectation of a random variable, given a realization of θ_k.

A.3.2 (a) Descent Lemma

We begin by proving Lemma A.3.1, which decomposes the change of a function F when updating its argument $\theta \in \mathbb{R}^p$, in relation to a vector $w \in \mathbb{R}^p$, into two parts: one that remains fixed, corresponding to the unchanged entries of θ, and a second part corresponding to the objective decrease due to the update. At this point, the vector w is arbitrary, but we will later choose w to be a minimizer of F, that is a solution to (A.3.1).

Lemma A.3.1. *Let ℓ, f, ψ, and F be defined as in Section A.3.1. Take a random variable $\theta \in \mathbb{R}^p$ and two arbitrary vectors $w, g \in \mathbb{R}^p$. Let a random variable j, taking its values uniformly randomly in $[p]$, Choose $\gamma_1, \ldots, \gamma_p > 0$ and $\Gamma = \operatorname{diag}(\gamma_1, \ldots, \gamma_p)$. It holds that*

$$\mathbb{E}_j[F(\theta - \gamma_j g_j e_j) - F(w)|\theta] - \frac{p-1}{p}(F(\theta) - F(w)) \\ \leq \frac{1}{p}\left(f(\theta) - f(w) + \langle \nabla f(\theta), -\Gamma g\rangle + \frac{1}{2}\|\Gamma g\|_M^2 + \psi(\theta - \Gamma g) - \psi(w)\right) \quad . \tag{A.3.2}$$

Remark A.3.1. *To avoid notational clutter, we will write $\gamma_j g_j$ instead of $\gamma_j g_j e_j$ throughout this section.*

Proof. We start the proof by finding an upper bound on $\mathbb{E}_j[F(\theta - \gamma_j g_j e_j) - F(w)|\theta]$, using the *$M$-coordinate-smoothness* of f:

$$\begin{aligned}
\mathbb{E}_j[F(\theta - \gamma_j g_j e_j) - F(w)|\theta] &= \sum_{j=1}^p \frac{1}{p}(F(\theta - \gamma_j g_j) - F(w)) \\
&\overset{F=f+\psi}{=} \frac{1}{p}\sum_{j=1}^p f(\theta - \gamma_j g_j) - f(w) + \psi(\theta - \gamma_j g_j) - \psi(w) \\
&\overset{f \text{ smooth}}{\leq} \frac{1}{p}\sum_{j=1}^p \left(f(\theta) + \langle \nabla f(\theta), -\gamma_j g_j\rangle + \frac{1}{2}\|\gamma_j g_j\|_M^2 - f(w) + \psi(\theta - \gamma_j g_j) - \psi(w)\right) \\
&= f(\theta) - f(w) + \frac{1}{p}\sum_{j=1}^p \left(\langle \nabla f(\theta), -\gamma_j g_j\rangle + \frac{1}{2}\|\gamma_j g_j\|_M^2 + (\psi(\theta - \gamma_j g_j) - \psi(w))\right) \\
&= f(\theta) - f(w) + \frac{1}{p}\langle \nabla f(\theta), -\Gamma g\rangle + \frac{1}{2p}\|\Gamma g\|_M^2 + \frac{1}{p}\sum_{j=1}^p (\psi(\theta - \gamma_j g_j) - \psi(w)) \quad .
\end{aligned} \tag{A.3.3}$$

The regularization terms can now be reorganized using the separability of ψ, as done by Richtárik and Takáč, 2014. Indeed, we notice that

$$\begin{aligned}
\sum_{j=1}^p (\psi(\theta - \gamma_j g_j) - \psi(w)) &= \sum_{j=1}^p \left(\psi_j(\theta_j - \gamma_j g_j) - \psi_j(w_j) + \sum_{j' \neq j} \psi_{j'}(\theta_{j'}) - \psi(w_{j'})\right) \\
&= \psi(\theta - \Gamma g) - \psi(w) + (p-1)(\psi(\theta) - \psi(w)) \quad .
\end{aligned} \tag{A.3.4}$$

Plugging (A.3.4) in (A.3.3) results in the following:

$$\begin{aligned}
&\mathbb{E}_j[F(\theta - \gamma_j g_j e_j) - F(w)|\theta] \\
&\leq f(\theta) - f(w) + \frac{1}{p}\langle \nabla f(\theta), -\Gamma g\rangle + \frac{1}{2p}\|\Gamma g\|_M^2 \\
&\quad + \frac{1}{p}(\psi(\theta - \Gamma g) - \psi(w)) + \frac{p-1}{p}(\psi(\theta) - \psi(w)) \\
&= \frac{1}{p}\left(f(\theta) - f(w) + \langle \nabla f(\theta), -\Gamma g\rangle + \frac{1}{2}\|\Gamma g\|_M^2 + \psi(\theta - \Gamma g) - \psi(w)\right) \\
&\quad + \frac{p-1}{p}(f(\theta) + \psi(\theta) - f(w) - \psi(w)) \ , \qquad \text{(A.3.5)}
\end{aligned}$$

which gives the lemma since $F = f + \psi$. □

To exploit this result, we need to upper bound the right hand side of (A.3.2) for the realizations of θ^k in Algorithm 4.3.1. This is where our proof differs from classical convergence proofs for coordinate descent methods. Namely, we rewrite the right hand side of (A.3.2) so as to obtain telescopic terms plus a bias term resulting from the addition of noise, as shown in Lemma A.3.2.

Lemma A.3.2. *Let ℓ, f, ψ, and F defined as in Section A.3.1. For $k > 0$, let θ^k and θ^{k+1} be two consecutive iterates of the inner loop of Algorithm 4.3.1, $\gamma_1 = \frac{1}{M_1}, \dots, \gamma_p = \frac{1}{M_p} > 0$ the coordinate-wise step sizes (where M_j are the coordinate-wise smoothness constants of f), and $g_j = \frac{1}{\gamma_j}(\theta_j^{k+1} - \theta_j^k)$. Let $w \in \mathbb{R}^p$ an arbitrary vector and $\sigma_1, \dots, \sigma_p > 0$ the coordinate-wise noise scales given as input to Algorithm 4.3.1. It holds that*

$$\begin{aligned}
&\mathbb{E}_{j,\eta}\big[F(\theta^{k+1}) - F(w)\big|\theta^k\big] - \frac{p-1}{p}(F(\theta^k) - F(w)) \\
&\qquad \leq \tfrac{1}{2}\|\theta^k - w\|_{\Gamma^{-1}}^2 - \tfrac{1}{2}\mathbb{E}_{j,\eta}\big[\|\theta^{k+1} - w\|_{\Gamma^{-1}}^2\big|\theta^k\big] + \tfrac{1}{p}\|\sigma\|_\Gamma^2 \ , \qquad \text{(A.3.6)}
\end{aligned}$$

where $\|\sigma\|_\Gamma^2 = \sum_{j=1}^p \gamma_j \sigma_j^2$ and the expectations are taken over the random choice of j and η, conditioned upon the realization of θ^k.

Proof. We define g the vector $(g_1, \dots, g_p) \in \mathbb{R}^p$ with $g_j = \frac{1}{\gamma_j}(\theta_j^{k+1} - \theta_j^k)$ when coordinate j is chosen in Algorithm 4.3.1. We also denote by $\Gamma = \mathrm{diag}(\gamma_1, \dots, \gamma_p)$ the diagonal matrix having the step sizes as its coefficients.

From Lemma A.3.1 with $\theta = \theta^k$, $w = w$ and $g = g$ as defined above we obtain

$$\begin{aligned}
&\mathbb{E}_j\big[F(\theta^k - \gamma_j g_j e_j) - F(w)\big|\theta^k\big] - \frac{p-1}{p}(F(\theta^k) - F(w)) \\
&\leq \frac{1}{p}\left(f(\theta^k) - f(w) + \langle \nabla f(\theta^k), -\Gamma g\rangle + \frac{1}{2}\|\Gamma g\|_M^2 + \psi(\theta^k - \Gamma g) - \psi(w)\right) \ . \qquad \text{(A.3.7)}
\end{aligned}$$

We can upper bound the right hand term of (A.3.7) using the convexity of f and ψ:

$$\begin{aligned} &f(\theta^k) - f(w) + \langle \nabla f(\theta^k), -\Gamma g \rangle + \frac{1}{2}\|\Gamma g\|_M^2 + \psi(\theta^k - \Gamma g) - \psi(w) \\ &\quad \leq \langle \nabla f(\theta^k), \theta^k - w \rangle + \langle \nabla f(\theta^k), -\Gamma g \rangle + \frac{1}{2}\|\Gamma g\|_M^2 + \langle \partial\psi(\theta^k - \Gamma g), \theta^k - \Gamma g - w \rangle \\ &\quad = \langle \nabla f(\theta^k) + \partial\psi(\theta^k - \Gamma g), \theta^k - \Gamma g - w \rangle + \frac{1}{2}\|\Gamma g\|_M^2 \ , \end{aligned} \tag{A.3.8}$$

where we use the slight abuse of notation $\partial\psi(\theta^k - \Gamma g)$ to denote any vector in the subdifferential of ψ at the point $\theta^k - \Gamma g$. We now rewrite the dot product:

$$\begin{aligned} &\langle \nabla f(\theta^k) + \partial\psi(\theta^k - \Gamma g), \theta^k - \Gamma g - w \rangle + \frac{1}{2}\|\Gamma g\|_M^2 \\ &\quad = \langle g, \theta^k - \Gamma g - w \rangle + \frac{1}{2}\|\Gamma g\|_M^2 + \langle \nabla f(\theta^k) + \partial\psi(\theta^k - \Gamma g) - g, \theta^k - \Gamma g - w \rangle \\ &\quad = \underbrace{\langle g, \theta^k - w \rangle - \|g\|_\Gamma^2 + \frac{1}{2}\|g\|_{\Gamma^2 M}^2}_{\text{“descent” term}} + \underbrace{\langle \nabla f(\theta^k) + \partial\psi(\theta^k - \Gamma g) - g, \theta^k - \Gamma g - w \rangle}_{\text{“noise” term}} \ , \end{aligned} \tag{A.3.9}$$

where the second equality follows from $\langle g, -\Gamma g \rangle = -\|g\|_\Gamma^2$ and $\|\Gamma g\|_M^2 = \|g\|_{\Gamma^2 M}^2$. We split (A.3.9) into two terms: a “descent” term and a “noise” term.

Rewriting the “descent” term. We first focus on the “descent” term. As $\gamma_j = \frac{1}{M_j}$ for all $j \in [p]$, it holds that $\gamma_j^2 M_j = \gamma_j$ which gives $-\|g\|_\Gamma^2 + \frac{1}{2}\|g\|_{\Gamma^2 M}^2 = -\|g\|_\Gamma^2 + \frac{1}{2}\|g\|_\Gamma^2 = -\frac{1}{2}\|g\|_\Gamma^2$. We can now rewrite the “descent” term as a difference of two norms, materializing the distance to w, weighted by the inverse of the step sizes Γ^{-1}:

$$\begin{aligned} \text{“descent” term} &= \langle g, \theta^k - w \rangle - \frac{1}{2}\|g\|_\Gamma^2 \\ &= \langle \Gamma g, \theta^k - w \rangle_{\Gamma^{-1}} - \frac{1}{2}\|\Gamma g\|_{\Gamma^{-1}}^2 \\ &= \frac{1}{2}\|\theta^k - w\|_{\Gamma^{-1}}^2 - \frac{1}{2}\|\theta^k - w\|_{\Gamma^{-1}}^2 + \langle \Gamma g, \theta^k - w \rangle_{\Gamma^{-1}} - \frac{1}{2}\|\Gamma g\|_{\Gamma^{-1}}^2 \\ &= \frac{1}{2}\|\theta^k - w\|_{\Gamma^{-1}}^2 - \frac{1}{2}\|\theta^k - \Gamma g - w\|_{\Gamma^{-1}}^2 \ , \end{aligned} \tag{A.3.10}$$

where we factorized the norm to obtain the last inequality. We can rewrite (A.3.10) as an expectation over the random choice of the coordinate j (drawn uniformly in

$[p]$), given the realizations of θ^k and of the noise η (which determines g):

$$\begin{aligned}
&\frac{1}{2}\|\theta^k - w\|^2_{\Gamma^{-1}} - \frac{1}{2}\|\theta^k - \Gamma g - w\|^2_{\Gamma^{-1}} \\
&\quad = \frac{p}{2} \times \left(\frac{1}{p} \sum_{j=1}^{p} \gamma_j^{-1} |\theta_j^k - w_j|^2 - \gamma_j^{-1} |\theta_j^k - \gamma_j g_j - w_j|^2 \right) && \text{(A.3.11)} \\
&\quad = \frac{p}{2} \times \mathbb{E}_j\big[\gamma_j^{-1} |\theta_j^k - w_j|^2 - \gamma_j^{-1} |\theta_j^k - \gamma_j g_j - w_j|^2 \big| \theta^k, \eta \big] \quad . && \text{(A.3.12)}
\end{aligned}$$

Finally, we remark that $\gamma_j^{-1}|\theta_j^k - w_j|^2 - \gamma_j^{-1}|\theta_j^k - \gamma_j g_j - w_j|^2 = \|\theta^k - w\|^2_{\Gamma^{-1}} - \|\theta^k - \gamma_j g_j - w\|^2_{\Gamma^{-1}}$, as only one coordinate changes between the two vectors, and the squared norm $\|\cdot\|^2_{\Gamma^{-1}}$ is separable. We thus obtain

$$\begin{aligned}
\text{“descent” term} &= \mathbb{E}_j\Big[\frac{p}{2}\|\theta^k - w\|^2_{\Gamma^{-1}} - \frac{p}{2}\|\theta^k - \gamma_j g_j - w\|^2_{\Gamma^{-1}} \Big| \theta^k, \eta \Big] && \text{(A.3.13)} \\
&= \frac{p}{2}\|\theta^k - w\|^2_{\Gamma^{-1}} - \frac{p}{2}\mathbb{E}_j\big[\|\theta^{k+1} - w\|^2_{\Gamma^{-1}} \big| \theta^k, \eta \big] \quad . && \text{(A.3.14)}
\end{aligned}$$

Upper bounding the “noise” term. We now upper bound the “noise” term in (A.3.9). We first recall the definition of the noisy proximal update g_j and define its non-noisy counterpart $\tilde{g}_j$:

$$\begin{aligned}
g_j &= \gamma_j^{-1} \Big(\operatorname{prox}_{\gamma_j \psi_j}(\theta_j^k - \gamma_j(\nabla_j f(\theta^k) + \eta_j)) - \theta_j^k \Big) && \text{(A.3.15)} \\
\tilde{g}_j &= \gamma_j^{-1} \Big(\operatorname{prox}_{\gamma_j \psi_j}(\theta_j^k - \gamma_j(\nabla_j f(\theta^k)) - \theta_j^k \Big) \quad . && \text{(A.3.16)}
\end{aligned}$$

For an update of the coordinate $j \in [p]$, the optimality condition of the proximal operator gives, for η_j the realization of the noise drawn at the current iteration when coordinate j is chosen:

$$\begin{aligned}
&0 \in \theta_j^{k+1} - \theta_j^k + \gamma_j(\nabla_j f(\theta^k) + \eta_j)) + \frac{1}{M_j} \partial \psi_j(\theta_j^k - \gamma_j g_j) && \text{(A.3.17)} \\
&\quad = \gamma_j \times \left(\frac{1}{\gamma_j}(\theta_j^{k+1} - \theta_j^k) + \nabla_j f(\theta^k) + \eta_j + \partial \psi_j(\theta_j^k - \gamma_j g_j) \right) \quad . && \text{(A.3.18)}
\end{aligned}$$

As such, there exists a real number $v_j \in \partial\psi_j(\theta_j^k - \gamma_j g_j)$ such that $g_j = -\frac{1}{\gamma_j}(\theta_j^{k+1} - \theta_j^k) = \nabla_j f(\theta^k) + \eta_j + v_j$. We denote by $v \in \mathbb{R}^p$ the vector having this v_j as j-th coordinate. Recall that ψ is separable, therefore $v \in \partial\psi(\theta^k - \Gamma g)$. The “noise” term of (A.3.9) can be thus be rewritten using v:

$$\begin{aligned}
\text{“noise” term} &= \langle \nabla f(\theta^k) + v - g, \theta^k - \Gamma g - w \rangle \\
&= \langle \eta, \theta^k - \Gamma g - w \rangle \quad , && \text{(A.3.19)}
\end{aligned}$$

and we now separate this term in two using $\widetilde{g}$:

$$\begin{aligned}\text{“noise” term} &= \sum_{j=1}^{p} \eta_j(\theta_j^k - \gamma_j g_j - w_j) \\ &= \sum_{j=1}^{p} \eta_j(\theta_j^k - \gamma_j \widetilde{g}_j - w_j) + \sum_{j=1}^{p} \eta_j(\gamma_j \widetilde{g}_j - \gamma_j g_j) \ . \end{aligned} \tag{A.3.20}$$

It is now time to consider the expectation with respect to the noise of these terms. First, as $\widetilde{g}_j$ is not dependent on the noise anymore, it simply holds that

$$\mathbb{E}_\eta\Big[\sum_{j=1}^{p} \eta_j(\theta_j^k - \gamma_j \widetilde{g}_j - w_j) \mid \theta^k\Big] = \sum_{j=1}^{p} \mathbb{E}_\eta[\eta_j]\,(\theta_j^k - \gamma_j \widetilde{g}_j - w_j) = 0 \ . \tag{A.3.21}$$

The last step of our proof now takes care of the following term:

$$\begin{aligned}\mathbb{E}_\eta\Big[\sum_{j=1}^{p} \eta_j(\gamma_j \widetilde{g}_j - \gamma_j g_j) \mid \theta^k\Big] &\leq \mathbb{E}_\eta\Big[\gamma_j \Big| \sum_{j=1}^{p} \eta_j(\widetilde{g}_j - g_j)\Big| \mid \theta^k\Big] \\ &\leq \sum_{j=1}^{p} \gamma_j \mathbb{E}_\eta\big[\ |\eta_j||\widetilde{g}_j - g_j|\ \big|\ \theta^k\big] \ , \end{aligned} \tag{A.3.22}$$

where each inequality comes from the triangle inequality. The non-expansiveness property of the proximal operator (see Parikh and Boyd (2014), Section 2.3) is now key to our result, as it yields

$$\begin{aligned}|\widetilde{g}_j - g_j| &= \gamma_j^{-1} |\operatorname{prox}_{\gamma_j \psi_j}(\theta_j^k - \gamma_j(\nabla_j f(\theta^k))) - \operatorname{prox}_{\gamma_j \psi_j}(\theta_j^k - \gamma_j(\nabla_j f(\theta^k) + \eta_j))| \\ &\leq |\eta_j| \ , \end{aligned} \tag{A.3.23}$$

which directly gives, as $\mathbb{E}_\eta\big[\eta_j^2\big] = \sigma_j^2$ (and $\|\sigma\|_\Gamma^2 = \sum_{j=1}^p \gamma_j \sigma_j^2$),

$$\sum_{j=1}^{p} \gamma_j \mathbb{E}_\eta\big[|\eta_j||\widetilde{g}_j - g_j|\big|\theta^k\big] \leq \sum_{j=1}^{p} \gamma_j \mathbb{E}_\eta[|\eta_j||\eta_j|] = \sum_{j=1}^{p} \gamma_j \mathbb{E}_\eta\big[\eta_j^2\big] = \|\sigma\|_\Gamma^2 \ . \tag{A.3.24}$$

We now have everything to prove the lemma by plugging (A.3.24) and (A.3.21) into expected value of (A.3.20), and then (A.3.20) and (A.3.10) back into (A.3.9) to obtain, after using the Tower property of conditional expectations:

$$\frac{1}{p}\mathbb{E}_{j,\eta}\Big[f(\theta^k) - f(w) + \langle \nabla f(\theta^k), -\Gamma g\rangle + \frac{1}{2}\|\Gamma g\|_M^2 + \psi(\theta^k - \Gamma g) - \psi(w)\Big|\theta^k\Big] \tag{A.3.25}$$

$$\leq \frac{1}{p}\,(\text{“descent” term} + \text{“noise” term}) \tag{A.3.26}$$

$$\leq \frac{1}{2}\|\theta^k - w\|_{\Gamma^{-1}}^2 - \frac{1}{2}\mathbb{E}_{j,\eta}\big[\|\theta^{k+1} - w\|_{\Gamma^{-1}}^2\big|\theta^k\big] + \frac{1}{p}\|\sigma\|_\Gamma^2 \ , \tag{A.3.27}$$

which is the result of the lemma. □

A.3.2 (b) Convergence Lemma

Lemma A.3.2 allows us to prove a result on the mean of K consecutive noisy coordinate-wise gradient updates, by simply summing it and rewriting the terms. This gives Lemma A.3.3, which is the key lemma of our proof.

Lemma A.3.3. *Assume $\ell(\cdot, d)$ is convex, L-coordinate-Lipschitz and M-coordinate-smooth for all $d \in \mathcal{X}$, ψ is convex and separable, such that $F = f + \psi$ and w^* is a minimizer of F. For $t \in [T]$, consider the K successive iterates $\theta^1, \dots, \theta^K$ computed from the inner loop of Algorithm 4.3.1 starting from the point $\bar{w}^t$, with step sizes $\gamma_j = \frac{1}{M_j}$ and noise scales σ_j. Letting $\bar{w}^{t+1} = \frac{1}{K}\sum_{k=1}^{K} \theta^k$, it holds that*

$$\mathbb{E}\big[F(\bar{w}^{t+1}) - F(w^*)\big] \leq \frac{p(\|\bar{w}^t - w^*\|_M^2 + 2(F(\bar{w}^t) - F(w^*)))}{2K} + \|\sigma\|_{M^{-1}}^2 \ . \quad \text{(A.3.28)}$$

Remark A.3.2. *The term $F(\bar{w}^t) - F(w^*)$ essentially remains in the inequality due to the composite nature of F. When $\psi = 0$, M-coordinate-smoothness of $f(\cdot; d)$ (for $d \in \mathcal{X}$) gives*

$$f(\bar{w}^t) \leq f(w^*) + \langle \nabla f(w^*), \bar{w}^t - w^* \rangle + \frac{1}{2}\|\bar{w}^t - w^*\|_M^2 = f(w^*) + \frac{1}{2}\|\bar{w}^t - w^*\|_M^2 \ , \quad \text{(A.3.29)}$$

and the result of Lemma A.3.3 further simplifies as:

$$\mathbb{E}\big[F(\bar{w}^{t+1}) - F(w^*)\big] \leq \frac{p\|\bar{w}^t - w^*\|_M^2}{K} + \|\sigma\|_{M^{-1}}^2 \ . \quad \text{(A.3.30)}$$

Proof. Summing Lemma A.3.2 for $k = 0$ to $k = K$ and $w = w^*$, taking expectation with respect to all choices of coordinate and random noise and using the tower property gives:

$$\begin{aligned}
&\sum_{k=0}^{K-1} \mathbb{E}\big[F(\theta^{k+1}) - F(w^*)\big] - \frac{p-1}{p}\sum_{k=0}^{K-1} \mathbb{E}\big[(F(\theta^k) - F(w^*))\big] \\
&\quad \leq \sum_{k=0}^{K-1} \frac{1}{2}\mathbb{E}\big[\|\theta^k - w^*\|_{\Gamma^{-1}}^2\big] - \frac{1}{2}\mathbb{E}\big[\|\theta^{k+1} - w^*\|_{\Gamma^{-1}}^2\big] + \frac{1}{p}\|\sigma\|_{\Gamma}^2 \quad \text{(A.3.31)} \\
&\quad = \frac{1}{2}\mathbb{E}\big[\|\bar{w}^0 - w^*\|_{\Gamma^{-1}}^2\big] - \frac{1}{2}\mathbb{E}\big[\|\theta^K - w^*\|_{\Gamma^{-1}}^2\big] + \frac{K}{p}\|\sigma\|_{\Gamma}^2 \ . \quad \text{(A.3.32)}
\end{aligned}$$

Remark that $\sum_{k=0}^{K-1} \mathbb{E}\big[F(\theta^k) - F(w^*)\big] = \sum_{k=1}^{K} \mathbb{E}\big[F(\theta^k) - F(w^*)\big] + (F(\bar{w}^0) - F(w^*)) - \mathbb{E}\big[F(\theta^K) - F(w^*)\big]$, then as $\mathbb{E}\big[F(\theta^K) - F(w^*)\big] \geq 0$, we obtain a lower bound on the

left hand side of (A.3.32):

$$\sum_{k=0}^{K-1} \mathbb{E}\big[F(\theta^{k+1}) - F(w^*)\big] - \tfrac{p-1}{p} \sum_{k=0}^{K-1} \mathbb{E}\big[(F(\theta^k) - F(w^*))\big]$$
$$\geq \tfrac{1}{p} \sum_{k=1}^{K} \mathbb{E}\big[F(\theta^k) - F(w^*)\big] - (F(\bar{w}^0) - F(w^*)) \ . \tag{A.3.33}$$

As $\bar{w}^{t+1} = \frac{1}{K}\sum_{k=1}^{K} \theta^k$, the convexity of F gives $F(\bar{w}^{t+1}) \leq \frac{1}{K}\sum_{k=1}^{K} F(\theta^k) - F(w^*)$. Plugging this inequality into (A.3.33) and combining the result with (A.3.32) gives

$$F(\bar{w}^{t+1}) - F(w^*) \leq \frac{p(\frac{1}{2}\|\bar{w}^0 - w^*\|^2_{\Gamma^{-1}} + F(\bar{w}^0) - F(w^*))}{K} + \|\sigma\|^2_\Gamma \ . \tag{A.3.34}$$

We conclude the proof by using the fact that $\Gamma_j = M_j^{-1}$ for all $j \in [p]$, thus $\|\cdot\|_\Gamma = \|\cdot\|_{M^{-1}}$ and $\|\cdot\|_{\Gamma^{-1}} = \|\cdot\|_M$. □

A.3.2 (c) Convex Case

Theorem A.3.1. *4.3.2[Convex case] Let w^* be a minimizer of F and $R_M^2 = \max(\|\bar{w}^0 - w^*\|_M^2, F(\bar{w}^0) - F(w^*))$. The output w^{priv} of DP-CD (Algorithm 4.3.1), starting from $\bar{w}^0 \in \mathbb{R}^p$ with $T = 1$, $K > 0$ and the σ_j's as in Theorem 4.3.1, satisfies:*

$$F(w^{priv}) - F(w^*) \leq \frac{3pR_M^2}{2K} + \frac{12\|L\|^2_{M^{-1}} K \log(1/\delta)}{n^2\epsilon^2} \ . \tag{A.3.35}$$

Setting $K = \frac{R_M\sqrt{p}n\epsilon}{\|L\|_{M^{-1}}\sqrt{8\log(1/\delta)}}$ yields:

$$F(w^{priv}) - F(w^*) \leq \frac{9\sqrt{p}\|L\|_{M^{-1}} R_M \sqrt{\log(1/\delta)}}{n\epsilon} = \widetilde{O}\left(\frac{\sqrt{p}R_M\|L\|_{M^{-1}}}{n\epsilon}\right) \ . \tag{A.3.36}$$

Proof. In the convex case, we iterate only once in the inner loop (since $T = 1$). As such, $w^{priv} = \bar{w}^1$, and applying Lemma A.3.3 with $\bar{w}^{t+1} = \bar{w}^1$, $w^t = \bar{w}^0$ and σ_j chosen as in Theorem 4.3.1 gives the result. Taking $K = \frac{R_M\sqrt{p}n\epsilon}{\|L\|_{M^{-1}}\sqrt{8\log(1/\delta)}}$ then gives

$$F(\bar{w}_1^{t+1}) - F(w^*) \leq \frac{2\sqrt{8p\log(1/\delta)}\|L\|_{M^{-1}}R_M}{n\epsilon} + \frac{12\sqrt{p\log(1/\delta)}\|L\|_{M^{-1}}R_M}{\sqrt{8}n\epsilon} \ , \tag{A.3.37}$$

and the result follows from $2\sqrt{8} + \frac{12}{\sqrt{8}} \approx 8.48 < 9$. □

A.3.2 (d) Strongly Convex Case

Theorem A.3.2. *4.3.2[Strongly-convex case] Let F be μ_M-strongly convex w.r.t. $\|\cdot\|_M$ and w^* be the minimizer of F. The output w^{priv} of DP-CD (Algorithm 4.3.1), starting from $\bar{w}^0 \in \mathbb{R}^p$ with $T > 0$, $K = 2p(1 + 1/\mu_M)$ and the σ_j's as in Theorem 4.3.1, satisfies:*

$$F(w^{priv}) - F(w^*) \leq \frac{F(\bar{w}^0) - F(w^*)}{2^T} + \frac{24p(1+1/\mu_M)T\|L\|_{M^{-1}}^2 \log(1/\delta)}{n^2\epsilon^2} \ . \tag{A.3.38}$$

Setting $T = \log_2\left(\frac{32n^2\epsilon^2(F(\bar{w}^0)-F(w^*))}{p(1+1/\mu_M)\|L\|_{M^{-1}}^2\log(1/\delta)}\right)$ *yields:*

$$\begin{aligned}
&\mathbb{E}\big[F(w^{priv}) - F(w^*)\big] \\
&\quad \leq \left(1 + \log_2\left(\frac{(F(\bar{w}^0)-F(w^*))n^2\epsilon^2}{24p(1+1/\mu_M)\|L\|_{M^{-1}}^2\log(1/\delta)}\right)\right)\frac{24p(1+1/\mu_M)\|L\|_{M^{-1}}^2\log(1/\delta)}{n^2\epsilon^2} \\
&\quad = O\left(\frac{p\|L\|_{M^{-1}}^2\log(1/\delta)}{\mu_M n^2\epsilon^2}\log_2\left(\frac{(F(\bar{w}^0)-F(w^*))n\epsilon\mu_M}{p\|L\|_{M^{-1}}\log(1/\delta)}\right)\right) \ .
\end{aligned} \tag{A.3.39}$$

Proof. As F is μ_M-strongly-convex with respect to norm $\|\cdot\|_M$, we obtain for any $w \in \mathbb{R}^p$, that $F(w) \geq F(w^*) + \frac{\mu_M}{2}\|w - w^*\|_M^2$. Therefore, $F(\bar{w}^0) - F(w^*) \geq \frac{2}{\mu_M}\|\bar{w}^0 - w^*\|_M^2$ and Lemma A.3.3 gives, for $1 \leq t \leq T-1$,

$$F(\bar{w}^{t+1}) - F(w^*) \leq \frac{(1+1/\mu_M)p(F(\bar{w}^t) - F(w^*))}{K} + \|\sigma\|_M^2 \ . \tag{A.3.40}$$

It remains to set $K = 2p(1 + 1/\mu_M)$ to obtain

$$F(\bar{w}^{t+1}) - F(w^*) \leq \frac{F(\bar{w}^t) - F(w^*)}{2} + \|\sigma\|_M^2 \ . \tag{A.3.41}$$

Recursive application of this inequality gives

$$\begin{aligned}
\mathbb{E}\big[F(\bar{w}^T) - F(w^*)\big] &\leq \frac{F(\bar{w}^0) - F(w^*)}{2^T} + \sum_{t=0}^{T-1}\frac{1}{2^t}\|\sigma\|_M^2 \\
&\leq \frac{F(\bar{w}^0) - F(w^*)}{2^T} + 2\|\sigma\|_M^2 \ ,
\end{aligned} \tag{A.3.42}$$

where we upper bound the sum by the value of the complete series. It remains to replace $\|\sigma\|_M^2$ by its value to obtain the result. Taking $T = \log_2\left(\frac{(F(\bar{w}^0)-F(w^*))n^2\epsilon^2}{24p(1+1/\mu_M)\|L\|_{M^{-1}}^2\log(1/\delta)}\right)$ then gives

$$\begin{aligned}
\mathbb{E}\big[F(\bar{w}^T) - F(w^*)\big] &\leq \left(1 + \log_2\left(\frac{(F(\bar{w}^0)-F(w^*))n^2\epsilon^2}{24p(1+1/\mu_M)\|L\|_{M^{-1}}^2\log(1/\delta)}\right)\right)\frac{24p(1+1/\mu_M)\|L\|_{M^{-1}}^2\log(1/\delta)}{n^2\epsilon^2} \\
&= O\left(\frac{p\|L\|_{M^{-1}}^2\log(1/\delta)}{\mu_M n^2\epsilon^2}\log_2\left(\frac{(F(\bar{w}^0)-F(w^*))n\epsilon\mu_M}{p\|L\|_{M^{-1}}\log(1/\delta)}\right)\right) \ ,
\end{aligned} \tag{A.3.43}$$

which is the result of our theorem. □

A.3.3 Proof of Remark 1

We recall the notations of Tappenden et al. (2016). For $\theta \in \mathbb{R}^p$, $t \in \mathbb{R}$ and $j \in [p]$, let $V_j(\theta, t) = \nabla_j(\theta)t + \frac{M_j}{2}|t|^2 + \psi_j(\theta_j^k + t)$. For $\eta \in \mathbb{R}$, we also define its noisy counterpart, $V_j^\eta(\theta, t) = (\nabla_j(\theta) + \eta)t + \frac{M_j}{2}|t|^2 + \psi_j(\theta_j^k + t)$. We aim at finding δ_j such that for any $\theta^k \in \mathbb{R}^p$ used in the inner loop of Algorithm 4.3.1:

$$\mathbb{E}_{\eta_j}\big[V_j(\theta^k, -\gamma_j g_j)\big] \leq \min_{\widetilde{g}\in\mathbb{R}} V_j(\theta^k, -\gamma_j \widetilde{g}) + \delta_j \ , \tag{A.3.44}$$

where the expectation is taken over the random noise η_j, and $-\gamma_j g_j = \mathrm{prox}_{\gamma_j\psi_j}(\theta_j^k - \gamma_j(\nabla_j f(\theta^k) + \eta_j)) - \theta_j^k$ as defined in the analysis of Algorithm 4.3.1. We need to link the proximal operator we use in DP-CD with the quantity $V_j^{\eta_j}$ that we just defined:

$$\begin{aligned}
&\mathrm{prox}_{\gamma_j\psi_j}(\theta_j^k - \gamma_j(\nabla_j f(\theta^k) + \eta_j)) \\
&= \arg\min_{v\in\mathbb{R}} \frac{1}{2}\|v - \theta_j^k + \gamma_j(\nabla_j f(\theta^k) + \eta_j)\|_2^2 \\
&= \arg\min_{v\in\mathbb{R}} \langle \gamma_j(\nabla_j f(\theta_j^k) + \eta_j), v - \theta_j^k\rangle + \frac{1}{2}\|v - \theta_j^k\|_2^2 + \gamma_j\psi_j(v) \\
&= \arg\min_{v\in\mathbb{R}} \langle \nabla_j f(\theta^k) + \eta_j, v - \theta_j^k\rangle + \frac{M_j}{2}\|v - \theta_j^k\|_2^2 + \psi_j(v) \\
&= \theta_j^k + \arg\min_{t\in\mathbb{R}} \langle \nabla_j f(\theta^k) + \eta_j, t\rangle + \frac{M_j}{2}\|t\|_2^2 + \psi_j(\theta_j^k + t) \ .
\end{aligned} \tag{A.3.45}$$

Which means that $-\gamma_j g_j = \mathrm{prox}_{\gamma_j\psi_j}(\theta_j^k - \gamma_j(\nabla_j f(\theta^k) + \eta_j)) - \theta_j^k \in \arg\min_{t\in\mathbb{R}} V_j^{\eta_j}(\theta^k, t)$. Let $-\gamma_j g_j^* = \mathrm{prox}_{\gamma_j\psi_j}(\theta_j^k - \gamma_j\nabla_j(\theta^k)) - \theta_j^k$ be the non-noisy counterpart of $-\gamma_j g_j$. Since $-\gamma_j g_j$ is a minimizer of $V_j^{\eta_j}(\theta^k, \cdot)$, it holds that

$$V_j^{\eta_j}(\theta^k, -\gamma_j g_j) \leq \langle \nabla_j f(\theta^k) + \eta_j, -\gamma_j g_j^*\rangle + \frac{M_j}{2}\| - \gamma_j g_j^*\|_2^2 + \psi_j(\theta_j^k + -\gamma_j g_j^*) \tag{A.3.46}$$

$$= \min_t V_j(\theta^k, t) + \langle \eta_j, -\gamma_j g_j^*\rangle \ , \tag{A.3.47}$$

which can be rewritten as $V_j(\theta^k, -\gamma_j g_j) \leq \min_t V_j(\theta^k, t) + \langle \eta_j, \gamma_j(g_j - g_j^*)\rangle$. Taking the expectation yields

$$\mathbb{E}_{\eta_j}\big[V_j(\theta^k, -\gamma_j g_j)\big] \leq \min_t V_j(\theta^k, t) + \mathbb{E}_{\eta_j}\big[\langle \eta_j, \gamma_j(g_j - g_j^*)\rangle\big] \ . \tag{A.3.48}$$

Finally, we remark that $|g_j - g_j^*| \leq |\gamma_j\eta_j|$ and the non-expansiveness of the proximal operator gives

$$\mathbb{E}_{\eta_j}\big[V_j(\theta^k, -\gamma_j g_j)\big] \leq \min_t V_j(\theta^k, t) + \gamma_j\sigma_j^2 \ , \tag{A.3.49}$$

which implies an upper bound on the expectation of δ_j: $\mathbb{E}_{j,\eta_j}[\delta_j] = \frac{1}{p}\sum_{j=1}^p \mathbb{E}_{\eta_j}[\delta_j] \leq \frac{1}{p}\sum_{j=1}^p \gamma_j\sigma_j^2 = \frac{1}{p}\sum_{j=1}^p \sigma_j^2/M_j$, when $\gamma_j = 1/M_j$. In the formalism of Tappenden et al. (2016), this amounts to setting $\alpha = 0$ and $\beta = \frac{1}{p}\|\sigma\|_{M^{-1}}^2$.

Convex functions. When the objective function F is convex, we use Lemma A.3.3 to obtain, since $\|\sigma\|_{M^{-1}}^2 = \beta p$,

$$F(w^1) - F(w^*) \leq \frac{2pR_M^2}{K} + \|\sigma\|_{M^{-1}}^2 = \frac{2pR_M^2}{K} + \beta p \ . \tag{A.3.50}$$

Therefore, when F is convex, we get $F(w^1) - F(w^*) \leq \xi$, for $\xi > \beta p$, as long as $\frac{2pR_M^2}{K} \leq \xi - \beta p$, that is $K \geq \frac{2pR_M^2}{\xi - \beta p}$.

In comparison, Tappenden et al. (2016, Theorem 5.1 therein) gives convergence to $\xi > \sqrt{2pR_M^2\beta}$ when $K \geq \frac{2pR_M^2}{\xi - \sqrt{2pR_M^2\beta}}$. We thus gain a factor $\sqrt{\beta p / 2R_M^2}$ in utility. Importantly, our utility upper bound does not depend on initialization in that setting, whereas the one of Tappenden et al. (2016) does.

Strongly-convex functions. When the objective function F is μ_M-strongly-convex *w.r.t.*, to $\|\cdot\|_M$, then from (A.3.42) we obtain, as long as $K \geq 4/\mu_M$, that

$$\mathbb{E}\big[F(w^T) - F(w^*)\big] \leq \frac{F(w^0) - F(w^*)}{2^T} + 2\beta p \ . \tag{A.3.51}$$

This proves that $\mathbb{E}\big[F(w^T) - F(w^*)\big] \leq \xi$ for $\xi > 2\beta p$ when $\frac{F(w^0)-F(w^*)}{2^T} \leq \xi - 2\beta p$ that is $T \geq \log \frac{F(w^0)-F(w^*)}{\xi - 2\beta p}$ and $TK \geq \frac{4p}{\mu_M} \log \frac{F(w^0)-F(w^*)}{\xi - 2\beta p}$. In comparison, Tappenden et al. (2016, Theorem 5.2 therein) shows convergence to $\xi > \frac{\beta p}{\mu_M}$ for $K \geq \frac{p}{\mu_M} \log \frac{F(w^0)-F(w^*)-\frac{\beta p}{\mu_M}}{\xi - \frac{\beta p}{\mu_M}}$. We thus gain a factor $\mu_M/2$ in utility.

A.4 Comparison with DP-SGD

In this section, we provide more details on the arguments of Section 4.3.4, where we suppose that ℓ is L-coordinate-Lipschitz and Λ-Lipschitz. To ease the comparison, we assume that $R_M = \|w^0 - w^*\|_M$, which is notably the case in the smooth setting with $\psi = 0$ (see Remark A.3.1).

Balanced. We start by the scenario where coordinate-wise smoothness constants are balanced and all equal to $M = M_1 = \cdots = M_p$. We observe that

$$\|L\|_{M^{-1}} = \sqrt{\sum_{j=1}^p \frac{1}{M_j} L_j^2} = \sqrt{\frac{1}{M} \sum_{j=1}^p L_j^2} = \frac{1}{\sqrt{M}} \|L\|_2 \ . \tag{A.4.1}$$

We then consider the convex and strongly-convex functions separately:

- *Convex functions:* it holds that $R_M = \sqrt{M} R_I$, which yields the equality $\|L\|_{M^{-1}} R_M = \|L\|_2 R_I$.
- *Strongly convex functions:* if f is μ_M-strongly-convex with respect to $\|\cdot\|_M$, then for any $x, y \in \mathbb{R}^p$,

$$\begin{aligned} f(y) &\geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu_M}{2} \|y - x\|_M^2 \\ &= f(x) + \langle \nabla f(x), y - x \rangle + \frac{M \mu_M}{2} \|y - x\|_2^2 \ , \end{aligned} \tag{A.4.2}$$

  which means that f is $M\mu_M$-strongly-convex with respect to $\|\cdot\|_2$. This gives $\frac{\|L\|_{M^{-1}}^2}{\mu_M} = \frac{\|L\|_2^2 / M}{\mu_I / M} = \frac{\|L\|_2^2}{\mu_I}$.

In light of the results summarized in Table 4.1, it remains to compare $\|L\|_2 = \sqrt{\sum_{j=1}^p L_j^2}$ with Λ, for which it holds that $\Lambda \leq \sqrt{\sum_{j=1}^p L_j^2} \leq \sqrt{p}\Lambda$, which is our result.

Unbalanced. When smoothness constants are disparate, we discuss the case where

- *one coordinate of the gradient dominates the others:* we assume without loss of generality that the dominating coordinate is the first one. It holds that $M_1 =: M_{\max} \gg M_{\min} =: M_j$, for all $j \neq 1$ and $L_1 =: L_{\max} \gg L_{\min} =: L_j$, for all $j \neq 1$ such that $\frac{L_1^2}{M_1} \gg \sum_{j \neq 1} \frac{L_j^2}{M_j}$. As L_1 dominates the other coordinate-Lipschitz constants, most of the variation of the loss comes from its first coordinate. This implies that L_1 is close to the global Lipschitz constant Λ of ℓ. As such, it holds that

$$\|L\|_{M^{-1}}^2 = \sum_{j=1}^p \frac{L_j^2}{M_j} \approx \frac{L_1^2}{M_1} \approx \frac{\Lambda^2}{M_{\max}} \ . \tag{A.4.3}$$

- *the first coordinate of $\bar{w}^0$ is already very close to its optimal value* so that $M_1 |\bar{w}_1^0 - w_1^*| \ll \sum_{j \neq 1} M_j |\bar{w}_j^0 - w_j^*|$. Under this hypothesis,

$$R_M^2 \approx \sum_{j \neq 1} M_j |w_j^0 - w_j^*|^2 = M_{\min} \sum_{j \neq 1} |w_j^0 - w_j^*|^2 \approx M_{\min} R_I^2 \ . \tag{A.4.4}$$

We can now easily compare DP-CD with DP-SGD in this scenario. First, if ℓ is convex, then $\|L\|_{M^{-1}} R_M \approx \sqrt{\frac{M_{\min}}{M_{\max}}} \Lambda R_I$. Second, when ℓ is strongly-convex, we observe that

for $x, y \in \mathbb{R}^p$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu_M}{2} \|y - x\|_M^2$$
$$\geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{M_{\min}\mu_M}{2} \|y - x\|_2^2 \ , \qquad \text{(A.4.5)}$$

which implies that when f is μ_M strongly-convex with respect to $\|\cdot\|_M$, it is $M_{\min}\mu_M$ strongly-convex with respect to $\|\cdot\|_2$. This yields, under our hypotheses, $\frac{\|L\|_{M^{-1}}^2}{\mu_M} \approx \frac{\Lambda^2/M_{\max}}{\mu_I/M_{\min}} = \frac{M_{\min}}{M_{\max}}\frac{\Lambda^2}{\mu_I}$. In both cases, DP-CD can get arbitrarily better than DP-SGD, and gets better as the ratio $M_{\max}/M_{\min}$ increases.

The two hypotheses we describe above are of course very restrictive. However, it gives some insight about when and why DP-CD can outperform DP-SGD. Our numerical experiments in Section 4.6 confirm this analysis, even in less favorable cases.

A.5 Proof of Lower Bounds

To prove lower bounds on the utility of L-coordinate-Lipschitz functions, we extend the proof of Bassily et al. (2014b) to our setting (that is, L-coordinate-Lipschitz functions and unconstrained composite optimization). There are three main difficulties in adapting their proof:

- First, the optimization problem stated in Equation ($\star'$) is not constrained. We stress that while convex constraints can be enforced using the regularizer ψ (using the characteristic function of a convex set), its separable nature only allows box constraints. In contrast, Bassily et al. (2014b) rely on an ℓ_2-norm constraint to obtain their lower bounds.

- Second, Lemma 5.1 of Bassily et al. (2014b) must be extended to our L-coordinate-Lipschitz setting. To do so, we consider datasets with points in $\prod_{j=1}^p \{-L_j, L_j\}$ rather than $\{-1/\sqrt{p}, 1/\sqrt{p}\}^p$, and carefully adapt the construction of the dataset D so that $\|\sum_{i=1}^n d_i\|_2 = \Omega(\min(n\|L\|_2, \sqrt{p}\|L\|_2/\epsilon))$, which is essential to prove our lower bounds.

- Third, the lower bounds of Bassily et al. (2014b) rely on fingerprinting codes, and in particular on the result of Bun et al. (2014) which uses such codes to prove that (when n is smaller than some n^* we describe later) differential privacy is incompatible with precisely and simultaneously estimating *all* p counting queries defined over the columns of the dataset D. In our construction, since all columns of D now have different scales, we need an additional hypothesis on the repartition of the L_j's, (*i.e.*, that $\sum_{j\in\mathcal{J}} L_j^2 = \Omega(\|L\|_2)$ for all $\mathcal{J} \subseteq [p]$ of a

given size), which is not required in existing lower bounds (where all columns have equal scale).

A.5.1 Counting Queries and Accuracy

We start our proof by recalling and extending to our setting the notions of counting queries (Definition A.5.1) and accuracy (Definition A.5.2), as described by Bun et al. (2014). The main feature of our definitions is that we allow the set $\mathcal{X}$ to have different scales for each of its coordinates, and that we account for this scale in the definition of accuracy. We denote by $\mathrm{conv}(\mathcal{X})$ the convex hull of a set $\mathcal{X}$.

Definition A.5.1 (Counting query)**.** *Let $n > 0$. A counting query on $\mathcal{X}$ is a function $q : \mathcal{X}^n \to \mathrm{conv}(\mathcal{X})$ defined using a predicate $q : \mathcal{X} \to \mathcal{X}$. The evaluation of the query q over a dataset $\mathcal{D} \in \mathcal{X}^n$ is defined as the arithmetic mean of q on $\mathcal{D}$:*

$$q(\mathcal{D}) = \frac{1}{n} \sum_{i=1}^{n} q(d_i) \ . \tag{A.5.1}$$

Definition A.5.2 (Accuracy)**.** *Let $n, p \in \mathbb{N}$, $\alpha, \beta \in [0, 1]$, $L_1, \ldots, L_p > 0$, and $\mathcal{X} = \prod_{j=1}^{p} \{-L_j; L_j\}$ or $\mathcal{X} = \{0, L_j\}^p$. Let $\mathcal{Q} = \{q_1, \ldots, q_p\}$ be a set of p counting queries on $\mathcal{X}$ and $D \in \mathcal{X}^n$ a dataset of n elements. A sequence of answers $a = (a_1, \ldots, a_p)$ is said (α, β)-accurate for $\mathcal{Q}$ if $|q_j(D) - a_j| \leq L_j \alpha$ for at least a $1 - \beta$ fraction of indices $j \in [p]$. A randomized algorithm $\mathcal{A} : \mathcal{X}^n \to \mathbb{R}^{\mathrm{card}\, \mathcal{Q}}$ is said (α, β)-accurate for $\mathcal{Q}$ on $\mathcal{X}$ if for every $D \in \mathcal{X}^n$,*

$$\mathbb{P}\, \mathcal{A}(D) \text{ is } (\alpha, \beta)\text{-accurate for } \mathcal{Q} \geq 2/3 \ . \tag{A.5.2}$$

In our proof, we will use a specific class of queries: one-way marginals (Definition A.5.3), that compute the arithmetic mean of a dataset along one of its column.

Definition A.5.3 (One-way marginals)**.** *Let $\mathcal{X} = \prod_{j=1}^{p} \{-L_j; L_j\}$ or $\mathcal{X} = \{0, L_j\}^p$. The family of one-way marginals on $\mathcal{X}$ is defined by queries with predicates $q_j(x) = x_j$ for $x \in \mathcal{X}$. For a dataset $D \in \mathcal{X}^n$ of size n, we thus have $q_j(D) = \frac{1}{n} \sum_{i=1}^{n} d_{i,j}$.*

A.5.2 Lower Bound for One-Way Marginals

We can now restate a key result from Bun et al. (2014), which shows that there exists a minimal number n^* of records needed in a dataset to allow achieving both accuracy and privacy on the estimation of one-way marginals on $\mathcal{X} = (\{0, 1\}^p)^n$. This lemma relies on the construction of re-identifiable distribution (see Bun et al. 2014, Definition 2.10). One can then use this distribution to find a dataset on which a private algorithm can not be accurate (see Bun et al. 2014, Lemma 2.11).

Lemma A.5.1 (Bun et al. 2014, Corollary 3.6)**.** *For $\epsilon > 0$ and $p > 0$, there exists a number $n^* = \Omega(\frac{\sqrt{p}}{\epsilon})$ such that for all $n \leq n^*$, there exists no algorithm that is both $(1/3, 1/75)$-accurate and $(\epsilon, o\left(\frac{1}{n}\right))$-differentially private for the estimation of one-way marginals on $(\{0,1\}^p)^n$.*

To leverage this result in our setting of private empirical risk minimization, we start by extending it to queries on $\mathcal{X} = \prod_{j=1}^{p}\{-L_j; L_j\}$. Before stating the main theorem of this section (Theorem A.5.1), we describe a procedure $\chi_L : (\{0,1\}^p)^n \to \mathcal{X}^{3n}$ (with $L_1, \ldots, L_p > 0$), that takes as input a dataset $D \in (\{0,1\}^p)^n$ and outputs an augmented and rescaled version. This procedure is crucial to our proof and is defined as follows. First, it adds $2n$ rows filled with 1's to D, which ensures that the sum of each column of D is $\Theta(n)$ (which gives the lower bound on M in Theorem A.5.1). Then it rescales each of these columns by subtracting $1/2$ to each coefficient and multiplying the j-th column of D ($j \in [p]$) by $2L_j$. The resulting dataset $D_L^{aug} = \chi_L(D)$ is a set of $3n$ points with values in $\mathcal{X} = \prod_{j=1}^{p}\{-L_j, L_j\}$, with the property that, for all $j \in [p]$, $3nL_j \geq \sum_{i=1}^{n}(D_L^{aug})_{i,j} \geq nL_j$. For $D \in (\{0,1\}^p)^n$, we show how to reconstruct $q_j(\chi_L(D))$ from $q_j(D)$ in Lemma A.5.2.

Lemma A.5.2. *Let $n \in \mathbb{N}$, $j \in [p]$, $L_j > 0$ and q_j the j-th one-way marginal on datasets D with p columns such that for $d_i \in D$, $q_j(d_i) = d_{i,j}$. Let $D_L^{aug} = \chi_L(D)$. It holds that*

$$q_j(D_L^{aug}) = \frac{2L_j}{3} q_j(D) + \frac{L_j}{3} \ , \tag{A.5.3}$$

where we use the slight abuse of notation by denoting the one-way marginals $q_j : \mathcal{X}^{3n} \to \mathrm{conv}(\mathcal{X})$ and $q_j : (\{0,1\}^p)^n \to [0,1]^p$ in the same way.

Proof. Let $D \in (\{0,1\}^p)^n$, and let $D^{aug} \in (\{0,1\}^p)^{3n}$ constructed by adding $2n$ rows of 1's at the end of D. Let $D_L^{aug} = \chi_L(D)$. We remark that

$$\begin{aligned} q_j(D^{aug}) &= \frac{1}{3n}\sum_{i=1}^{3n} D_{i,j}^{aug} \\ &= \frac{1}{3}\left(\frac{1}{n}\sum_{i=1}^{n} D_{i,j}^{aug}\right) + \frac{1}{3n}\sum_{i=n+1}^{3n} 1 \\ &= \frac{1}{3}q_j(D) + \frac{2}{3} \in [0,1] \ . \end{aligned} \tag{A.5.4}$$

Then, we link $q_j(D^{aug})$ with $q_j(D_L^{aug})$:

$$\begin{aligned} q_j(D_L^{aug}) &= \frac{1}{3n}\sum_{i=1}^{3n}(D_L^{aug})_{i,j} \\ &= \frac{1}{3n}\sum_{i=1}^{3n} 2L_j((D^{aug})_{i,j} - 1/2) \\ &= 2L_j(q_j(D^{aug}) - 1/2) \in [-L_j, L_j] \ , \end{aligned} \tag{A.5.5}$$

combining (A.5.4) and (A.5.5) gives the result. □

Theorem A.5.1. *Let $n, p \in \mathbb{N}$, and $L_1, \dots, L_p > 0$. Assume that for all subsets $\mathcal{J} \subseteq [p]$ of size at least $\lceil \frac{p}{75} \rceil$, $\sqrt{\sum_{j\in\mathcal{J}} L_j^2} = \Omega(\|L\|_2)$. Define $\mathcal{X} = \prod_{j=1}^{p}\{-L_j; +L_j\}$, and let $q_j : \mathcal{X} \to \{-L_j, L_j\}$ be the predicate of the j-th one-way marginal on $\mathcal{X}$. Take $\epsilon > 0$ and $\delta = o(\frac{1}{n})$. There exists a number $M = \Omega\left(\min\left(n\|L\|_2, \frac{\sqrt{p}\|L\|_2}{\epsilon}\right)\right)$ such that for every (ϵ, δ)-differentially private algorithm $\mathcal{A}$, there exists a dataset $D = \{d_1, \dots, d_n\} \in \mathcal{X}^n$ with $\|\sum_{i=1}^n d_i\|_2 \in [M-1, M+1]$ such that, with probability at least $1/3$ over the randomness of $\mathcal{A}$:*

$$\|\mathcal{A}(D) - q(D)\|_2 = \Omega\left(\min\left(\|L\|_2, \frac{\sqrt{p}\|L\|_2}{n\epsilon}\right)\right) \ . \tag{A.5.6}$$

Proof. Let $M = \Omega\left(\min\left(n\|L\|_2, \frac{\sqrt{p}\|L\|_2}{\epsilon}\right)\right)$, and define the set of queries $\mathcal{Q}$ composed of p queries $q_j(D) = \frac{1}{n}\sum_{i=1}^n d_{i,j}$ for $j \in [p]$. Let $\mathcal{A}$ be a (ϵ, δ)-differentially-private randomized algorithm. Let $\alpha, \beta \in [0, 1]$. We will show that there exists a dataset D such that $\|\sum_{i=1}^n d_i\|_2 \in [M-1, M+1]$ for which $\mathcal{A}(D)$ is not (α, β)-accurate.

When $n \leq n^*$. Assume, for the sake of contradiction, that $\mathcal{A} : \mathcal{X}^{3n} \to \text{conv}(\mathcal{X})$ is $(\frac{1}{3}\alpha, \beta)$-accurate for $\mathcal{Q}$. Then, for each dataset $D' \in \mathcal{X}^{3n}$, we have

$$\mathbb{P}\left(\exists \mathcal{J} \subseteq [p] \text{ with } |\mathcal{J}| \geq (1-\beta)p \text{ and } \forall j \in \mathcal{J},\ |\mathcal{A}_j(D') - q_j(D')| < \frac{2L_j}{3}\alpha\right) \geq 2/3 \ . \tag{A.5.7}$$

Importantly, for all $D \in (\{0,1\})^p)^n$, the randomized algorithm $\mathcal{A}$ satisfies (A.5.7) for the dataset $D_L^{aug} = \chi_L(D) \in \mathcal{X}^{3n}$. We now construct the mechanism $\widetilde{\mathcal{A}} : (\{0,1\}^p)^n \to [0,1]^p$ that takes a dataset $D \in (\{0,1\}^p)^n$, constructs $D_L^{aug} = \chi_L(D)$ and runs $\mathcal{A}$ on it. It then outputs $\widetilde{\mathcal{A}}(D)$ such that, for $j \in [p]$, $\widetilde{\mathcal{A}}_j(D) = \frac{3}{2L_j}\mathcal{A}_j(D_L^{aug}) - \frac{L_j}{3}$. Using

Lemma A.5.2, the results of $\widetilde{\mathcal{A}}$ and be linked to the ones of $\mathcal{A}$, as

$$|\widetilde{\mathcal{A}}(D) - q_j(D)| = |\frac{3}{2L_j}\mathcal{A}_j(D_L^{aug}) - \frac{L_j}{3} - \frac{3}{2L_j}q_j(D_L^{aug}) + \frac{L_j}{3}| \\ = \frac{3}{2L_j}|\mathcal{A}_j(D_L^{aug}) - q_j(D_L^{aug})| \ . \tag{A.5.8}$$

Therefore, if $\mathcal{A}$ satisfies (A.5.7) and (A.5.8), then $\widetilde{\mathcal{A}} : (\{0,1\}^p)^n \to [0,1]^p$ satisfies, for all $D \in (\{0,1\}^p)^n$,

$$\mathbb{P}\left(\exists \mathcal{J} \subseteq [p] \text{ with } |\mathcal{J}| \geq (1-\beta)p \text{ and } \forall j \in \mathcal{J},\ |\widetilde{\mathcal{A}}_j(D) - q_j(D)| < \alpha\right) \geq 2/3 \ , \tag{A.5.9}$$

which is exactly the definition of (α, β)-accuracy for $\widetilde{\mathcal{A}}$. Remark that since $\widetilde{\mathcal{A}}$ is only a post-processing of $\mathcal{A}$, without additional access to the dataset itself, $\widetilde{\mathcal{A}}$ is itself (ϵ, δ)-differentially-private. We have thus constructed an algorithm that is both accurate and private for $n \leq n^*$, which contradicts the result of Lemma A.5.1 when $\beta = \frac{1}{75}$. This proves the existence of a dataset $D \in (\{0,1\}^p)^n$ such that for $D_L^{aug} = \chi_L(D)$, $\mathcal{A}(D_L^{aug})$ is not $(\frac{1}{3}\alpha, \beta)$-accurate on $\mathcal{Q}$, which means that with probability at least $1/3$, there exists a subset $\mathcal{J} \subseteq [p]$ of cardinal $\operatorname{card}\mathcal{J} \geq \lceil \beta p \rceil$ such that

$$\|\mathcal{A}(D_L^{aug}) - q(D_L^{aug})\|_2 \overset{(A.5.7)}{\geq} \sqrt{\sum_{j \in \mathcal{J}} \frac{4L_j^2}{9}} \geq \Omega(\|L\|_2) \ , \tag{A.5.10}$$

where the second inequality comes from the fact that $\operatorname{card}\mathcal{J} \geq \lceil \beta p \rceil = \lceil \frac{p}{75} \rceil$ and our hypothesis on $\sum_{j \in \mathcal{J}} L_j^2$. Notice that when $L_1 = \cdots = L_p = \frac{1}{\sqrt{p}}$, we recover the result of Bassily et al. (2014b), since $\|L\|_2 = 1$ it holds with probability at least $1/3$ that

$$\|\mathcal{A}(D_L^{aug}) - q(D_L^{aug})\|_2 \overset{(A.5.7)}{\geq} \sqrt{\sum_{j \in \mathcal{J}} \frac{4L_j^2}{9}} \geq \sqrt{\frac{4}{9 \times 75}}\|L\|_2 \geq \frac{2}{27} \ , \tag{A.5.11}$$

and in that case, since all L_j's are equal, it indeed holds that $\sqrt{\sum_{j \in \mathcal{J}} L_j^2} = \Omega(\|L\|_2)$. Finally, we remark that the sum of each column of D_L^{aug} is $\sum_{i=1}^n d_{i,j} \geq nL_j$, and as such, we have $\|\sum_{i=1}^n d_i\|_2 = \sqrt{\sum_{j=1}^p (\sum_{i=1}^n d_{i,j})^2} \geq \sqrt{\sum_{j=1}^p n^2 L_j^2} = n\|L\|_2$.

When $n > n^*$. We get the result in that case by augmenting the dataset D^* that we constructed in the first part of this proof. To do so, we follow the steps described by Bassily et al. (2014b) in the proof of their Lemma 5.1. The construction

consists in choosing a vector $c \in \mathcal{X}$, and adding $\lceil \frac{n-n^*}{2} \rceil$ rows with c, and $\lfloor \frac{n-n^*}{2} \rfloor$ rows with $-c$ to the dataset D^*. This results in a dataset D' such that $\| \sum_{i=1}^n d_i \| = \Omega(n^* \|L\|_2) = \Omega(\frac{\sqrt{p}\|L\|_2}{\epsilon})$, since the contributions of rows $-c$ and c (almost) cancel out. The theorem follows from observing that $(\frac{n^*}{n}\alpha, \beta)$-accuracy on this augmented dataset implies (α, β)-accuracy on the original dataset. As such, if an algorithm is both private and $(\frac{n^*}{n}\alpha, \beta)$-accurate on the dataset D', we get a contradiction, which gives the theorem as $\frac{n^*}{n} = \frac{\sqrt{p}}{n\epsilon}$. □

Remark A.5.1. *Without the assumption on the distribution of the L_j's, we can still get an inequality that resembles (A.5.10): $\|\mathcal{A}(D_L^{aug}) - q(D_L^{aug})\|_2 \overset{(A.5.7)}{\geq} \sqrt{\sum_{j\in\mathcal{J}} \frac{4L_j^2}{9}} \geq \frac{2}{27}\frac{L_{\min}}{L_{\max}}\|L\|_2$, with probability at least $1/3$, and we get a result similar to Theorem A.5.1, except with an additional multiplicative factor $L_{\min}/L_{\max}$.*

A.5.3 Lower Bound for Convex Functions

To prove a lower bound for our problem in the convex case, we let $L_1, \cdots, L_p > 0$ and define a dataset $D = \{d_1, \ldots, d_n\}$ taking its values in a set $\mathcal{X} = \prod_{j=1}^p \{\pm L_j\}$. For $\beta > 0$, we consider the problem $(\star')$ with $\mathcal{W} = \mathbb{R}^p$, the convex, smooth and L-coordinate-Lipschitz loss function $\ell(w; d) = -\langle w, d\rangle$ and the convex, separable regularizer $\psi(w) = \frac{\|\sum_{i=1}^n d_i\|_2}{\beta n}\|w\|_2^2$:

$$w^* = \operatorname*{arg\,min}_{w\in\mathbb{R}^p} \left\{ F(w; D) = -\frac{1}{n}\langle w, \textstyle\sum_{i=1}^n d_i\rangle + \frac{\|\sum_{i=1}^n d_i\|_2}{\beta n}\|w\|_2^2 \right\} \ , \tag{A.5.12}$$

To find the solution of (A.5.12), we look for w^* so that the objective's gradient is zero, that is

$$w^* = \frac{\beta}{\|\sum_{i=1}^n d_i\|_2} \sum_{i=1}^n d_i \ , \tag{A.5.13}$$

so that $\|w^*\|_2 = \frac{\beta}{\|\sum_{i=1}^n d_i\|_2}\|\sum_{i=1}^n d_i\|_2 = \beta$. To prove the lower bound, we remark that

$$\begin{aligned}
F(w; D) - F(w^*; D) &= -\frac{1}{n}\langle w - w^*, \textstyle\sum_{i=1}^n d_i\rangle + \frac{\|\sum_{i=1}^n d_i\|}{2\beta n}(\|w\|_2^2 - \|w^*\|_2^2) \\
&= -\frac{1}{n}\langle w - w^*, \frac{\|\sum_{i=1}^n d_i\|}{\beta} w^*\rangle + \frac{\|\sum_{i=1}^n d_i\|}{2\beta n}(\|w\|_2^2 - \|w^*\|_2^2) \\
&= \frac{\|\sum_{i=1}^n d_i\|}{\beta n}\left(\langle w^* - w, w^*\rangle + \frac{1}{2}\|w\|_2^2 - \frac{1}{2}\|w^*\|_2^2\right) \\
&= \frac{\|\sum_{i=1}^n d_i\|}{\beta n}\left(-\langle w, w^*\rangle + \frac{1}{2}\|w\|_2^2 + \frac{1}{2}\|w^*\|_2^2\right) \\
&= \frac{\|\sum_{i=1}^n d_i\|}{2\beta n}\|w - w^*\|_2^2 \ .
\end{aligned} \tag{A.5.14}$$

At this point, we can proceed similarly to Bassily et al. (2014b) to relate this quantity to private estimation of one-way marginals. We let $M = \Omega(\min(n\|L\|_2, \|L\|_2\sqrt{p}/\epsilon))$ and $\mathcal{A}$ be an (ϵ, δ)-differentially private mechanism that outputs a private solution w^{priv} to (A.5.12). Suppose, for the sake of contradiction, that for every dataset D with $\|\sum_{i=1}^n d_i\|_2 \in [M-1; M+1]$, it holds with probability at least 2/3 that

$$\|w^{priv} - w^*\| \neq \Omega(\beta) \ . \tag{A.5.15}$$

We now derive from $\mathcal{A}$ a mechanism $\widetilde{\mathcal{A}}$ to estimate one-way marginals. To do this, $\widetilde{\mathcal{A}}$ runs $\mathcal{A}$ to obtain w^{priv} and outputs $\frac{M}{n\beta} w^{priv}$. We obtain that with probability at least 2/3,

$$\begin{aligned}\|\widetilde{\mathcal{A}}(D) - q(D)\|_2 &= \frac{M}{n\beta}\|w^{priv} - \frac{\beta}{M}\textstyle\sum_{i=1}^n d_i\|_2 \\ &\neq \Omega\left(\frac{M}{n}\right) = \Omega\left(\min\left(\|L\|_2, \frac{\|L\|_2\sqrt{p}}{n\epsilon}\right)\right) \ . \end{aligned} \tag{A.5.16}$$

where $q(D) = \frac{1}{n}\sum_{i=1}^n d_i$. This is in contradiction with Theorem A.5.1. We thus proved that $\|w^{priv} - w^*\| = \Omega(\beta)$, with probability at least 1/3. As a consequence, we now obtain that with probability at least 1/3,

$$\begin{aligned} F(w^{priv}; D) - F(w^*; D) &= \frac{\|\sum_{i=1}^n d_i\|}{2\beta n}\|w^{priv} - w^*\|_2^2 \\ &= \Omega\left(\min\left(\|L\|_2\beta, \frac{\beta\|L\|_2\sqrt{p}}{n\epsilon}\right)\right) \ , \end{aligned} \tag{A.5.17}$$

which gives the desired result on the expectation of $F(w^{priv}; D) - F(w^*; D)$.

Finally, if we do not make any hypothesis on the L_j's distribution, we can directly use the non-augmented dataset constructed by Bun et al. (2014) to prove Lemma A.5.1 (that is the dataset from Theorem A.5.1, rescaled but not augmented). The ℓ_2-norm of the sum of this dataset is $\|\sum_{i=1}^n d_j\|_2 = [M'-1, M'+1]$ with $M' = \Omega\left(\min\left(\frac{L_{\min}}{L_{\max}} n\|L\|_2, \frac{L_{\min}}{L_{\max}}\frac{\sqrt{p}\|L\|_2}{\epsilon}\right)\right)$. This holds since four columns of this dataset out of five have sum of $\pm nL_j$ (for some j's), but no lower bound on the sum of the remaining columns can be derived. Thus, assuming (A.5.15) holds, then (A.5.16) can be rewritten as

$$\begin{aligned}\|\widetilde{\mathcal{A}}(D) - q(D)\|_2 &= \frac{M'}{n\beta}\|w^{priv} - \frac{\beta}{M}\textstyle\sum_{i=1}^n d_i\|_2 \\ &\neq \Omega\left(\frac{M'}{n}\right) = \Omega\left(\min\left(\frac{L_{\min}}{L_{\max}}\|L\|_2, \frac{L_{\min}}{L_{\max}}\frac{\|L\|_2\sqrt{p}}{n\epsilon}\right)\right) \ , \end{aligned} \tag{A.5.18}$$

with probability at least 1/3, which is in contradiction with Remark A.5.1. We thus get an additional factor of $L_{\min}/L_{\max}$ in the lower bound:

$$F(w^{priv}; D) - F(w^*; D) = \frac{\|\sum_{i=1}^n d_i\|}{2\beta n}\|w^{priv} - w^*\|_2^2 \\ = \Omega\left(\min\left(\frac{L_{\min}}{L_{\max}}\|L\|_2\beta, \frac{L_{\min}}{L_{\max}}\frac{\beta\|L\|_2\sqrt{p}}{n\epsilon}\right)\right) \ . \quad \text{(A.5.19)}$$

A.5.4 Lower Bound for Strongly-Convex Functions

To prove a lower bound for strongly-convex functions, we let $\mu_I > 0$, $L_1, \dots, L_p > 0$, $\mathcal{W} = \prod_{j=1}^p [-\frac{L_j}{2\mu_I}, +\frac{L_j}{2\mu_I}]$ and $D = \{d_1, \dots, d_n\} \in \prod_{j=1}^p \{\pm\frac{L_j}{2\mu_I}\}$. We consider the following problem, which fits in our setting:

$$w^* = \arg\min_{w\in\mathbb{R}^p} \left\{ F(w; D) = \frac{\mu_I}{2n}\sum_{i=1}^n \|w - d_i\|_2^2 + i_{\mathcal{W}}(w) \right\} \ . \quad \text{(A.5.20)}$$

where $i_{\mathcal{W}}$ is the (separable) characteristic function of the set $\mathcal{W}$. Since $\psi = i_{\mathcal{W}}$ is the characteristic function of a box-set, the proximal operator is equal to the projection on $\mathcal{W}$ and DP-CD iterates are thus guaranteed to remain in $\mathcal{W}$. Therefore, regularity assumptions on f only need to hold on $\mathcal{W}$. The loss function $\ell(w; d_i) = \frac{\mu_I}{2}\|w - d_i\|_2^2$ is L-coordinate-Lipschitz on $\mathcal{W}$ since, for $w \in \mathcal{W}$ and $j \in [p]$, the triangle inequality gives:

$$|\nabla_j \ell(w; d_i)| \leq \mu_I(|w_j| + |d_{i,j}|) \leq \mu_I\left(\frac{L_j}{2\mu_I} + \frac{L_j}{2\mu_I}\right) \leq L_j \ . \quad \text{(A.5.21)}$$

This loss is also μ_I-strongly convex *w.r.t.*, ℓ_2-norm since for $w, w' \in \mathcal{W}$,

$$\ell(w; d_i) = \frac{\mu_I}{2}\|w - d_i\|_2^2 \\ = \frac{\mu_I}{2}\|w' - d_i + w - w'\|_2^2 \\ = \frac{\mu_I}{2}\left(\|w' - d_i\|_2^2 + 2\langle w' - d_i, w - w'\rangle + \|w - w'\|_2^2\right) \ , \quad \text{(A.5.22)}$$

which is exactly μ_I-strong convexity since $\ell(w'; d_i) = \frac{\mu_I}{2}\|w' - d_i\|_2^2$ and $\nabla\ell(w'; d_i) = \mu_I(w - d_i)$. The minimum of the objective function in (A.5.20) is attained at

$$w^* = \frac{1}{n}\sum_{i=1}^n d_i = q(D) \in \mathcal{W} \ .$$

The excess risk of F is thus

$$F(w;D) - F(w^*) = \frac{\mu_I}{2n}\sum_{i=1}^{n} \|w - d_i\|_2^2 - \|w^* - d_i\|_2^2 \tag{A.5.23}$$

$$= \frac{\mu_I}{2n}\sum_{i=1}^{n} \|w\|^2 - \|w^*\|^2 + 2\langle d_i, w^* - w\rangle \tag{A.5.24}$$

$$= \frac{\mu_I}{2}\|w\|^2 - \frac{1}{2}\|w^*\|^2 + \langle w^*, w^* - w\rangle \tag{A.5.25}$$

$$= \frac{\mu_I}{2}\|w - q(D)\|_2^2 \ . \tag{A.5.26}$$

It remains to apply Theorem A.5.1 to obtain that, with probability at least 1/3,

$$F(w^{priv};D) - F(w^*) = \Omega\left(\min\left(\frac{\|L\|_2^2}{\mu_I}, \frac{\|L\|_2^2 p}{\mu_I n^2 \epsilon^2}\right)\right) \ , \tag{A.5.27}$$

which gives the lower bound on the expected value of $F(w^{priv};D) - F(w^*)$. Note that without the additional assumption on the distribution of the L_j's, Remark A.5.1 directly gives the result with an additional multiplicative factor $(L_{\min}/L_{\max})^2$:

$$F(w^{priv};D) - F(w^*) = \Omega\left(\min\left(\frac{L_{\min}^2}{L_{\max}^2}\frac{\|L\|_2^2}{\mu_I}, \frac{L_{\min}^2}{L_{\max}^2}\frac{\|L\|_2^2 p}{\mu_I n^2 \epsilon^2}\right)\right) \ , \tag{A.5.28}$$

with probability at least 1/3.

Appendix B

Proofs of Chapter 5

B.1 Proof of Privacy

Theorem B.1.1. *5.3.1 Let $\epsilon, \delta \in (0, 1]$. Algorithm 5.3.1 with noise scales $\lambda_j = \lambda'_j = \frac{8L_j}{n\epsilon}\sqrt{T\log(1/\delta)}$ is (ϵ, δ)-differentially private.*

Proof. In each iteration of Algorithm 5.3.1, the data is accessed twice: once to choose the coordinate and once to compute the private gradient. In total, data is thus queried $2T$ times.

Let $\lambda_j = \lambda'_j = \frac{2L_j}{n\epsilon'}$. For $j \in [p]$, the gradient's j-th entry has sensitivity $2L_j$. Thus, by the report noisy max mechanism (Dwork and Roth, 2014), the greedy choice of j is ϵ'-DP. By the Laplace mechanism (Dwork and Roth, 2014), computing the corresponding gradient coordinate is also ϵ'-DP.

The advanced composition theorem for differential privacy thus ensures that the $2T$-fold composition of these mechanisms is (ϵ, δ)-DP for $\delta > 0$ and

$$\epsilon = \sqrt{4T\log(1/\delta)}\epsilon' + 2T\epsilon'(\exp(\epsilon') - 1) \ , \tag{B.1.1}$$

where we recall that $\epsilon' = \frac{2L_j}{n\lambda_j} = \frac{2L_j}{n\lambda'_j}$ for all $j \in [p]$. When $\epsilon \leq 1$, we can give a simpler expression (see Corollary 3.21 of Dwork and Roth, 2014): with $\epsilon' = \epsilon/4\sqrt{T\log(1/\delta)}$, Algorithm 5.3.1 is (ϵ, δ)-DP for $\lambda_j = \lambda'_j = 8L_j\sqrt{T\log(1/\delta)}/n\epsilon$. □

B.2 Proof of Utility

In this section, we prove Theorem 5.3.2 and Theorem 5.3.3, giving utility upper bounds for DP-GCD. We obtain these high-probability results through a careful ex-

amination of the properties of DP-GCD's iterates, and obtain high-probability results by using concentration inequalities (see Appendix B.2.1).

In Appendix B.2.2, we prove a general descent lemma, which implies that iterates of DP-GCD converge (with high probability) to a neighborhood of the optimum. This property is proven rigorously in Appendix B.2.3 (b), and we give the utility results for general convex functions in Appendix B.2.3 (c). Under the additional assumption that the objective is strongly convex, we prove better utility bounds in Appendix B.2.4. These bounds follow from a key lemma (see Appendix B.2.4 (a)), which implies linear convergence to a neighborhood of the optimum. We then use this result in two settings, obtaining two different rates: first in a general setting (in Appendix B.2.4 (b)), then under the additional assumption that the problem's solution is quasi-sparse (in Appendix B.2.4 (c)).

B.2.1 Concentration Lemma

To prove high-probability utility results, we first bound (in Lemma B.2.1) the probability for a sum of squared Laplacian variables to exceed a given threshold.

Lemma B.2.1. *Let $K > 0$ and $\lambda_1, \dots, \lambda_K > 0$. Define $X_k \sim \mathrm{Lap}(\lambda_k)$ and $\lambda_{\max} = \max_{k\in[K]} \lambda_k$. For any $\beta > 0$, it holds that*

$$\mathbb{P}\left(\sum_{k=1}^{K} X_k^2 \geq \beta\right) \leq 2^K \exp\left(-\frac{\sqrt{\beta}}{2\lambda_{\max}}\right) \quad . \tag{B.2.1}$$

Proof. We first remark that $(\sum_{k=1}^{K} |X_k|)^2 = \sum_{k=1}^{K}\sum_{k'=1}^{K} |X_k||X_{k'}| \geq \sum_{k=1}^{K} X_k^2$. Therefore

$$\mathbb{P}\left(\sum_{k=1}^{K} X_k^2 \geq a^2\right) \leq \mathbb{P}\left(\Big(\sum_{k=1}^{K} |X_k|\Big)^2 \geq a^2\right) = \mathbb{P}\left(\Big(\sum_{k=1}^{K} |X_k|\Big) \geq a\right) \quad . \tag{B.2.2}$$

Chernoff's inequality now gives, for any $\gamma > 0$,

$$\mathbb{P}\left(\sum_{k=1}^{K} |X_k| \geq a\right) \leq \exp(-\gamma a)\mathbb{E}\left[\exp(\gamma \sum_{k=1}^{K} |X_k|)\right] \quad . \tag{B.2.3}$$

By the properties of the exponential and the X_k's independence, we can rewrite the inequality as

$$\begin{aligned}\mathbb{P}\left(\sum_{k=1}^{K} |X_k| \geq a\right) &\leq \exp(-\gamma a)\mathbb{E}\left[\prod_{k=1}^{K} \exp\Big(\gamma|X_k|\Big)\right] \\ &= \exp(-\gamma a)\prod_{k=1}^{K} \mathbb{E}\Big[\exp\Big(\gamma|X_k|\Big)\Big] \quad . \end{aligned} \tag{B.2.4}$$

We can now compute the expectation of $\exp(\gamma|X_k|)$ for $k \in [K]$,

$$\begin{aligned}\mathbb{E}\Big[\exp\Big(\gamma|X_k|\Big)\Big] &= \frac{1}{2\lambda_k}\int_{-\infty}^{+\infty}\exp(\gamma|x|)\exp(-\frac{|x|}{\lambda_k})dx \\ &= \frac{1}{\lambda_k}\int_0^{+\infty}\exp\Big((\gamma-\frac{1}{\lambda_k})x\Big)dx \ . \end{aligned} \tag{B.2.5}$$

We choose $\gamma = 1/2\lambda_{\max}$, such that $\gamma \leq 1/2\lambda_k$ for all $k \in [K]$ and obtain

$$\mathbb{E}\Big[\exp\Big(\gamma|X_k|\Big)\Big] = \frac{1}{\lambda_k}\frac{1}{\frac{1}{\lambda_k}-\gamma} = \frac{1}{1-\gamma\lambda_k} \leq 2 \ . \tag{B.2.6}$$

Plugging everything together, we have proved that

$$\mathbb{P}\left(\sum_{k=1}^K X_k^2 \geq a^2\right) \leq \mathbb{P}\left(\sum_{k=1}^K |X_k| \geq a\right) \leq 2^K\exp(-\frac{a}{2\lambda_{\max}}) \ , \tag{B.2.7}$$

and taking $a = \sqrt{\beta}$ gives the result. □

B.2.2 Descent Lemma

We now prove a noisy descent lemma for DP-GCD (Lemma B.2.2). This lemma bounds the suboptimality $f(w^{t+1}) - f(w^*)$ at time $t+1$ as a function of the suboptimality $f(w^t) - f(w^*)$ at time t, of the gradient's largest entry and of the noise. At this point, we remark that when the gradient is large enough, it is very probable that $\frac{1}{8}\|\nabla f(w^t)\|^2_{M^{-1},\infty} \geq \frac{1}{2M_j}|\eta_j^t|^2 + \frac{1}{2M_j}|\chi_j^t|^2 + \frac{1}{4M_{j^*}}|\chi_{j^*}^t|^2$: this implies that the value of the objective function decreases with high probability, even under the presence of noise. This observation will be crucial for proving utility for general convex functions.

Lemma B.2.2. *Let $t \geq 0$ and $w^t, w^{t+1} \in \mathbb{R}^p$ two consecutive iterates of Algorithm 5.3.1, with $\gamma_j = 1/M_j$ and λ_j, λ_j' chosen as in Theorem 5.3.1 to ensure ϵ, δ-DP. We denote by $j \in [p]$ the coordinate chosen at this step t, and by*

$$j^* = \arg\max_{j\in[p]} |\nabla_j f(w^t)|/\sqrt{M_j}$$

the coordinate that would have been chosen without noise. The following inequality holds

$$\begin{aligned} f(w^{t+1}) - f(w^*) \leq f(w^t) - f(w^*) - \frac{1}{8}\|\nabla f(w^t)\|^2_{M^{-1},\infty} \\ + \frac{1}{2M_j}|\eta_j^t|^2 + \frac{1}{2M_j}|\chi_j^t|^2 + \frac{1}{4M_{j^*}}|\chi_{j^*}^t|^2 \ . \end{aligned} \tag{B.2.8}$$

Proof. The smoothness of f gives a first inequality

$$f(w^{t+1}) \leq f(w^t) + \langle \nabla f(w^t), w^{t+1} - w^t \rangle + \frac{1}{2}\|w^{t+1} - w^t\|_M^2 \tag{B.2.9}$$

$$= f(w^t) - \frac{1}{M_j}\nabla_j f(w^t)(\nabla_j f(w^t) + \eta_j^t) + \frac{1}{2M_j}(\nabla_j f(w^t) + \eta_j^t)^2 \tag{B.2.10}$$

$$= f(w^t) - \frac{1}{M_j}\nabla_j f(w^t)^2 - \frac{1}{M_j}\nabla_j f(w^t)\eta_j^t + \frac{1}{2M_j}(\nabla_j f(w^t))^2 + \frac{1}{M_j}\nabla_j f(w^t)\eta_j^t + \frac{1}{2M_j}(\eta_j^t)^2 \tag{B.2.11}$$

$$= f(w^t) - \frac{1}{2M_j}\nabla_j f(w^t)^2 + \frac{1}{2M_j}(\eta_j^t)^2 \ . \tag{B.2.12}$$

We will make the noisy gradient appear, so as to use the noisy greedy rule. To do so, we remark that the classical inequality $(a+b)^2 \leq 2a^2 + 2b^2$ for any $a, b \in \mathbb{R}$ implies that $-a^2 \leq -\frac{1}{2}(a+b)^2 + b^2$. Applied with $a = \nabla_j f(w^t)/\sqrt{M_j}$ and $b = \chi_j^t/\sqrt{M_j}$, this results in

$$-\frac{1}{2M_j}\nabla_j f(w^t)^2 \leq -\frac{1}{4M_j}(\nabla_j f(w^t) + \chi_j^t)^2 + \frac{1}{2M_j}(\chi_j^t)^2 \ . \tag{B.2.13}$$

And, by the noisy greedy rule, $\frac{1}{\sqrt{M_{j^*}}}|\nabla_{j^*} f(w^t) + \chi_{j^*}^t| \leq \frac{1}{\sqrt{M_j}}|\nabla_j f(w^t) + \chi_j^t|$. We replace in (B.2.13) and use the inequality $-a^2 \leq -\frac{1}{2}(a+b)^2 + b^2$ with $a = (\nabla_{j^*} f(w^t) + \chi_{j^*})/\sqrt{M_{j^*}}$ and $b = -\chi_{j^*}/\sqrt{M_{j^*}}$ to obtain

$$-\frac{1}{2M_j}\nabla_j f(w^t)^2 \leq -\frac{1}{4M_{j^*}}(\nabla_{j^*} f(w^t) + \chi_{j^*}^t)^2 + \frac{1}{2M_j}(\chi_j^t)^2 \tag{B.2.14}$$

$$\leq -\frac{1}{8M_{j^*}}(\nabla_{j^*} f(w^t))^2 + \frac{1}{4M_{j^*}}(\chi_{j^*}^t)^2 + \frac{1}{2M_j}(\chi_j^t)^2 \ . \tag{B.2.15}$$

The result follows from (B.2.12) and $\frac{1}{M_{j^*}}(\nabla_{j^*} f(w^t))^2 = \|\nabla f(w^t)\|_{M^{-1},\infty}^2$. □

B.2.3 Utility for General Convex Functions

In this section, we derive an upper bound on the utility of DP-GCD for convex objective functions. First, we use convexity of f to upper bound the decrease described in Lemma B.2.2. This gives Lemma B.2.3 in Appendix B.2.3 (a), where the suboptimality gap $f(w^{t+1}) - f(w^*)$ at time $t+1$ is upper bound by a function of the suboptimality gap $f(w^t) - f(w^*)$ at time t and the noise injected in step t. The novelty of our analysis lies in Lemma B.2.4, where examine the decrease of the objective. Specifically, we show that either (i) $f(w^t)$ is far from its minimum, and the suboptimality gap decreases with high probability, either (ii) $f(w^t)$ is close to its

minimum, then all future iterates of DP-GCD will remain in a ball whose radius is determined by the variance of the noise. This observation is essential for proving the utility results stated in Section 5.3.3.

B.2.3 (a) Descent Lemma for Convex Functions

Lemma B.2.3. *Under the hypotheses of Lemma B.2.2, for a convex objective function f, we have*

$$f(w^{t+1}) - f(w^*) \leq f(w^t) - f(w^*) - \frac{(f(w^t) - f(w^*))^2}{8\|w^t - w^*\|_{M,1}^2} + \frac{1}{2M_j}|\eta_j^t|^2 + \frac{1}{2M_j}|\chi_j^t|^2 + \frac{1}{4M_{j^*}}|\chi_{j^*}^t|^2 \ . \tag{B.2.16}$$

Proof. Since f is convex, it holds that

$$f(w^*) \geq f(w^t) + \langle \nabla f(w^t), w^* - w^t \rangle \ . \tag{B.2.17}$$

After reorganizing the terms, we can upper bound them using Hölder's inequality

$$f(w^t) - f(w^*) \leq \langle \nabla f(w^t), w^t - w^* \rangle \tag{B.2.18}$$

$$\leq \|\nabla f(w^t)\|_{M^{-1},\infty} \|w^t - w^*\|_{M,1} \ , \tag{B.2.19}$$

where the second inequality holds since $\|\cdot\|_{M,1}$ and $\|\cdot\|_{M^{-1},\infty}$ are conjugate norms. We now divide (B.2.19) by $\|w^t - w^*\|_{M,1}$, square it and reorganize to get $-\|\nabla f(w^t)\|_{M^{-1},\infty}^2 \leq -\frac{(f(w^t)-f(w^*))^2}{\|w^t-w^*\|_{M,1}^2}$. Replacing in Lemma B.2.2 gives the result. □

B.2.3 (b) Key Lemma on the Behavior of DP-GCD's Iterates

Now that we have an inequality in the form of Lemma B.2.3, we prove that iterates of DP-GCD converge to a vicinity of the optimum. In the general lemma below, think of ξ_t as $f(w^t) - f(w^*)$ and of β as the variance of the term. This result will be combined with Lemma B.2.1 to obtain high-probability bounds.

Lemma B.2.4. *Let $\{c_t\}_{t\geq 0}$ and $\{\xi_t\}_{t\geq 0}$ be two sequences of positive values that satisfy, for all $t \geq 0$,*

$$\xi_{t+1} \leq \xi_t - \frac{\xi_t^2}{c_t} + \beta, \tag{B.2.20}$$

such that if $\xi_t \leq \xi_0$ then $c_t \leq c_0$. Assume that $\beta \leq c_0$ and $\xi_0 \geq 2\sqrt{\beta c_0}$. Then:

1. *For all $t > 0$, $c_t \leq c_0$, and there exists $t^* > 0$ such that $\xi_{t+1} \leq \xi_t$ if $t < t^*$ and $\xi_t \leq 2\sqrt{\beta c_0}$ if $t \geq t^*$.*

2. *For all* $t \geq 1$, $\xi_t \leq \frac{c_0}{t} + 2\sqrt{\beta c_0}$.

Proof. 1. Assume that for $t \geq 0$, $\sqrt{\beta c_0} \leq \xi_t \leq \xi_0$. Then,

$$\xi_{t+1} \leq \xi_t - \frac{\xi_t^2}{c_t} + \beta \leq \xi_t - \frac{\sqrt{\beta c_0}^2}{c_0} + \beta = \xi_t \ , \tag{B.2.21}$$

where the second inequality comes from $\xi_t \geq \sqrt{\beta c_0}$ and $\xi_t \leq \xi_0$ (which implies $c_t \leq c_0$). We now define the following value t^*, which defines the point of rupture between two regimes for ξ_t:

$$t^* = \min \left\{ t \geq 0 \,\middle|\, \xi_t \leq \sqrt{\beta c_0} \right\} \ . \tag{B.2.22}$$

Let $t < t^*$, assume that $\xi_t \leq \xi_0$, then (B.2.21) holds, that is $\xi_{t+1} \leq \xi_t \leq \xi_0$. By induction, it follows that for all $t < t^*$, $\xi_{t+1} \leq \xi_t \leq \xi_0$ and $c_t \leq c_0$.

Remark now that $\xi_{t^*} \leq \sqrt{\beta c_0}$, we prove by induction that ξ_t stays under $2\sqrt{\beta c_0}$ for $t \geq t^*$. Assume that for $t \geq t^*$, $\xi_t \leq 2\sqrt{\beta c_0}$. Then, there are two possibilities. If $\xi_t \leq \sqrt{\beta c_0}$, then

$$\xi_{t+1} \leq \xi_t - \frac{\xi_t^2}{c_t} + \beta \leq \sqrt{\beta c_0} + \beta \leq 2\sqrt{\beta c_0} \ , \tag{B.2.23}$$

and $\xi_{t+1} \leq 2\sqrt{\beta c_0}$. Otherwise, $\sqrt{\beta c_0} \leq \xi_t \leq 2\sqrt{\beta c_0} \leq \xi_0$ and (B.2.21) holds, which gives $\xi_{t+1} \leq \xi_t \leq 2\sqrt{\beta c_0}$. We proved that for $t \geq t^*$, $\xi_t \leq 2\sqrt{\beta c_0}$, which concludes the proof of the first part of the lemma.

2. We start by proving this statement for $0 < t < t^* - 1$. Define $\omega = \frac{2u}{c_0}$ and $u = \sqrt{\beta c_0}$. The assumption on ξ_t implies, by the first part of the lemma, $\xi_{t+1} \leq \xi_t - \frac{\xi_t^2}{c_t} + \beta \leq \xi_t - \frac{\xi_t^2}{c_0} + \beta$, which can be rewritten

$$\xi_{t+1} - u \leq (1-\omega)(\xi_t - u) - \frac{(\xi_t - u)^2}{c_0} \ , \tag{B.2.24}$$

since $(1-\omega)(\xi_t - u) - \frac{(\xi_t - u)^2}{c_0} = \xi_t - \omega\xi_t - u + \omega u - \frac{\xi_t^2}{c_0} - \frac{2\xi_t u}{c_0} - \frac{u^2}{c_0} = \xi_t - \frac{\xi_t^2}{c_0} - u + \omega u - \frac{u^2}{c_0}$, and $\omega u - \frac{u^2}{c_0} = \frac{u^2}{c_0} = \beta$. Since $t < t^* - 1$, $\xi_{t+1} - u > 0$ and $\xi_t - u > 0$, we can thus divide (B.2.24) by $(\xi_{t+1} - u)(\xi_t - u)$ to obtain

$$\frac{1}{\xi_t - u} \leq \frac{1-\omega}{\xi_{t+1} - u} - \frac{\xi_t - u}{(\xi_{t+1} - u)c_0} \leq \frac{1-\omega}{\xi_{t+1} - u} - \frac{1}{c_0} \leq \frac{1}{\xi_{t+1} - u} - \frac{1}{c_0} \ , \tag{B.2.25}$$

where the second inequality comes from $\xi_{t+1} - u \leq \xi_t - u$ from the first part of the lemma. By applying this inequality recursively and taking the inverse of the result, we obtain the desired resuld $\xi_t \leq \frac{c_0}{t} + \sqrt{\beta c_0} \leq \frac{c_0}{t} + 2\sqrt{\beta c_0}$ for all $0 < t < t^*$.

For $t \geq t^*$, we have already proved that $\xi_t \leq 2\sqrt{\beta c_0} \leq \frac{c_0}{t} + 2\sqrt{\beta c_0}$, which concludes our proof. □

B.2.3 (c) Convex Utility Result

Theorem B.2.1. *5.3.2 (Convex Case) Let $\epsilon, \delta \in (0,1]$. Assume $\ell(\cdot; d)$ is a convex and L-coordinate-Lipschitz loss function for all $d \in \mathcal{X}$, and f is M-coordinate-smooth. Define $\mathcal{W}^*$ the set of minimizers of f, and f^* the minimum of f. Let $w_{priv} \in \mathbb{R}^p$ be the output of Algorithm 5.3.1 with step sizes $\gamma_j = 1/M_j$, and noise scales $\lambda_1, \dots, \lambda_p, \lambda'_1, \dots, \lambda'_p$ set as in Theorem 5.3.1 (with T chosen below) to ensure (ϵ, δ)-DP. Then, the following holds for $\zeta \in (0,1]$:*

$$f(w_{priv}) - f(w^*) \leq \frac{8R_M^2}{T} + \sqrt{32R_M^2\beta} \ , \tag{B.2.26}$$

where $\beta = \frac{2\lambda_{\max}^2}{M_{\min}} \log(\frac{8Tp}{\zeta})^2$, and $R_M = \max_{w \in \mathbb{R}^p} \min_{w^* \in \mathcal{W}^*} \left\{ \|w - w^*\|_{M,1} \mid f(w) \leq f(w^0) \right\}$. If we set $T = \left(\frac{n^2\epsilon^2 R_M^2 M_{\min}}{2^7 L_{\max}^2 \log(1/\delta)} \right)^{1/3}$, then with probability at least $1 - \zeta$,

$$f(w^T) - f(w^0) = \widetilde{O}\Big(\frac{R_M^{4/3} L_{\max}^{2/3} \log(p/\zeta)}{M_{\min}^{1/3} n^{2/3} \epsilon^{2/3}} \Big) \ . \tag{B.2.27}$$

Proof. Let $\xi_t = f(w^t) - f(w^*)$. We upper bound the following probability by the union bound, and the fact that for $t \geq 0$, the events E_j^t : "coordinate j is updated at step t" for $j \in [p]$ partition the probability space:

$$\begin{aligned}
&\mathbb{P}\left(\exists t, \xi_{t+1} \geq \xi_t - \frac{\xi_t^2}{8\|w^t - w^*\|_{M,1}^2} + \beta \right) \\
&\leq \sum_{t=0}^{T-1} \mathbb{P}\left(\xi_{t+1} \geq \xi_t - \frac{\xi_t^2}{8\|w^t - w^*\|_{M,1}^2} + \beta \right) &\text{(B.2.28)} \\
&= \sum_{t=0}^{T-1} \sum_{j=1}^{p} \mathbb{P}\left(\xi_{t+1} \geq \xi_t - \frac{\xi_t^2}{8\|w^t - w^*\|_{M,1}^2} + \beta \ \wedge \ E_j^t \right) \ . &\text{(B.2.29)}
\end{aligned}$$

Lemma B.2.3 gives $\xi_{t+1} \leq \xi_t - \frac{\xi_t^2}{8\|w^t - w^*\|_{M,1}^2} + \frac{1}{2M_j}|\eta_j^t|^2 + \frac{1}{2M_j}|\chi_j^t|^2 + \frac{1}{4M_{j^*}}|\chi_{j^*}^t|^2$. We thus have the following upper bound:

$$\begin{aligned}
&\mathbb{P}\left(\exists t, \xi_{t+1} \geq \xi_t - \frac{1}{8\|w^t - w^*\|_{M,1}^2} \xi_t^2 + \beta \right) \\
&\leq \sum_{t=0}^{T-1} \sum_{j=1}^{p} \mathbb{P}\left(\frac{|\eta_j^t|^2}{2M_j} + \frac{|\chi_j^t|^2}{2M_j} + \frac{|\chi_{j^*}^t|^2}{4M_{j^*}} \geq \beta \right) &\text{(B.2.30)} \\
&\leq \sum_{t=0}^{T-1} \sum_{j=1}^{p} \mathbb{P}\left(|\eta_j^t|^2 + |\chi_j^t|^2 + |\chi_{j^*}^t|^2 \geq 2M_{\min}\beta \right) \ . &\text{(B.2.31)}
\end{aligned}$$

By Lemma B.2.1 with $X_1 = \eta_j^t \sim \text{Lap}(\lambda_j)$, $X_2 = \chi_j^t \sim \text{Lap}(\lambda'_j)$ and $X_3 = \chi_{j^*}^t \sim \text{Lap}(\lambda'_{j^*})$, it holds that

$$\mathbb{P}\left(|\eta_j^t|^2 + |\chi_j^t|^2 + |\chi_{j^*}^t|^2 \geq 2M_{\min}\beta\right) \leq 8\exp\left(-\frac{\sqrt{2M_{\min}\beta}}{2\lambda_{\max}}\right) = \frac{\zeta}{Tp} \ , \tag{B.2.32}$$

where the last equality comes from $\beta = \frac{2\lambda_{\max}^2}{M_{\min}}\log(\frac{8Tp}{\zeta})^2$. We have proved that

$$\mathbb{P}\left(\exists t, \xi_{t+1} \geq \xi_t - \frac{\xi_t^2}{8\|w^t - w^*\|_{M,1}^2} + \beta\right) \leq \sum_{t=0}^{T-1}\sum_{j=1}^{p}\frac{\zeta}{Tp} = \zeta \ . \tag{B.2.33}$$

We now use our Lemma B.2.4, with $\xi_t = f(w^t) - f(w^*)$; $c_0 = 8R_M^2$ and $c_t = 8\|w^t - w^*\|_{M,1}^2$ for $t > 0$; and $\beta = \frac{2\lambda_{\max}^2}{M_{\min}}\log(\frac{8Tp}{\zeta})^2$. These values satisfies the assumptions of Lemma B.2.4 since, by the definition of R_M, it holds that $c_t \leq c_0$ whenever $\xi_t \leq \xi_0$ (*i.e.*, $f(w^t) - f(w^*) \leq f(w^0) - f(w^*)$). Additionally, $f(w^0) - f(w^*) \geq \sqrt{32R_M^2\beta}$, therefore $f(w^0) - f(w^*) \geq 2\sqrt{\beta c_0}$, and $\beta \leq c_0$.

We obtain the result, with probability at least $1 - \zeta$:

$$f(w^t) - f(w^0) \leq \frac{c_0}{t} + 2\sqrt{\beta c_0} = \frac{8R_M^2}{t} + \frac{64R_M L_{\max}\log(8Tp/\zeta)\sqrt{T\log(1/\delta)}}{\sqrt{M_{\min}}n\epsilon} \ . \tag{B.2.34}$$

For $T = \frac{R_M^{2/3}M_{\min}^{1/3}n^{2/3}\epsilon^{2/3}}{4L_{\max}^{2/3}\log(1/\delta)^{1/3}}$, we obtain that, with probability at least $1 - \zeta$,

$$f(w^t) - f(w^0) \leq \frac{64R_M^{4/3}L_{\max}^{2/3}\log(1/\delta)^{1/3}}{M_{\min}^{1/3}n^{2/3}\epsilon^{2/3}}\log\left(\frac{pR_M^{2/3}M_{\min}^{1/3}n^{2/3}\epsilon^{2/3}}{4\zeta L_{\max}^{2/3}\log(1/\delta)^{1/3}}\right) \ , \tag{B.2.35}$$

which is the result of the theorem. □

B.2.4 Utility for Strongly-Convex Functions

B.2.4 (a) A Key Inequality for Strongly-Convex Functions

We now prove a link between f's largest gradient entry and the suboptimality gap, under the assumption that there exists a unique minimizer w^* of f that is (α, τ)-quasi-sparse. Note that this assumption is not restrictive in general as any vector in $\mathbb{R}^p$ is $(0, p)$-quasi-sparse, and for any τ there exists $\alpha > 0$ such that the vector is (α, τ)-quasi-sparse. We will denote by $\mathcal{W}_{\tau,\alpha} \subseteq \mathbb{R}^p$ the set of (α, τ)-quasi-sparse vectors of $\mathbb{R}^p$:

$$\mathcal{W}_{\tau,\alpha} = \{w \in \mathbb{R}^p \mid |\{j \in [p] \mid |w_j| \geq \alpha\}| \leq \tau\} \ . \tag{B.2.36}$$

When $\alpha = 0$, we simply write $\mathcal{W}_\tau = \mathcal{W}_{\tau,0}$, that is the set of τ-sparse vectors. We also define the associated thresholding operator π_α, that puts to 0 the coordinates that are smaller than α, "projecting" vectors from $\mathcal{W}_{\tau,\alpha}$ to $\mathcal{W}_\tau$, *i.e.,* for $w \in \mathbb{R}^p$,

$$\pi_\alpha(w) = \begin{cases} 0 & \text{if } |w_j| \leq \alpha \ , \\ w_j & \text{otherwise } . \end{cases} \tag{B.2.37}$$

Importantly, restricting a function to τ-sparse vectors changes its strong-convexity parameter. Let $\tau \geq 0$ and $q \in \{1, 2\}$, we say a function is $\mu^{(\tau)}_{M,q}$-strongly-convex when restricted to τ-sparse vectors if for all τ-sparse vectors $v, w \in \mathcal{W}_\tau$,

$$f(w) \geq f(v) + \langle \nabla f(v), w - v \rangle + \frac{\mu^{(\tau)}_{M,q}}{2} \|w - v\|^2_{M,q} \ . \tag{B.2.38}$$

Remark that when $\tau \geq p$, we recover the usual strong-convexity parameters. The parameters *w.r.t.,* ℓ_1- and ℓ_2-norms can be compared using the following inequality (Fang et al., 2020), for all $\tau \geq 0$,

$$\frac{1}{\tau}\mu^{(\tau)}_{M,2} \leq \mu^{(\tau)}_{M,1} \leq \mu^{(\tau)}_{M,2} \ . \tag{B.2.39}$$

We are ready to prove Lemma B.2.5.

Lemma B.2.5. *Let $f : \mathbb{R}^p \to \mathbb{R}$ be a function that is M-coordinate-smooth, and $\mu^{(\tau)}_{M,1}$-strongly-convex* w.r.t., *$\|\cdot\|_{M,1}$ when restricted to τ-sparse vectors, for $\tau \geq 0$. Assume that the unique minimizer w^* of f is (τ, α)-quasi-sparse, for $\alpha, \tau \geq 0$. Let $w^t \in \mathbb{R}^p$ be a t-sparse vector for some $t \geq 0$. Then we have*

$$-\frac{1}{2}\|\nabla f(w^t)\|_{M^{-1},\infty} \leq -\mu^{(t+\tau)}_{M,1}(f(w^t) - f(w^*)) + \frac{1}{2}M_{\max}\mu^{(t+\tau)}_{M,1}(p - \tau)\alpha^2 \ . \tag{B.2.40}$$

Proof. Let $w^t \in \mathbb{R}^p$ be a t-sparse vector. Remark that w^* is (α, τ)-quasi-sparse, meaning that $\pi_\alpha(w^*)$ is τ-sparse. The union of w^t and $\pi_\alpha(w^*)$'s supports ($\operatorname{supp}(w^t)$ and $\operatorname{supp}(\pi_\alpha(w^*))$) thus satisfies $|\operatorname{supp}(w) \cup \operatorname{supp}(\pi_\alpha(w^*))| \leq t + \tau$. As the function f is $\mu^{(t+\tau)}_{M,1}$-strongly-convex with respect to $\|\cdot\|_{M,1}$ and $t + \tau$ sparse vector,

$$f(\pi_\alpha(w)) \geq f(w^t) + \langle \nabla f(w^t), \pi_\alpha(w) - w^t \rangle + \frac{\mu^{(t+\tau)}_{M,1}}{2}\|\pi_\alpha(w) - w^t\|^2_{M,1} \ . \tag{B.2.41}$$

Since $\pi_\alpha : \mathcal{W}_{\tau,\alpha} \to \mathcal{W}_{\tau,0}$ is surjective, minimizing this equation for $w \in \mathcal{W}_{\tau,\alpha}$ on both sides gives

$$\begin{aligned} \inf_{w \in \mathcal{W}_\tau} f(w) &\geq f(w^t) - \sup_{w \in \mathcal{W}_{\tau,\alpha}} \left\{ \langle -\nabla f(w^t), w^t - \pi_\alpha(w) \rangle - \frac{\mu^{(t+\tau)}_{M,1}}{2}\|\pi_\alpha(w) - w^t\|^2_{M,1} \right\} \\ &\geq f(w^t) - \sup_{w \in \mathbb{R}^p} \left\{ \langle -\nabla f(w^t), w^t - w \rangle - \frac{\mu^{(t+\tau)}_{M,1}}{2}\|w - w^t\|^2_{M,1} \right\} \ . \end{aligned} \tag{B.2.42}$$

The second term corresponds to the conjugate of the function $\frac{1}{2}\|\cdot\|^2_{M,1}$, that is $\frac{1}{2}\|\cdot\|^2_{M^{-1},\infty}$ (Boyd and Vandenberghe, 2004). This gives

$$\inf_{w\in\mathcal{W}_\tau} f(w) \geq f(w^t) - \left(\frac{\mu^{(t+\tau)}_{M,1}}{2}\|\cdot\|_1^2\right)^* (-\nabla f(w')) \tag{B.2.43}$$

$$= f(w^t) - \frac{1}{2\mu^{(t+\tau)}_{M,1}}\|\nabla f(w')\|^2_{M^{-1},\infty} \ . \tag{B.2.44}$$

Finally, w^* is the minimizer of f (which is convex), thus $\nabla f(w^*) = 0$. The smoothness of f gives, for any $w \in \mathbb{R}^p$, $f(w) \leq f(w^*) + \frac{1}{2}\|w - w^*\|^2_{M,2}$. Hence

$$\inf_{w\in\mathcal{W}_\tau} f(w) \leq f(w^*) + \inf_{w\in\mathcal{W}_\tau} \frac{1}{2}\|w - w^*\|^2_{M,2}. \leq f(w^*) + \frac{1}{2}\|\pi_\alpha(w^*) - w^*\|^2_{M,2} \ , \tag{B.2.45}$$

where the second inequality comes from $\pi_\alpha(w^*) \in \mathcal{W}_\tau$, since $w^* \in \mathcal{W}_{\tau,\alpha}$. It remains to observe that $\|\pi_\alpha(w^*) - w^*\|^2_{M,2} \leq M_{\max}(p-\tau)\alpha^2$ to get the result. □

Corollary B.2.1. *For τ-sparse vectors, we have $\alpha = 0$ and thus $(p-\tau)\alpha = 0$. Lemma B.2.5 can thus be simplified as*

$$-\frac{1}{2}\|\nabla f(w^t)\|^2_{M^{-1},\infty} \leq -\mu^{(t+\tau)}_{M,1}(f(w^t) - f(w^*)) \ . \tag{B.2.46}$$

When vectors are not sparse ($\tau = p$), we recover the inequality $-\frac{1}{2}\|\nabla f(w^t)\|^2_{M^{-1},\infty} \leq -\mu_{M,1}(f(w^t) - f(w^*))$.

B.2.4 (b) General Strongly-Convex Utility Result

Theorem B.2.2. *5.3.2 (Strongly-Convex Case) Let $\epsilon, \delta \in (0,1]$. Assume $\ell(\cdot; d)$ is a $\mu_{M,1}$-strongly-convex* w.r.t., *$\|\cdot\|_{M,1}$ and L-coordinate-Lipschitz loss function for all $d \in \mathcal{X}$, and f is M-coordinate-smooth. Let $\mathcal{W}^*$ be the set of minimizers of f, and f^* the minimum of f. Let $w_{priv} \in \mathbb{R}^p$ be the output of Algorithm 5.3.1 with step sizes $\gamma_j = 1/M_j$, and noise scales $\lambda_1, \ldots, \lambda_p, \lambda'_1, \ldots, \lambda'_p$ set as in Theorem 5.3.1 (with T chosen below) to ensure (ϵ, δ)-DP. Then, the following holds for $\zeta \in (0,1]$:*

$$f(w^T) - f(w^*) \leq (1 - \frac{\mu_{M,1}}{2})^T (f(w^0) - f(w^*)) + \frac{64TL^2_{\max}\log(1/\delta)}{M_{\min}\mu_{M,1}n^2\epsilon^2}\log(\frac{2Tp}{\zeta}) \ . \tag{B.2.47}$$

If we set $T = \frac{2}{\mu_{M,1}}\log(\frac{M_{\min}\mu_{M,1}n^2\epsilon^2(f(w^0)-f(w^*)}{32L^2_{\max}\log(1/\delta)})$, then with probability at least $1-\zeta$,

$$f(w^T) - f(w^*) = \widetilde{O}\Big(\frac{L^2_{\max}\log(p/\zeta)}{M_{\min}\mu^2_{M,1}n^2\epsilon^2}\Big) \ . \tag{B.2.48}$$

Proof. When f is $\mu_{M,1}$-strongly-convex *w.r.t.*, the norm $\|\cdot\|_{M,1}$, Corollary B.2.1 with $\tau = p$ and $\alpha = 0$ (which holds for any vector) yields

$$-\frac{1}{2}\|\nabla f(w^t)\|^2_{M^{-1},\infty} \leq -\mu_{M,1}(f(w^t) - f(w^*)) \ . \tag{B.2.49}$$

We replace this in Lemma B.2.2 to obtain

$$f(w^{t+1}) - f(w^*) \leq (1 - \frac{\mu_{M,1}}{4})(f(w^t) - f(w^*)) + \frac{1}{2M_j}|\eta_j^t|^2 + \frac{1}{2M_j}|\chi_j^t|^2 + \frac{1}{4M_{j^*}}|\chi_{j^*}^t|^2 \ . \tag{B.2.50}$$

Analogously to the proof of Theorem B.2.1, we define $\xi_t = f(w^t) - f(w^*)$ for all $0 \leq t \leq T$, and show that $\mathbb{P}\left(\exists t, \xi_{t+1} \geq (1 - \frac{\mu_{M,1}}{4})\xi_t + \beta\right) \leq \zeta/Tp$, with $\beta = \frac{2\lambda^2_{\max}}{M_{\min}} \log(\frac{8Tp}{\zeta})^2$. This yields that, with probability at least $1 - \zeta$,

$$f(w^T) - f(w^*) \leq (1 - \frac{\mu_{M,1}}{4})^T (f(w^0) - f(w^*)) + \sum_{t=0}^{T-1} (1 - \frac{\mu_{M,1}}{4})^{T-t}\beta \tag{B.2.51}$$

$$\leq (1 - \frac{\mu_{M,1}}{4})^T (f(w^0) - f(w^*)) + \frac{4}{\mu_{M,1}} \frac{32TL^2_{\max}\log(1/\delta)}{M_{\min}n^2\epsilon^2} \log\left(\frac{8Tp}{\zeta}\right)^2 \ , \tag{B.2.52}$$

With $T = \frac{4}{\mu_{M,1}} \log\left(\frac{\mu_{M,1}M_{\min}n^2\epsilon^2(f(w^0)-f(w^*))}{128L^2_{\max}\log(1/\delta)\log(8p/\zeta)}\right)$ we have, with probability at least $1-\zeta$,

$$f(w^T) - f(w^*) \leq \frac{128L^2_{\max}\log(1/\delta)\log(8p/\zeta)^2}{\mu_{M,1}M_{\min}n^2\epsilon^2} + \frac{512L^2_{\max}\log(1/\delta)\log(8Tp/\zeta)^2}{\mu^2_{M,1}M_{\min}n^2\epsilon^2} \log\left(\frac{\mu_{M,1}M_{\min}n^2\epsilon^2(f(w^0)-f(w^*))}{128L^2_{\max}\log(1/\delta)\log(8p/\zeta)^2}\right) \ , \tag{B.2.53}$$

which is the desired result. □

B.2.4 (c) Better Utility for Quasi-Sparse Solutions

Theorem B.2.3. *5.3.3 Consider f satisfying the hypotheses of Theorem 5.3.2, with Algorithm 5.3.1 initialized at $w^0 = 0$. We denote its output w^T, and assume that its iterates remain s-sparse for some $s \leq p$. Assume that, for all $\tau' \geq 0$, f is $\mu^{(\tau')}_{M,1}$-strongly-convex* w.r.t., *$\|\cdot\|_{M,1}$ for τ'-sparse vectors and $\mu_{M,2}$-strongly-convex* w.r.t., *$\|\cdot\|_{M,2}$, and that the (unique) solution of problem ($\star'$) is (α,τ)-quasi-sparse for some $\alpha, \tau \geq 0$. Let $T \geq 0$, $\zeta \in [0,1]$, and $\beta = \frac{2\lambda^2_{\max}}{M_{\min}} \log(TP/\zeta)^2$. Then for all $t \leq T$ we have that, with probability at least $1 - \zeta$:*

$$f(w^T) - f(w^*)$$

$$\leq \left(1 - \frac{\mu^{(\min(s,T)+\tau)}_{M,1}}{4}\right)^T (f(w^0) - f(w^*)) + \frac{4(\min(s,T)+\tau)\beta}{\mu_{M,2}} + \frac{\min(s,T)+\tau}{8}(p-\tau)\alpha^2 \tag{B.2.54}$$

$$\leq \left(1 - \frac{\mu_{M,2}}{4(\min(s,T)+\tau)}\right)^T (f(w^0) - f(w^*)) + \frac{4(\min(s,T)+\tau)\beta}{\mu_{M,2}} + \frac{\min(s,T)+\tau}{8}(p-\tau)\alpha^2. \tag{B.2.55}$$

Setting $T = \frac{s+\tau}{\mu_{M,2}} \log(\frac{(f(w^0)-f^*)M_{\min}\mu_{M,2}n^2\epsilon^2}{L^2})$, *and assuming* $\alpha^2 = O\left(\frac{L_{\max}^2(s+\tau)}{M_{\min}\mu_{M,2}^2 pn^2\epsilon^2}\right)$, *we obtain that with probability at least* $1-\zeta$,

$$f(w^T) - f^* = \widetilde{O}\left(\frac{L_{\max}^2}{M_{\min}} \frac{(s+\tau)^2 \log(2p/\zeta)}{\mu_{M,2} n^2 \epsilon^2}\right) \quad . \tag{B.2.56}$$

Proof. First, we remark that at each iteration, we change only one coordinate. Therefore, after t iterations, the iterate w^t is at most t-sparse. Since all iterates are also s-sparse, it is $\min(s,t)$-sparse. Additionally, we assumed that w^* is (τ,α)-almost-sparse. Therefore, Lemma B.2.5 yields

$$-\frac{1}{2}\|\nabla f(w^t)\|_{M^{-1},\infty} \leq -\mu_{M,1}^{(\min(s,t)+\tau)}(f(w^t) - f(w^*)) + \frac{\mu_{M,1}^{(\min(s,t)+\tau)}}{2}(p-\tau)\alpha^2 \quad , \tag{B.2.57}$$

and Lemma B.2.2 becomes

$$\begin{aligned} f(w^{t+1}) - f(w^*) \leq (1 - \frac{\mu_{M,1}^{(\min(s,t)+\tau)}}{4})(f(w^t) - f(w^*)) + \frac{\mu_{M,1}^{(\min(s,t)+\tau)}}{8}(p-\tau)\alpha^2 \\ + \frac{1}{2M_j}|\eta_j^t|^2 + \frac{1}{2M_j}|\chi_j^t|^2 + \frac{1}{4M_{j^*}}|\chi_{j^*}^t|^2 \quad . \end{aligned} \tag{B.2.58}$$

Then by Chernoff's equality, we obtain (similarly to the proof of Theorem 5.3.2 for the convex case) that with probability at least $1-\zeta$, for $T \geq 0$,

$$\begin{aligned} f(w^T) - f(w^*) \leq \prod_{t=0}^{T}\left(1 - \tfrac{\mu_{M,1}^{(\min(s,t)+\tau)}}{4}\right)(f(w^0) - f(w^*)) \\ + \sum_{t=0}^{T-1}\prod_{k=T-t}^{T}\left(1 - \tfrac{\mu_{M,1}^{(\min(s,k)+\tau)}}{4}\right)\left(\beta + \tfrac{\mu_{M,1}^{(\min(s,t)+\tau)}}{8}(p-\tau)\alpha^2\right) \quad . \end{aligned} \tag{B.2.59}$$

Since for $k \in [T]$, $\mu_{M,1}^{\min(s,k)+\tau} \geq \mu_{M,1}^{\min(s,T)+\tau}$, we can further upper bound $\mu_{M,1}^{(\min(s,t)+\tau)} \leq \mu_{M,1}^{(\tau)}$, and $1 - \frac{\mu_{M,1}^{(\min(s,t)+\tau)}}{4} \leq 1 - \frac{\mu_{M,1}^{(\min(s,T)+\tau)}}{4}$ and

$$\sum_{t=0}^{T-1}\prod_{k=T-t}^{T}\left(1 - \frac{\mu_{M,1}^{(\min(s,k)+\tau)}}{4}\right) \leq \sum_{t=0}^{T-1}\left(1 - \frac{\mu_{M,1}^{(\min(s,T)+\tau)}}{4}\right)^t \leq \frac{4}{\mu_{M,1}^{(\min(s,T)+\tau)}} \quad , \tag{B.2.60}$$

which allows to simplify the above expression to

$$
\begin{aligned}
& f(w^T) - f(w^*) \\
& \leq \left(1 - \frac{\mu_{M,1}^{(\min(s,T)+\tau)}}{4}\right)^T (f(w^0) - f(w^*)) + \frac{4}{\mu_{M,1}^{(\min(s,T)+\tau)}}\left(\beta + \frac{\mu_{M,1}^{(\tau)}}{8}(p-\tau)\alpha^2\right) && \text{(B.2.61)} \\
& \leq \left(1 - \frac{\mu_{M,2}}{4(\min(s,T)+\tau)}\right)^T (f(w^0) - f(w^*)) + \frac{4(\min(s,T)+\tau)}{\mu_{M,2}}\left(\beta + \frac{\mu_{M,2}}{8}(p-\tau)\alpha^2\right) && \text{(B.2.62)} \\
& \leq \left(1 - \frac{\mu_{M,2}}{4(\min(s,T)+\tau)}\right)^T (f(w^0) - f(w^*)) + \frac{4(\min(s,T)+\tau)\beta}{\mu_{M,2}} + \frac{\min(s,T)+\tau}{8}(p-\tau)\alpha^2 \ , && \text{(B.2.63)}
\end{aligned}
$$

where the second inequality follows from $\mu_{M,1}^{(\min(s,T)+\tau)} \geq \frac{\mu_{M,2}^{(\min(s,T)+\tau)}}{\min(s,T)+\tau} \geq \frac{\mu_{M,2}}{\min(s,T)+\tau}$ and $\mu_{M,1}^{(\tau)} \leq \mu_{M,2}$. We have proven inequalities (B.2.54) and (B.2.55) of the theorem.

When $\alpha^2 = O\left(L_{\max}^2(s+\tau)/M_{\min}\mu_{M,2}^2 p n^2 \epsilon^2\right)$, the two additive terms of (B.2.63) are $O((s+\tau)\beta/\mu_{M,2})$. Since $\min(s,T)+\tau \leq s+\tau$, we choose $T = \frac{s+\tau}{\mu_{M,2}}\log((f(w^0) - f^*)M_{\min}\mu_{M,2}n^2\epsilon^2/L^2)$ to balance all the terms and obtain the result. □

Appendix C

Proofs of Chapter 6

C.1 Fairness functions

In this section we recall several well known fairness functions and show that they can be written in the form of Equation (6.3.1).

Example C.1.1 (**Equalized Odds (Hardt et al., 2016)**). *A model h is fair for Equalized Odds when the probability of predicting the correct label is independent of the sensitive attribute, that is, $\forall (y, r) \in \mathcal{Y} \times \mathcal{S}$*

$$F_{(y,r)}(h, D) = \mathbb{P}\left(H(X) = Y \mid Y = y, S = r\right) - \mathbb{P}\left(H(X) = Y \mid Y = y\right) \ .$$

We can then write $F_{(y,r)}(h, D)$ in the form of (6.3.1) as

$$F_{(y,r)}(h, D) = C^0_{(y,r)} + \sum_{(y',r') \in \mathcal{Y} \times \mathcal{S}} C^{(y',r')}_{(y,r)} \, \mathbb{P}\left(H(x) = Y | Y = y', S = r'\right) \ , \qquad \text{(C.1.1)}$$

with

$$\begin{aligned} C^0_{(y,r)} &= 0 \ , \\ C^{(y,r)}_{(y,r)} &= 1 - \mathbb{P}\left(S = r \mid Y = y\right) \ , \\ \forall r' \neq r, \ C^{(y,r')}_{(y,r)} &= \ -\mathbb{P}\left(S = r' \mid Y = y\right) \ , \\ \forall y' \neq y, \ \forall r' \in \mathcal{S}, \ C^{(y',r')}_{(y,r)} &= 0 \ . \end{aligned}$$

Proof. We have that

$$\begin{aligned} F_{(y,r)}(h, D) &= \mathbb{P}\left(H(X) = Y \mid Y = y, S = r\right) - \mathbb{P}\left(H(X) = Y \mid Y = y\right) \\ &= \mathbb{P}\left(H(X) = Y \mid Y = y, S = r\right) \\ &\quad - \sum_{r' \in \mathcal{S}} \mathbb{P}\left(H(X) = Y \mid Y = y, S = r'\right) \mathbb{P}\left(S = r' \mid Y = y\right) \ , \end{aligned}$$

which gives the result. □

Example C.1.2 (**Equality of Opportunity Hardt et al., 2016**). *A model h is fair for Equality of Opportunity when the probability of predicting the correct label is independent of the sensitive attribute for the set of desirable outcomes $\mathcal{Y}' \subset \mathcal{Y}$, that is $\forall (y, r) \in \mathcal{Y} \times \mathcal{S}$*

$$F_{(y,r)}(h, D) = \begin{cases} \mathbb{P}\left(H(X) = Y \mid Y = y, S = r\right) - \mathbb{P}\left(H(X) = Y \mid Y = y\right) & \text{if } y \in \mathcal{Y}' \ , \\ 0 & \text{otherwise} \ . \end{cases}$$

We can then write $F_{(y,r)}(h, D)$ in the form of (6.3.1) as

$$F_{(y,r)}(h, D) = C^0_{(y,r)} + \sum_{(y',r') \in \mathcal{Y} \times \mathcal{S}} C^{(y',r')}_{(y,r)} \, \mathbb{P}\left(H(X) = Y | Y = y', S = r'\right) \tag{C.1.2}$$

with, if $y \in \mathcal{Y}'$,

$$\begin{aligned} C^0_{(y,r)} &= 0 \ , \\ C^{(y,r)}_{(y,r)} &= 1 - \mathbb{P}\left(S = r \mid Y = y\right) \ , \\ \forall r' \neq r, \ C^{(y,r')}_{(y,r)} &= -\mathbb{P}\left(S = r' \mid Y = y\right) \ , \\ \forall y' \neq y, \ \forall r' \in \mathcal{S}, \ C^{(y',r')}_{(y,r)} &= 0 \ . \end{aligned}$$

and, if $y \in \mathcal{Y} \setminus \mathcal{Y}'$,

$$\forall y' \in \mathcal{Y}, \ \forall r' \in \mathcal{S}, \ C^{(y',r')}_{(y,r)} = 0 \ .$$

Proof. We consider the two cases. On the one hand, when $y \in \mathcal{Y} \setminus \mathcal{Y}'$, we have that

$$F_{(y,r)}(h, D) = 0 \ ,$$

which gives the first part of the result. On the other hand, when $y \in \mathcal{Y}'$, then

$$\begin{aligned} F_{(y,r)}(h, D) &= \mathbb{P}\left(H(X) = Y \mid Y = y, S = r\right) - \mathbb{P}\left(H(X) = Y \mid Y = y\right) \\ &= \mathbb{P}\left(H(X) = Y \mid Y = y, S = r\right) \\ &\quad - \sum_{r' \in \mathcal{S}} \mathbb{P}\left(H(X) = Y \mid Y = y, S = r'\right) \mathbb{P}\left(S = r' \mid Y = y\right) \ , \end{aligned}$$

which gives the second part of the result. □

Example C.1.3 (**Accuracy Parity Zafar et al., 2017**). *A model h is fair for Accuracy Parity when the probability of being correct is independent of the sensitive attribute, that is, $\forall (r) \in \mathcal{S}$*

$$F_{(r)}(h, D) = \mathbb{P}\left(H(X) = Y \mid S = r\right) - \mathbb{P}\left(H(X) = Y\right).$$

We can then write $F_{(r)}(h, D)$ *in the form of* (6.3.1) *as*

$$F_{(r)}(h, D) = C^0_{(r)} + \sum_{(r')\in\mathcal{S}} C^{(r')}_{(r)} \mathbb{P}\left(H(X) = Y | S = r'\right) \tag{C.1.3}$$

with

$$\begin{aligned} C^0_{(r)} &= 0 \ , \\ C^{(r)}_{(r)} &= 1 - \mathbb{P}\left(S = r\right) \ , \\ \forall r' \neq r,\ C^{(r')}_{(r)} &= -\mathbb{P}\left(S = r'\right) \ . \end{aligned}$$

Proof. We have that

$$\begin{aligned} F_{(r)}(h, D) &= \mathbb{P}\left(H(X) = Y \mid S = r\right) - \mathbb{P}\left(H(X) = Y\right) \\ &= \mathbb{P}\left(H(X) = Y \mid S = r\right) - \sum_{r'\in\mathcal{S}} \mathbb{P}\left(H(X) = Y \mid S = r'\right) \mathbb{P}\left(S = r'\right) \ , \end{aligned}$$

which gives the result. □

Example C.1.4 (**Demographic Parity (Binary Labels) Calders et al., 2009**). *A model* h *is fair for Demographic Parity with binary labels when the probability of predicting a label is independent of the sensitive attribute, that is,* $\forall (y, r) \in \mathcal{Y} \times \mathcal{S}$

$$F_{(y,r)}(h, D) = \mathbb{P}\left(H(X) = y \mid S = r\right) - \mathbb{P}\left(H(X) = y\right).$$

Assuming that given a label y*, the second binary label is denoted* $\bar{y}$*, we can then write* $F_{(y,r)}(h, D)$ *in the form of* (6.3.1) *as*

$$F_{(y,r)}(h, D) = C^0_{(y,r)} + \sum_{(y',r')\in\mathcal{Y}\times\mathcal{S}} C^{(y',r')}_{(y,r)} \mathbb{P}\left(H(X) = Y | Y = y', S = r'\right) \ , \tag{C.1.4}$$

with

$$\begin{aligned} C^0_{(y,r)} &= \mathbb{P}\left(Y = y\right) - \mathbb{P}\left(Y = y \mid S = r\right) \ , \\ C^{(y,r)}_{(y,r)} &= \mathbb{P}\left(Y = y \mid S = r\right) - \mathbb{P}\left(Y = y, S = r\right) \ , \\ C^{(\bar{y},r)}_{(y,r)} &= \mathbb{P}\left(Y = \bar{y}, S = r\right) - \mathbb{P}\left(Y = \bar{y} \mid S = r\right) \ , \\ \forall r' \neq r,\ C^{(y,r')}_{(y,r)} &= -\mathbb{P}\left(Y = y, S = r'\right) \ , \\ \forall r' \neq r,\ C^{(\bar{y},r')}_{(y,r)} &= \mathbb{P}\left(Y = \bar{y}, S = r'\right) \ . \end{aligned}$$

Proof. We have that

$$
\begin{aligned}
F_{(y,r)}(h,D) &= \mathbb{P}\left(H(X)=y \mid S=r\right) - \mathbb{P}\left(H(X)=y\right) \\
&= \mathbb{P}\left(H(X)=y \mid Y=y, S=r\right)\mathbb{P}\left(Y=y \mid S=r\right) \\
&\quad + \mathbb{P}\left(H(X)=y \mid Y\neq y, S=r\right)\mathbb{P}\left(Y\neq y \mid S=r\right) \\
&\quad - \sum_{r'\in\mathcal{S}} \big(\mathbb{P}\left(H(X)=y \mid Y=y, S=r'\right)\mathbb{P}\left(Y=y, S=r'\right) \\
&\qquad\quad + \mathbb{P}\left(H(X)=y \mid Y\neq y, S=r'\right)\mathbb{P}\left(Y\neq y, S=r'\right)\big) \\
&= \mathbb{P}\left(H(X)=y \mid Y=y, S=r\right)\mathbb{P}\left(Y=y \mid S=r\right) \\
&\quad + 1 - \mathbb{P}\left(H(X)\neq y \mid Y\neq y, S=r\right)\mathbb{P}\left(Y\neq y \mid S=r\right) \\
&\quad - \sum_{r'\in\mathcal{S}} \big(\mathbb{P}\left(H(X)=y \mid Y=y, S=r'\right)\mathbb{P}\left(Y=y, S=r'\right) \\
&\qquad\quad + 1 - \mathbb{P}\left(H(X)\neq y \mid Y\neq y, S=r'\right)\mathbb{P}\left(Y\neq y, S=r'\right)\big) \quad .
\end{aligned}
$$

Here, we only consider binary labels, y and $\bar{y}$. Hence, $H(X)\neq y \Leftrightarrow H(X)=\bar{y}$ and $Y\neq y \Leftrightarrow Y=\bar{y}$. Thus, we obtain

$$
\begin{aligned}
&F_{(y,r)}(h,D) \\
&= \mathbb{P}\left(H(X)=y \mid Y=y, S=r\right)\mathbb{P}\left(Y=y \mid S=r\right) \\
&\quad + \left(1-\mathbb{P}\left(H(X)=\bar{y} \mid Y=\bar{y}, S=r\right)\right)\mathbb{P}\left(Y=\bar{y} \mid S=r\right) \\
&\quad - \sum_{r'\in\mathcal{S}} \big(\mathbb{P}\left(H(X)=y \mid Y=y, S=r'\right)\mathbb{P}\left(Y=y, S=r'\right) \\
&\qquad\quad + \left(1-\mathbb{P}\left(H(X)=\bar{y} \mid Y=\bar{y}, S=r'\right)\right)\mathbb{P}\left(Y=\bar{y}, S=r'\right)\big) \\
&= \mathbb{P}\left(H(X)=y \mid Y=y, S=r\right)\left[\mathbb{P}\left(Y=y \mid S=r\right) - \mathbb{P}\left(Y=y, S=r\right)\right] \\
&\quad + \mathbb{P}\left(H(X)=\bar{y} \mid Y=\bar{y}, S=r\right)\left[\mathbb{P}\left(Y=\bar{y}, S=r\right) - \mathbb{P}\left(Y=\bar{y} \mid S=r\right)\right] \\
&\quad + \sum_{r'\in\mathcal{S}, r'\neq r} \mathbb{P}\left(H(X)=y \mid Y=y, S=r'\right)\left(-\mathbb{P}\left(Y=y, S=r'\right)\right) \\
&\quad + \sum_{r'\in\mathcal{S}, r'\neq r} \mathbb{P}\left(H(X)=\bar{y} \mid Y=\bar{y}, S=r'\right)\mathbb{P}\left(Y=\bar{y}, S=r'\right) \\
&\quad + \mathbb{P}\left(Y=\bar{y} \mid S=r\right) - \mathbb{P}\left(Y=\bar{y}\right) \\
&= \mathbb{P}\left(H(X)=Y \mid Y=y, S=r\right)\left[\mathbb{P}\left(Y=y \mid S=r\right) - \mathbb{P}\left(Y=y, S=r\right)\right] \\
&\quad + \mathbb{P}\left(H(X)=Y \mid Y=\bar{y}, S=r\right)\left[\mathbb{P}\left(Y=\bar{y}, S=r\right) - \mathbb{P}\left(Y=\bar{y} \mid S=r\right)\right] \\
&\quad + \sum_{r'\in\mathcal{S}, r'\neq r} \mathbb{P}\left(H(X)=Y \mid Y=y, S=r'\right)\left(-\mathbb{P}\left(Y=y, S=r'\right)\right) \\
&\quad + \sum_{r'\in\mathcal{S}, r'\neq r} \mathbb{P}\left(H(X)=Y \mid Y=\bar{y}, S=r'\right)\mathbb{P}\left(Y=\bar{y}, S=r'\right) \\
&\quad + \mathbb{P}\left(Y=y\right) - \mathbb{P}\left(Y=y \mid S=r\right) \quad ,
\end{aligned}
$$

which gives the result. □

C.2 Proof of Theorem 6.4.1

Theorem C.2.1 (Pointwise Lipschitzness of Conditional Negative Predictions)**.** *Let $\mathcal{H}$ be a set of real vector-valued functions with $L_{X,Y}$ the Lipschitz constants defined in Assumption 6.3.1. Let $h, h' \in \mathcal{H}$ be two models, (X, Y, S) be a triple of random variables having distribution $\mathcal{D}$, and E be an arbitrary event. Assume that $\mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h,X,Y)|} \mid E\right) < +\infty$, then*

$$|\,\mathbb{P}(H(X) = Y \mid E) - \mathbb{P}(H'(X) = Y \mid E)| \leq \mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h, X, Y)|} \mid E\right) \|h - h'\|_{\mathcal{H}} \ .$$

Proof. The proof of this theorem is in two steps. First, we use the Lipschitz continuity property associated with $\mathcal{H}$, the triangle inequality, and the union bound to show that $|\,\mathbb{P}\,(H(X) = Y \mid E) - \mathbb{P}\,(H'(X) = Y \mid E)\,| \leq \mathbb{P}\left(\frac{L_{X,Y}}{|\rho(h,X,Y)|} \leq \|h - h'\|_{\mathcal{H}} \mid E\right)$. Then, applying Markov's inequality gives the desired result.

Bounding $|\,\mathbb{P}\,(H(X) = Y \mid E) - \mathbb{P}\,(H'(X) = Y \mid E)\,|$. We have that Similarly, we have that It implies that

$$|\,\mathbb{P}\,(H(X) = Y \mid E) - \mathbb{P}\,(H'(X) = Y \mid E)\,| \leq \mathbb{P}\left(\frac{|\rho(h, X, Y)|}{L_{X,Y}} \leq \|h - h'\|_{\mathcal{H}} \mid E\right)$$

Bounding $\mathbb{P}\left(\frac{|\rho(h,X,Y)|}{L_{X,Y}} \leq \|h - h'\|_{\mathcal{H}} \mid E\right)$. We use the Markov's Inequality and we assume that $\mathbb{E}\left(\frac{L_{X,Y}}{|\rho(h,X,Y)|} \mid E\right) < +\infty$. Hence, we have that It concludes the proof. □

Remark C.2.1. *In the last step of the proof of Theorem 6.4.1, we can also use the Chernoff bound:*

$$\begin{aligned}
&\mathbb{P}\left(\frac{|\rho(h, X, Y)|}{L_{X,Y}} \leq \|h - h'\|_{\mathcal{H}} \mid E\right) \\
&= \mathbb{P}\left(\exp\left(-t\frac{|\rho(h, X, Y)|}{L_{X,Y}}\right) \geq \exp\left(-t\|h - h'\|_{\mathcal{H}}\right) \mid E\right) \\
&\leq \mathbb{E}\left(\exp\left(-t\frac{|\rho(h, X, Y)|}{L_{X,Y}}\right) \mid E\right) \exp\left(t\|h - h'\|_{\mathcal{H}}\right) \ .
\end{aligned}$$

A correct choice of t would lead to potentially tighter bounds than the Markov's inequality at the expense of readability.

Remark C.2.2. *Before using Markov's inequality or Chernoff bound in Theorem 6.4.1, we can modify the probability as*

$$\mathbb{P}\left(\frac{|\rho(h,X,Y)|}{L_{X,Y}} \leq \|h-h'\|_{\mathcal{H}} \mid E\right) = \mathbb{P}\left(\left[\frac{|\rho(h,X,Y)|}{L_{X,Y}}\right]^{\|h-h'\|_{\mathcal{H}}} \leq \|h-h'\|_{\mathcal{H}} \mid E\right) \ ,$$

where

$$\left[\frac{|\rho(h,X,Y)|}{L_{X,Y}}\right]^{\|h-h'\|_{\mathcal{H}}} = \begin{cases} \frac{|\rho(h,X,Y)|}{L_{X,Y}} & \text{if } \ |\rho(h,X,Y)| \leq L_{X,Y}\|h-h'\|_{\mathcal{H}} \ , \\ +\infty & \text{otherwise} \ . \end{cases}$$

This essentially means that, whenever the model's margin on a data record is large enough, its precise value is no more meaningful, as its prediction will not change whatsoever. The remaining of Theorem 6.4.1's proof is unchanged, except that we have $\left[\frac{|\rho(h,X,Y)|}{L_{X,Y}}\right]^{\|h-h'\|_{\mathcal{H}}}$ *instead of* $\frac{|\rho(h,X,Y)|}{L_{X,Y}}$.

Note that this can lead to much tighter bounds. Notably, when distance $\|h-h'\|_{\mathcal{H}}$ *between* h *and* h' *is small enough, the difference of fairness may even become zero.*

C.3 Proof of Theorem 6.4.2

Theorem C.3.1 (Pointwise Lipschitzness of Fairness)**.** *Let* $h, h' \in \mathcal{H}$, $L_{X,Y}$ *be defined as in Assumption 6.3.1, and* $(X,S,Y) \sim \mathcal{D}$. *For any fairness notion of the form of* (6.3.1)*, we have:*

$$\forall k \in [K], |F_k(h,D) - F_k(h',D)| \leq \chi_k(h,D)\|h-h'\|_{\mathcal{H}} \ ,$$

with $\chi_k(h,D) = \sum_{k'=1}^{K} |C_k^{k'}|\, \mathbb{E}\left(\frac{1}{|h(X)|} \mid D_{k'}\right)$. *Similarly, for the aggregate measure of fairness defined in* (6.3.2)*, we have:*

$$|Fair(h,D) - Fair(h',D)| \leq \frac{1}{K}\sum_{k=1}^{K} \chi_k(h,D)\|h-h'\|_{\mathcal{H}} \ .$$

Proof. The first part follows from the following derivation. For all k,

The second part is obtained thanks to the triangle inequality: which gives the claim when combined with the first part of the theorem. □

C.4 Bound for Output Perturbation (Proof of Lemma 6.5.1)

Lemma C.4.1. *Let h^{priv} be the vector released by output perturbation with noise $\sigma^2 = {8\Lambda^2 \log(1.25/\delta)}/{\mu^2 n^2 \epsilon^2}$, and $0 < \zeta < 1$, then with probability at least $1-\zeta$,*

$$\|h^{priv} - h^*\|_2^2 \leq \frac{32 p \Lambda^2 \log(1.25/\delta)\log(2/\zeta)}{\mu^2 n^2 \epsilon^2} \ .$$

Proof. We prove this lemma in two steps. First, we show that for a given sensitivity, the distance $\|h^{\text{priv}} - h^*\|$ is bounded. Second, we estimate the sensitivity.

Bounding the Error. Let Δ be the sensitivity of the function $D \to \arg\min_{w\in\mathcal{C}} f(w; D)$. Its value can be released under (ϵ, δ) differential privacy (Chaudhuri et al., 2011; Lowy and Razaviyayn, 2021) as follows:

$$h^{\text{priv}} = h^* + \mathcal{N}(0, \sigma^2 \mathbb{I}_p) \ , \tag{C.4.1}$$

where $\sigma^2 = \frac{2\Delta^2 \log(1.25/\delta)}{\epsilon^2}$ and $h^* = \arg\min_{h\in\mathcal{C}} f(h)$. Then, Chernoff's bound gives, for $t, \alpha > 0$,

$$\mathbb{P}(\|h^{priv} - h^*\|^2 \geq \alpha) \leq \exp(-t\alpha)\, \mathbb{E}(\exp(t\|h^{priv} - h^*\|^2)) \tag{C.4.2}$$

$$= \exp(-t\alpha) \prod_{j=1}^{p} \mathbb{E}(\exp(t(h_j^{priv} - h_j^*)^2)) \ , \tag{C.4.3}$$

by independence of the noise's p coordinates. Since $h_j^{priv} - h_j^*$ is a Gaussian random variable of mean 0 and variance σ^2, we can compute $\mathbb{E}(\exp(t(h_j^{priv} - h_j^*))) = (1 - 2t\sigma^2)^{-1/2}$. We then obtain

$$\mathbb{P}(\|h^{priv} - h^*\|^2 \geq \alpha) \leq \exp(-t\alpha)(1 - 2t\sigma^2)^{-p/2} \ . \tag{C.4.4}$$

Let $t = 1/4p\sigma^2$, then it holds that $1 - 2t\sigma^2 = 1 - 1/2p \leq 1$ and

$$(1 - 2t\sigma^2)^{-p/2} = \exp\left(-\frac{p}{2}\log(1 - \frac{1}{2p})\right) \leq \exp\left(\frac{1}{2(1-\frac{1}{p})}\right) \leq \exp(1/2) \leq 2 \ , \tag{C.4.5}$$

since $\frac{p}{2}\log(1-\frac{1}{2p}) \geq \frac{p}{2}\frac{-1/2p}{1-1/2p} \geq -\frac{1}{2}$. Let $0 < \zeta < 1$, $t = 1/4p\sigma^2$ and $\alpha = 4p\sigma^2 \log(2/\zeta)$, we have proven

$$\mathbb{P}(\|h^{priv} - h^*\|^2 \geq \alpha) \leq 2\exp\left(-\frac{\alpha}{4p\sigma}\right) \leq \zeta \ . \tag{C.4.6}$$

The error obtained by output perturbation is thus upper bounded by $\|h^{priv} - h^*\|^2 \leq 4p\sigma^2 \log(2/\zeta) = \frac{8p\Delta^2 \log(1.25/\delta)\log(2/\zeta)}{\epsilon^2}$ with probability at least $1-\zeta$.

Estimating the Sensitivity. Define $g(h) = \frac{1}{n}\sum_{i=1}^n \ell(w; d_i')$ with $d_i' \in \mathcal{X} \times \mathcal{Y}$ such that $d_i' = d_i$ for all $i \neq 1$. By strong convexity, the two following inequalities hold for h, h',

$$f(h) \geq f(h') + \langle \nabla f(h'), h - h' \rangle + \frac{\mu}{2}\|h - h'\|^2 \ , \tag{C.4.7}$$

$$f(h') \geq f(h) + \langle \nabla f(h), h' - h \rangle + \frac{\mu}{2}\|h - h'\|^2 \ . \tag{C.4.8}$$

Summing these two inequalities give $\langle \nabla f(h) - \nabla f(h'), h - h' \rangle \geq \frac{\mu}{2}\|h - h'\|^2$. Let h_1^* and h_2^* be the respective minimizers of f and g over $\mathcal{C}$, taking $h = h_1^*$ and $h' = h_2^*$ gives

$$\frac{\mu}{2}\|h_1^* - h_2^*\|^2 \leq \langle \nabla f(h_1^*) - \nabla f(h_2^*), h_1^* - h_2^* \rangle \leq \|\nabla f(h_1^*) - \nabla f(h_2^*)\| \cdot \|h_1^* - h_2^*\| \ . \tag{C.4.9}$$

Now, if $\mathcal{C} = \mathbb{R}^p$, optimality conditions give

$$\nabla f(h_1^*) = 0 = \nabla g(h_2^*) = \nabla f(h_2^*) - \nabla F(h_2^*; d_1) + F(h_2^*; d_1') \ , \tag{C.4.10}$$

resulting in $\|\nabla f(h_1^*) - \nabla f(h_2^*)\| = \|\frac{1}{n}\nabla(h_2^*; d_1) - \frac{1}{n}\nabla(h_2^*; d_1')\| \leq \frac{2\Lambda}{n}$. Combined with (C.4.9), this shows that the sensitivity of $\arg\min_{h\in\mathcal{C}} f(h)$ is $\Delta = \frac{2\Lambda}{n\mu}$, which concludes the proof. □

C.5 Convergence of DP-SGD (Proof of Lemma 6.5.2)

Lemma C.5.1. *Let h^{priv} be the vector released by DP-SGD with noise scale $\sigma^2 = \frac{64\Lambda^2 T^2 \log(3T/\delta)\log(2/\delta)}{n^2\epsilon^2}$. Assume that $\sigma_*^2 = \mathbb{E}_{i\sim[n]}\|\nabla\ell(h^*; x_i, y_i)\|^2 \leq \sigma^2$. Let $0 < \zeta < 1$, then with probability at least $1-\zeta$,*

$$\|h^{priv} - h^*\|_2^2 = \widetilde{O}\left(\frac{p\Lambda^2 \log(1/\delta)^2}{\zeta\mu^2 n^2 \epsilon^2}\right) \ ,$$

where $\widetilde{O}$ ignores logarithmic terms in n (the number of examples) and p (the number of model parameters).

Proof. We start by recalling that in DP-SGD,

$$h^{t+1} = \pi_{\mathcal{H}}(h^t - \gamma(g^t + \eta^t)) \ . \tag{C.5.1}$$

Since $h^* \in \mathcal{H}$, and $\mathcal{H}$ is convex, we have

$$\begin{aligned}\|h^{t+1} - h^*\|^2 &= \|\pi_{\mathcal{H}}(h^t - \gamma(g^t + \eta^t)) - h^*\|^2 && \text{(C.5.2)}\\ &= \|h^t - h^*\|^2 - 2\gamma\langle g^t + \eta^t, h^t - h^*\rangle + \gamma^2\|g^t + \eta^t\|^2 && \text{(C.5.3)}\\ &\leq \|h^t - h^*\|^2 - 2\gamma\langle g^t + \eta^t, h^t - h^*\rangle + 2\gamma^2\|g^t\|^2 + 2\gamma^2\|\eta^t\|^2 \ , && \text{(C.5.4)}\end{aligned}$$

where we developed the square and used $\|a+b\|^2 \leq 2\|a\|^2+2\|b\|^2$ for $a, b \in \mathbb{R}^p$. Taking the expectation with respect to the stochastic gradient computation and noise, we obtain

$$\mathbb{E}\,\|h^{t+1} - h^*\|^2 \leq \|h^t - h^*\|^2 - 2\gamma\langle \nabla f(h^t), h^t - h^*\rangle + 2\gamma^2\,\mathbb{E}\,\|g^t\|^2 + 2\gamma^2\,\mathbb{E}\,\|\eta^t\|^2 \ , \tag{C.5.5}$$

since $\mathbb{E}(\eta^t) = 0$ and $\mathbb{E}(g^t) = \nabla f(h^t)$. Now recall that, by strong-convexity of f, we have

$$f(h^*) \geq f(h^t) + \langle \nabla f(h^t), h^* - h^t\rangle + \frac{\mu}{2}\|h^t - h^*\|^2 \ . \tag{C.5.6}$$

By reorganizing, we obtain $-2\gamma\langle \nabla f(h^t), h^t - h^*\rangle \leq -2\gamma(f(h^t) - f(h^*)) - \gamma\mu\|h^t - h^*\|^2$, which gives

$$\mathbb{E}\,\|h^{t+1} - h^*\|^2 \leq (1 - \gamma\mu)\|h^t - h^*\|^2 - 2\gamma(f(h^t) - f(h^*)) + 2\gamma^2\,\mathbb{E}\,\|g^t\|^2 + 2\gamma^2\,\mathbb{E}\,\|\eta^t\|^2 \ . \tag{C.5.7}$$

Finally, remark that if $f = \frac{1}{n}\sum_{i=1}^n f_i$ with each f_i being β-smooth and $\mathbb{E}\, f_i = f$, we have, for $i \sim [n]$,

$$\begin{aligned}\mathbb{E}\,\|\nabla f_i(h^t)\|^2 &= \mathbb{E}\,\|\nabla f_i(h^t) - \nabla f_i(h^*) + \nabla f_i(h^*)\|^2 && \text{(C.5.8)}\\ &\leq \mathbb{E}(2\|\nabla f_i(h^t) - \nabla f_i(h^*)\|^2 + 2\|\nabla f_i(h^*)\|^2) && \text{(C.5.9)}\\ &\leq \mathbb{E}(4\beta(f_i(h^t) - f_i(h^*) - \langle \nabla f_i(h^*), h^t - h^*\rangle) + 2\|\nabla f_i(h^*)\|^2) && \text{(C.5.10)}\\ &= 4\beta(f(h^t) - f(h^*)) + 2\,\mathbb{E}\,\|\nabla f_i(h^*)\|^2 \ , && \text{(C.5.11)}\end{aligned}$$

since f_i is β-smooth, which implies, for all $w, v \in \mathbb{R}^p$,

$$\|\nabla f_i(w) - \nabla f_i(v)\|^2 \leq 2\beta(f_i(w) - f_i(v) - \langle \nabla f_i(v), w - v\rangle \ , \tag{C.5.12}$$

and $\mathbb{E}\,\nabla f_i(h^*) = 0$. Combined with the fact that $\mathbb{E}\,\|\nabla f_i(h^*)\|^2 \leq \sigma_*^2$ and $\mathbb{E}\,\|\eta^t\|^2 = p\sigma^2$, we obtained

$$\begin{aligned}\mathbb{E}\,\|h^{t+1} - h^*\|^2 &\leq (1 - \gamma\mu)\|h^t - h^*\|^2 + (4\beta\gamma^2 - 2\gamma)(f(h^t) - f(h^*)) + 2\gamma^2(\sigma_*^2 + \sigma^2) && \text{(C.5.13)}\\ &\leq (1 - \gamma\mu)\|h^t - h^*\|^2 + 4\gamma^2\sigma^2 \ , && \text{(C.5.14)}\end{aligned}$$

since $\gamma \leq 1/2\beta$, which implies $4\beta\gamma^2 - 2\gamma \leq 0$ and $\sigma^* \leq \sigma$. By induction, we obtain that, after T iterations,

$$\mathbb{E}\,\|h^T - h^*\|^2 \leq (1-\gamma\mu)^T\|h^0 - h^*\|^2 + 4\gamma^2\sum_{t=0}^{T-1}(1-\gamma\mu)^{T-t}\sigma^2 \tag{C.5.15}$$

$$\leq (1-\gamma\mu)^T\|h^0 - h^*\|^2 + \frac{4\gamma\sigma^2}{\mu}\ . \tag{C.5.16}$$

Now, recall that DP-SGD is (ϵ,δ)-differentially private for $\sigma^2 = \frac{64\Lambda^2 T\log(3T/\delta)\log(2/\delta)}{n^2\epsilon^2}$ (following from the Gaussian mechanism, advanced composition theorem and amplification by subsampling). Thus, taking $\gamma = 1/2\beta$, and setting $T = \frac{2\beta}{\mu}\log(\mu\beta\|h^0 - h^*\|^2/2M^2)$, where $M^2 = \frac{64\Lambda^2 T\log(2/\delta)}{n^2\epsilon^2}$, yields

$$\begin{aligned}\mathbb{E}\,\|h^T - h^*\|^2 &\leq \frac{2(T\log(3T/\delta)+1)M^2}{\beta\mu}\\ &\leq \frac{8M^2}{\mu^2}\log\Big(\frac{\mu\beta\|h^0-h^*\|^2}{2M^2}\Big)\log\Big(\frac{6\beta\log(\frac{\mu\beta\|h^0-h^*\|^2}{2M^2})}{\mu\delta}\Big)\ .\end{aligned} \tag{C.5.17}$$

Using Markov inequality, we obtain

$$\mathbb{P}\left(\|h^T - h^*\|^2 \geq \frac{8M^2}{\zeta\mu^2}\log\Big(\frac{\mu\beta\|h^0-h^*\|^2}{2M^2}\Big)\log\Big(\frac{6\beta\log(\frac{\mu\beta\|h^0-h^*\|^2}{2M^2})}{\mu\delta}\Big)\right) \leq \zeta\ . \tag{C.5.18}$$

This results in the following upper bound, with probability at least $1-\zeta$,

$$\begin{aligned}\|h^T - h^*\|^2 &\leq \frac{512\Lambda^2\log(3T/\delta)\log(2/\delta)}{\zeta\mu^2 n^2\epsilon^2}\log\Big(\frac{\mu\beta\|h^0-h^*\|^2}{2M^2}\Big)\log\Big(\frac{6\beta\log(\frac{\mu\beta\|h^0-h^*\|^2}{2M^2})}{\mu\delta}\Big)\\ &= \widetilde{O}\left(\frac{G^2\log(1/\delta)}{\zeta\mu^2 n^2\epsilon^2}\right)\ ,\end{aligned} \tag{C.5.19}$$

which is the result of our lemma. □

Appendix D

Experimental Details

D.1 Experimental Details for Chapter 4

D.1.1 Hyperparameter Tuning

DP-SGD and DP-CD both depend on three hyperparameters: step size, clipping threshold and number of passes on data. For DP-CD, step sizes are adapted from a parameter as described in Section 4.6, and clipping thresholds as well (see Section 4.5.1). For DP-SGD, the step size is given by γ/β, where γ is the hyperparameter and β is the problem's global smoothness constant (which we consider given), and the clipping threshold is used directly to clip gradients along their ℓ_2-norm.

We simultaneously tune these three hyperparameters for each algorithm across the following grid:

- step size: 10 logarithmically-spaced values between 10^{-6} and 1 for DP-SGD, and between 10^{-2} and 10 for DP-CD.[1]
- clipping threshold: 100 logarithmically-spaced values, between 10^{-3} and 10^{6}.
- number of passes: 5 values (2, 5, 10, 20 and 50).

We run each algorithm on each dataset 5 times on each combination of hyperparameter values. We then keep the set of hyperparameters that yield the lowest value of the objective at the last iterate, averaged across the 5 runs.

In Table D.1, we report the best relative error (in comparison to optimal objective value) at the last iterate, averaged over five runs, for each dataset, algorithm, and

[1] Recall that step sizes for CD algorithms are coordinate-wise, and thus larger than in SGD algorithms. We empirically verify that the best step size always lies strictly inside the considered interval for both DP-CD and DP-SGD.

Table D.1: Relative error to non-private optimal value of the objective function for different number of passes on the data. Results are reported for each dataset and for DP-CD and DP-SGD, after tuning step size and clipping hyperparameters. A star indicates the lowest error in each row. On each row, the first line is the utility of `DP-CD`, the second the one of `DP-SGD`. Privacy budget is $\epsilon = 1, \delta = 1/n^2$, except for the Sparse Lasso where $\epsilon = 10$.

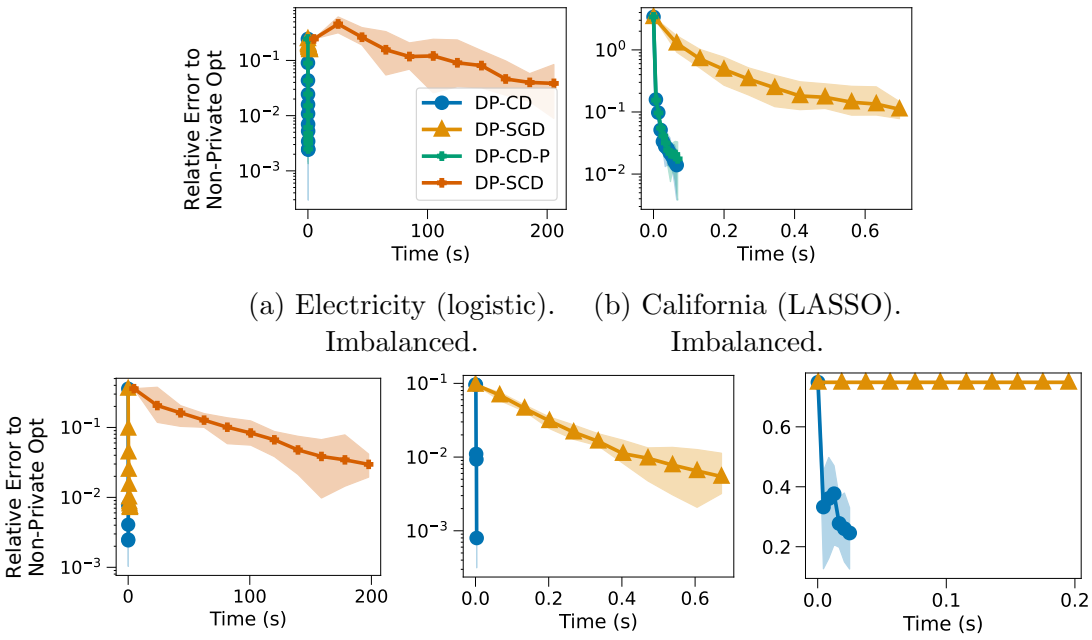
Passes on data	2	5	10	20	50
Electricity	0.1458 ± 6e-04	0.0842 ± 1e-03	0.0436 ± 2e-03	0.0147 ± 2e-03	0.0020 ± 1e-03*
Imbalanced	0.2047 ± 2e-02	0.1804 ± 2e-02	0.1766 ± 2e-02	0.1644 ± 2e-02	0.1484 ± 1e-02*
Electricity	0.0186 ± 4e-04	0.0023 ± 4e-04	0.0013 ± 6e-04*	0.0013 ± 4e-04	0.0019 ± 8e-04
Balanced	0.0391 ± 1e-02	0.0189 ± 5e-03	0.0123 ± 4e-03	0.0106 ± 3e-03	0.0040 ± 2e-03*
California	0.1708 ± 7e-03	0.1232 ± 1e-02	0.0598 ± 1e-02	0.0287 ± 5e-03	0.0124 ± 7e-03*
Imbalanced	0.2799 ± 9e-02	0.1863 ± 2e-02	0.1476 ± 2e-02	0.1094 ± 2e-02	0.1068 ± 2e-02*
California	0.0007 ± 3e-04*	0.0011 ± 6e-04	0.0012 ± 5e-04	0.0010 ± 1e-04	0.0017 ± 1e-03
Balanced	0.0351 ± 2e-02	0.0226 ± 8e-03	0.0125 ± 3e-03	0.0087 ± 2e-03	0.0042 ± 1e-03*
Sparse Lasso	0.2498 ± 4e-02*	0.4702 ± 9e-02	0.5982 ± 4e-02	0.7160 ± 2e-02	0.7551 ± 0e+00
Balanced	0.7551 ± 0e+00	0.7551 ± 3e-09*	0.7551 ± 0e+00	0.7551 ± 0e+00	0.7551 ± 0e+00

total number of passes on the data. As such, each cell of this table corresponds to the best value obtained after tuning the step size and clipping hyperparameters for a given number of passes.

D.1.2 Running Time

In this section, we report the running times of DP-CD and DP-SGD. We implemented DP-CD and DP-SGD in C++, with Python bindings. The design matrix and the labels are kept in memory as dense matrices of the Eigen library. No special code optimization nor tricks is applied to the algorithms, except for the update of residuals at each iteration of DP-CD, which prevents from accessing the complete dataset at each step. All experiments were run on a laptop with 16GB of RAM and an Intel(R) Core(TM) i7-10610U CPU @ 1.80GHz.

Figure D.2.3 shows the same experiments as in Figure 4.5.1 and Figure 4.6.1, but as a function of the running time. In our implementation, DP-CD runs about 4 times as fast as DP-SGD for a given number of iterations (see Figure D.1.1a and Figure D.1.1b for 50 iterations). On the three other plots, Figure D.1.1c, Figure D.1.1d and Figure D.1.1e, DP-CD yields better results in less iterations. DP-CD is thus particularly valuable in these scenarios: combined with its faster running time, it provides accurate results extremely fast. For completeness, we provide in Table D.2 the full table of running time, corresponding to Table D.1 and Figure D.2.3. These results show that, for a given number of passes on the data, DP-CD consistently runs about 5 times faster than DP-SGD.



(a) Electricity (logistic). Imbalanced.

(b) California (LASSO). Imbalanced.

(c) Electricity (logistic). Balanced.

(d) California (LASSO). Balanced.

(e) Sparse LASSO. Balanced.

Figure D.1.1: Relative error to non-private optimal for DP-CD (blue, round marks), DP-CD with privately estimated coordinate-wise smoothness constants (green, + marks) and DP-SGD (orange, triangle marks) on five problems. We report average, minimum and maximum values over 10 runs for each algorithm, as a function of the algorithm running time (in seconds).

Table D.2: Time of execution (in seconds) for different number of passes on the data (averaged over 10 runs). Results are reported for each dataset and for DP-CD and DP-SGD, after tuning step size and clipping hyperparameters. On each row, the first line is the utility of `DP-CD`, the second the one of `DP-SGD`. Privacy budget is $\epsilon = 1, \delta = 1/n^2$, except for the Sparse Lasso where $\epsilon = 10$.

Passes on data	2	5	10	20	50
Electricity	0.0128 ± 1e-03	0.0274 ± 1e-03	0.0500 ± 1e-03	0.0980 ± 7e-04	0.2457 ± 2e-03
Imbalanced	0.0663 ± 2e-03	0.1722 ± 1e-02	0.3321 ± 1e-02	0.6729 ± 1e-02	1.8588 ± 2e-01
Electricity	0.0121 ± 7e-04	0.0281 ± 3e-03	0.0529 ± 2e-03	0.1062 ± 6e-03	0.2577 ± 2e-03
Balanced	0.0686 ± 4e-03	0.1768 ± 1e-02	0.3578 ± 2e-02	0.6787 ± 2e-02	1.6766 ± 2e-02
California	0.0029 ± 9e-05	0.0065 ± 8e-05	0.0130 ± 1e-04	0.0258 ± 1e-04	0.0647 ± 2e-04
Imbalanced	0.0269 ± 1e-03	0.0665 ± 1e-03	0.1318 ± 2e-03	0.2628 ± 3e-03	0.6476 ± 8e-03
California	0.0031 ± 2e-04	0.0065 ± 2e-04	0.0132 ± 1e-04	0.0262 ± 2e-04	0.0649 ± 3e-04
Balanced	0.0261 ± 7e-04	0.0641 ± 5e-04	0.1295 ± 2e-03	0.2592 ± 4e-03	0.6469 ± 7e-03
Sparse LASSO	0.0244 ± 6e-04	0.0760 ± 6e-04	0.1614 ± 4e-03	0.3213 ± 5e-04	0.6598 ± 1e-02
Balanced	0.0718 ± 3e-03	0.1788 ± 4e-03	0.3654 ± 7e-03	0.7292 ± 2e-02	1.8110 ± 3e-02

D.2 Experimental Details for Chapter 5

In this section, we provide more information about the experiments, such as details on implementation, datasets and the hyperparameter grid we use for each algorithm. We then give the full results on our L1-regularized, non-smooth, problems, with the three greedy rules (as opposed to Section 5.4 where we only plotted results for the `GS-r` rule). Finally, we provide runtime plots.

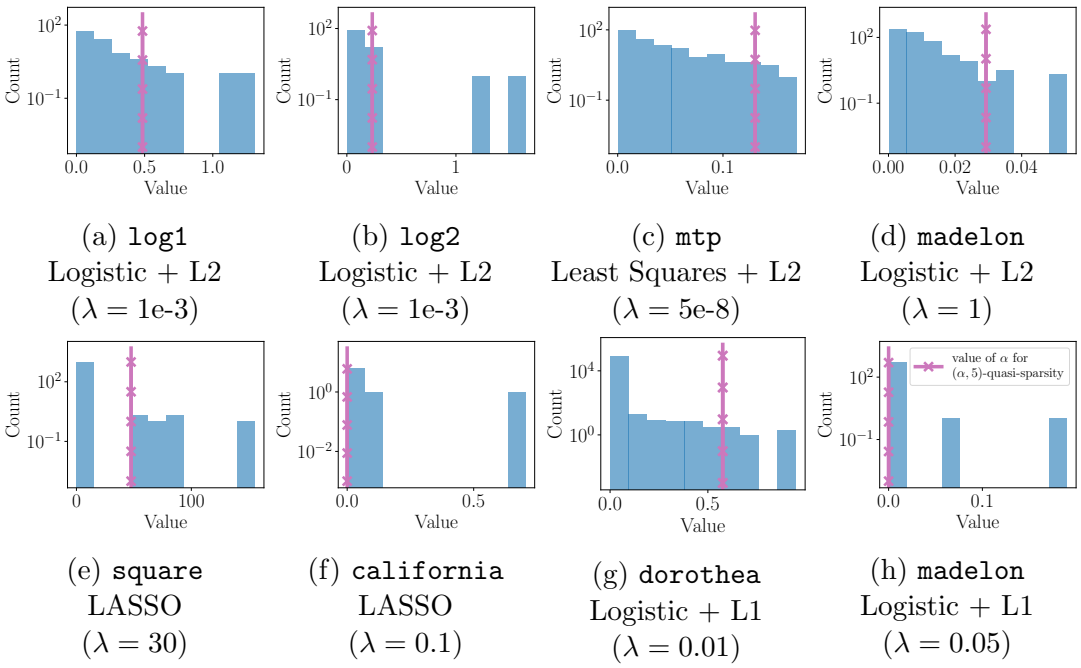
Code and setup. The algorithms are implemented in C++ for efficiency, together with a Python wrapper for simple use. It is provided as supplementary. Experiments are run on a computer with a Intel (R) Xeon(R) Silver 4114 CPU @ 2.20GHz and 64GB of RAM, and took about 10 hours in total to run (this includes all hyperparameter tuning).

Datasets. The datasets we use are described in Table 5.1. In Figure D.2.1, we plot the histograms of the absolute value of each problem solution's parameters. The purple line indicates the value of α that ensures that the parameters of the solution are $(\alpha, 5)$-quasi-sparse. Note the logarithmic scale on the y-axis. On the `log1`, `log2`, `madelon`, `square`, `california` and `dorothea` datasets, the solutions are very imbalanced. In these problems, a very limited number of parameters stand out, and DP-GCD is able to exploit this property. This illustrates the results from Section 5.3.4, since DP-GCD can exploit this structure even in quasi-sparse problems, where α is non zero. Conversely, the `mtp` solution is more balanced: the structural properties of this dataset are not strong enough for DP-GCD to outperform its competitors.

Hyperparameters. On all datasets, we use the same hyperparameter grid. For each algorithm, we choose between roughly the same number of hyperparameters. The number of passes on data represents p iterations of DP-CD, n iterations of DP-SGD, and 1 iteration of DP-GCD. The complete grid is described in Table D.3, and the chosen hyperparameters for each problem and algorithm are given in Table D.5.

Recovery of the support. In Table D.4, we report the number of coordinates that are correctly/incorrectly identified as non-zero on ℓ_1 regularized problems. Contrary to DP-SGD and DP-CD, DP-GCD never incorrectly identifies a coordinate as non-zero. Additionally, the suboptimality gap is lower for DP-GCD: its updates thus lead to better solutions.

Additional experiments on proximal DP-GCD. In Figure D.2.2, we show the results of the proximal DP-GCD algorithm, after tuning the hyperparameters with the grid described above for each of the `GS-s`, `GS-r` and `GS-q` rules.



(a) `log1` Logistic + L2 ($\lambda = 1\text{e-}3$)
(b) `log2` Logistic + L2 ($\lambda = 1\text{e-}3$)
(c) `mtp` Least Squares + L2 ($\lambda = 5\text{e-}8$)
(d) `madelon` Logistic + L2 ($\lambda = 1$)
(e) `square` LASSO ($\lambda = 30$)
(f) `california` LASSO ($\lambda = 0.1$)
(g) `dorothea` Logistic + L1 ($\lambda = 0.01$)
(h) `madelon` Logistic + L1 ($\lambda = 0.05$)

Figure D.2.1: Histograms of the absolute value of each problem solution's parameters. Purple line indicates the α for which the plotted vector is $(\alpha, 5)$-quasi-sparse. Y-axis is logarithmic.

The three rules seem to behave qualitatively the same on `square`, `dorothea` and `madelon`, our three high-dimensional non-smooth problems. There, most coordinates are chosen about one time. Thus, as described by Karimireddy et al. (2019), all the steps are "`good`" steps (along their terminology): and on such good steps, the three rules coincide. On the lower-dimensional dataset `california`, coordinates can be chosen more than one time, and "`bad`" steps are likely to happen. On these steps, the three rules differ.

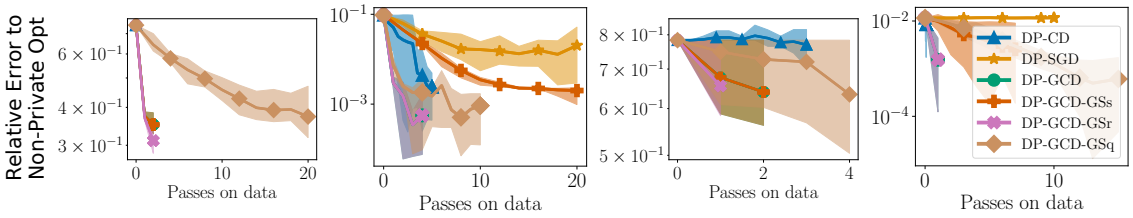
Runtime. Finally, we report the runtime of DP-GCD, in comparison with DP-CD and DP-SGD in Figure D.2.3, that is the counterpart of Figure 5.4.1, except with runtime on the x-axis. These results confirm the fact that DP-GCD can be efficient, although its iterations are expensive to compute. Indeed, in imbalanced problems, the small number of iterations of DP-GCD enables it to run faster than DP-SGD, and in roughly the same time as DP-CD, while improving utility.

Table D.3: Hyperparameter grid used in our experiments.

Algorithm	Parameter	Values
DP-CD	Passes on data	`[0.001, 0.01, 0.1, 1, 2, 3, 5, 10, 20]`
	Step sizes	`np.logspace(-2, 1, 10)`
	Clipping threshold	`np.logspace(-4, 6, 50)`
DP-SGD	Passes on data	`[0.001, 0.01, 0.1, 1, 2, 3, 5, 10, 20]`
	Step sizes	`np.logspace(-6, 0, 10)`
	Clipping threshold	`np.logspace(-4, 6, 50)`
DP-GCD	Passes on data	`[1, 2, 4, 7, 10, 15, 20]`
	Step sizes	`np.logspace(-2, 1, 10)`
	Clipping threshold	`np.logspace(-4, 6, 50)`

Table D.4: Coordinates correctly/incorrectly identified as non-zeros by each algorithm, and relative suboptimality gap $(f(w^{priv}) - f^*)/f^*$ (averaged over 5 runs).

	`square`	`california`	`dorothea`	`madelon`
$\|w^*\|_0$	7	3	72	3
DP-CD	0 / 0 (0.75)	3 / 2 (0.0024)	1 / 1 (0.77)	0 / 0 (0.0085)
DP-SGD	0 / 3 (0.75)	3 / 5 (0.020)	0 / 0 (0.78)	0 / 0 (0.012)
DP-GCD	2 / 0 (0.35)	2 / 0 (0.00056)	1 / 0 (0.64)	1 / 0 (0.0015)



(a) `sparse` LASSO ($\lambda = 30$)

(b) `california` LASSO ($\lambda = 0.1$)

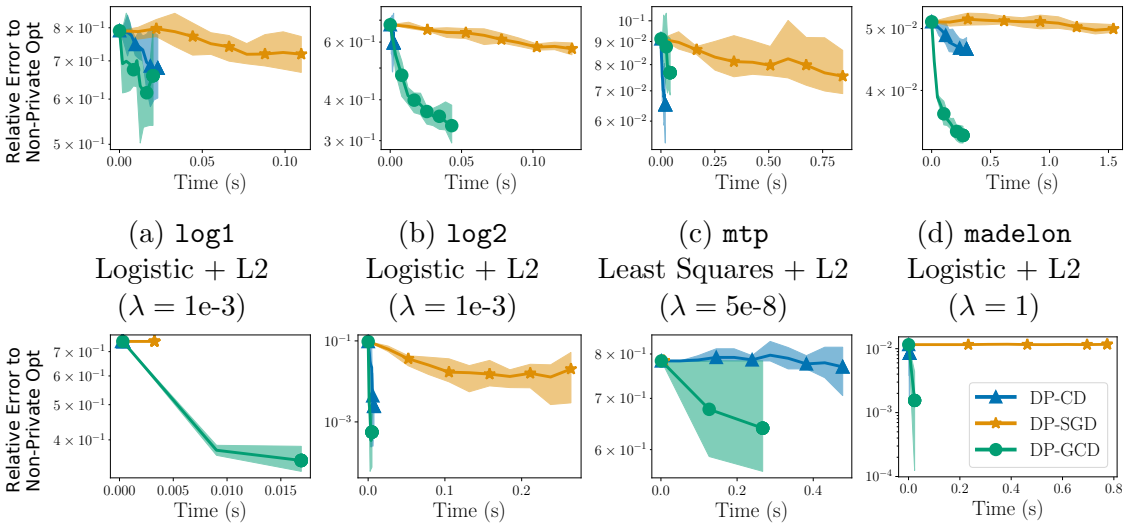
(c) `dorothea` Logistic + L1 ($\lambda = 0.01$)

(d) `madelon` Logistic + L1 ($\lambda = 0.05$)

Figure D.2.2: Relative error to non-private optimal for DP-CD, proximal DP-GCD (with `GS-r`, `GS-s` and `GS-q` rules) and DP-SGD on different problems. On the x-axis, 1 tick represents a full access to the data: p iterations of DP-CD, n iterations of DP-SGD and 1 iteration of DP-GCD. Number of iterations, clipping thresholds and step sizes are tuned simultaneously for each algorithm. We report min/mean/max values over 5 runs.

Table D.5: Selected hyperparameters for every dataset and algorithm.

Dataset	Loss	Algorithm	Passes on data	Clipping threshold	Step size
california	LeastSquares + L1	DP-CD	5.0	2.02e+01	1.00e+00
square	LeastSquares + L1	DP-CD	0.01	9.10e+03	1.00e+01
mtp	LeastSquares + L2	DP-CD	3.0	2.02e+01	2.15e-02
madelon	Logistic + L1	DP-CD	0.1	7.91e+00	2.15e+00
log1	Logistic + L2	DP-CD	10.0	1.84e-01	1.00e+00
log2	Logistic + L2	DP-CD	1.0	7.54e-01	2.15e+00
madelon	Logistic + L2	DP-CD	10.0	1.21e+00	1.00e-01
dorothea	Logistic + L1	DP-CD	3.0	4.50e-02	4.64e+00
california	LeastSquares + L1	DP-SGD	20.0	1.26e+01	2.15e-05
square	LeastSquares + L1	DP-SGD	0.01	4.94e+00	1.00e-04
mtp	LeastSquares + L2	DP-SGD	20.0	1.26e+01	2.15e-05
madelon	Logistic + L1	DP-SGD	10.0	6.87e-03	1.00e+00
log1	Logistic + L2	DP-SGD	20.0	1.84e-01	4.64e-04
log2	Logistic + L2	DP-SGD	20.0	1.84e-01	4.64e-04
madelon	Logistic + L2	DP-SGD	20.0	1.84e-01	1.00e-04
dorothea	Logistic + L1	DP-SGD	0.001	1.00e-04	1.00e-06
california	LeastSquares + L1	DP-GCD	4	5.18e+01	1.00e+00
square	LeastSquares + L1	DP-GCD	2	1.46e+04	2.15e+00
mtp	LeastSquares + L2	DP-GCD	7	2.02e+01	4.64e-01
madelon	Logistic + L1	DP-GCD	1	7.91e+00	2.15e+00
log1	Logistic + L2	DP-GCD	10	3.09e+00	2.15e+00
log2	Logistic + L2	DP-GCD	20	1.93e+00	4.64e-01
madelon	Logistic + L2	DP-GCD	10	7.91e+00	1.00e+00
dorothea	Logistic + L1	DP-GCD	2	1.26e+01	2.15e+00



(a) `log1` Logistic + L2 (λ = 1e-3)

(b) `log2` Logistic + L2 (λ = 1e-3)

(c) `mtp` Least Squares + L2 (λ = 5e-8)

(d) `madelon` Logistic + L2 (λ = 1)

(e) `sparse` LASSO (λ = 30)

(f) `california` LASSO (λ = 0.1)

(g) `dorothea` Logistic + L1 (λ = 0.01)

(h) `madelon` Logistic + L1 (λ = 0.05)

Figure D.2.3: Relative error to non-private optimal for DP-CD, DP-GCD and DP-SGD on different problems, as a function of running time. Number of iterations, clipping thresholds and step sizes are tuned simultaneously for each algorithm. We report min/mean/max values over 5 runs.

D.3 Experimental Details for Chapter 6

D.3.1 Experimental Setup

The first dataset is the `celebA` dataset (Liu et al., 2015). It is a face attributes dataset, that can be downloaded at `http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html`. The second dataset is the `folktables` dataset (Ding et al., 2021). It is derived from US Census, and can be downloaded using a Python package available here `https://github.com/zykls/folktables`.

On each dataset, for each value of n, we train a ℓ_2-regularized logistic regression model using `scikit-learn` (Pedregosa et al., 2011). Private models are then learned using the output perturbation mechanism as described in Section 6.5.1. We then compute our bounds using the non-private model as reference, over a test set containing 10% of the data, that has not been used for training (containing $20,260$ records for `celebA` and $166,450$ records for `folktables`). The value of the bound is computed by minimizing the experession given by the Chernoff bound using the golden section search algorithm (Kiefer, 1953). The code is in the supplementary, and will be made public.

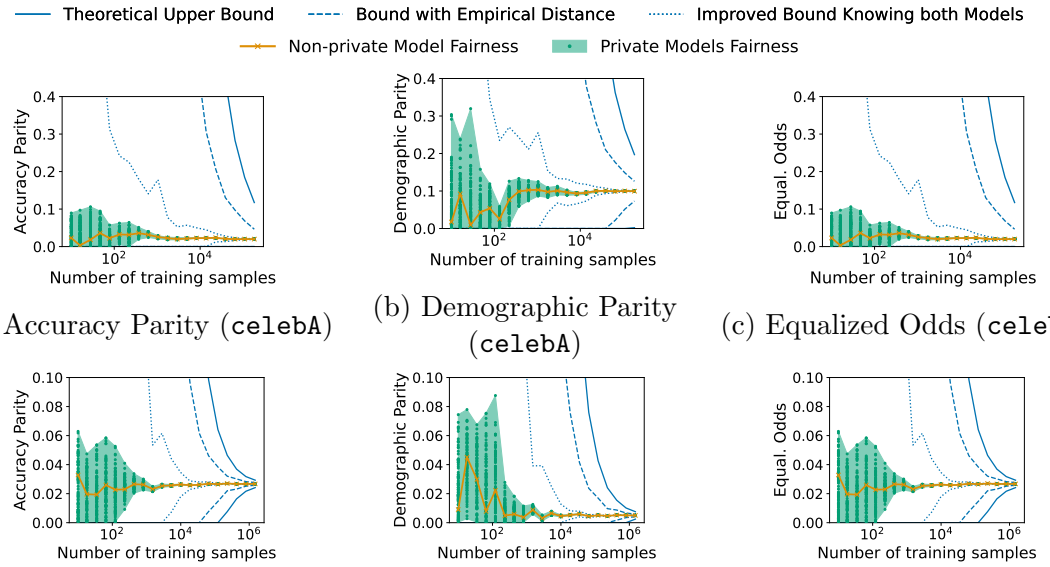
For the plots with different number of training records, we train 20 non-private models with a number of records logarithmically spaced between 10 and the number of records in the complete training set (that is, $182,339$ for `celebA` and $1,498,050$ for `folktables`). For the plots with different privacy budgets, we use 20 values logarithmically spaced between 10^{-3} and 10 for both datasets.

D.3.2 Results for Other Fairness Measures

Our bounds also hold for accuracy parity, demographic parity and equalized odds. The same plots as those presented in Figure 6.6.1 for these fairness notions are in Figure D.3.1 and Figure D.3.2. The comments from Section 6.6 on equality of opportunity and accuracy also hold for these three notions of fairness.

D.3.3 Refined Bounds with Additional Knowledge of $h^{\mathbf{priv}}$ and h^*

In Assumption 6.3.1, we use a uniform Lipschitz bound for all $h, h' \in \mathcal{H}$. Let's consider the class $\mathcal{H}$ of linear models, where, for $h \in \mathcal{H}$, we denote by h_y the parameters of h associated with the label y, that is $h(x, y) = h_y^T x$. For linear models, we derived the bound $\|\rho(h, x, y) - \rho(h', x, y)\|_{\mathcal{H}} \leq 2\|x\|_2 \|h - h'\|_{\mathcal{H}}$, as derived in Section 6.3. Note that this inequality can be very loose whenever x and $h_y - h'_y$ (for $y \in \mathcal{Y}$) are (close to) orthogonal. When they are orthogonal, this bounds only gives $0 = (h_y - h'_y)^T x \leq$



(a) Accuracy Parity (`celebA`) (b) Demographic Parity (`celebA`) (c) Equalized Odds (`celebA`)

(d) Accuracy Parity (`folktables`) (e) Demographic Parity (`folktables`) (f) Equalized Odds (`folktables`)

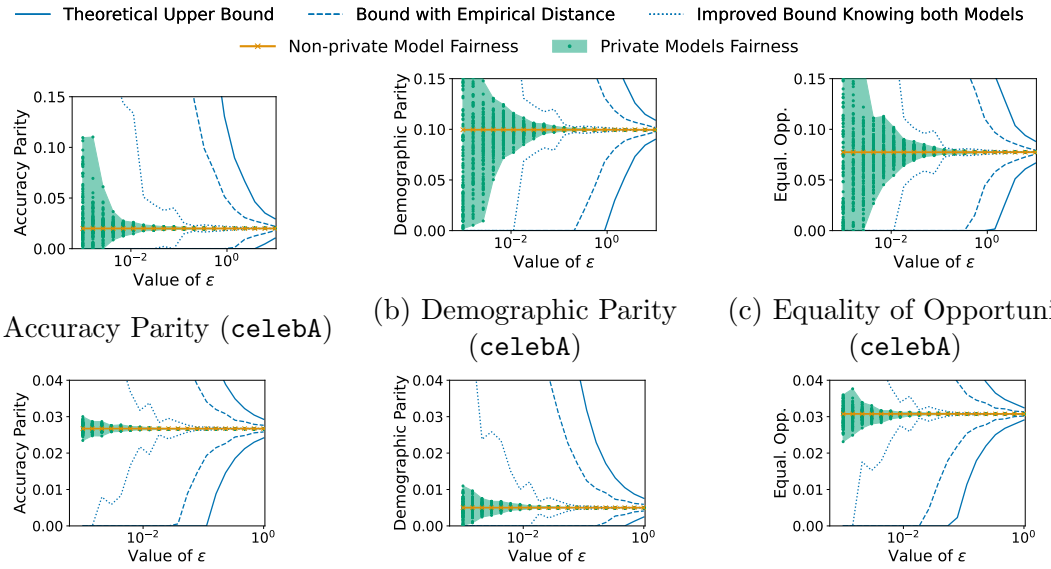
Figure D.3.1: Fairness and accuracy levels for optimal non-private model and random private ones as a function of the number n of training samples. For each value of n, we sample 100 private models and take their minimum and maximum fairness/accuracy values to mark the area of attainable values. The solid blue line and the dashed one give our guarantees, respectively from Theorem 6.5.1 with Lemma 6.5.1's bounds and with an empirical evaluation of $\|h^{\text{priv}} - h^*\|$.

$\|h_y - h'_y\|_2 \|x\|_2$. We can thus improve the inequality by remarking that we have

$$\begin{aligned}
|\rho(h,x,y) - \rho(h',x,y)| &\leq |h(x,y) - h'(x,y)| + \max_{y' \neq y} |h(x,y') - h'(x,y')| \\
&= |h_y^T x - h_y'^T x| + \max_{y' \neq y} |h_{y'}^T x - h'_{y'} x| \\
&= |(h_y - h'_y)^T x| + \max_{y' \neq y} |(h_{y'} - h'_{y'}) x| \\
&= |(h_y - h'_y)^T p_{h_y - h'_y}(x)| + \max_{y' \neq y} |(h_{y'} - h'_{y'}) p_{h_y - h'_y}(x)| \\
&\leq 2 \max_{y' \in \mathcal{Y}} \|p_{h_y - h'_y}(x)\| \|h - h'\|_{\mathcal{H}} \ ,
\end{aligned}$$

where $p_{h_y - h'_y}(x)$ is the projection of x on the axis defined by $h_y - h'_y$. We can thus define a variant of $L_{X,Y}$ which depends on $h - h'$

$$L_{X,Y}^{h-h'} = 2 \max_{y \in \mathcal{Y}} \|p_{h_y - h'_y}(x)\| \ . \tag{D.3.1}$$



(a) Accuracy Parity (`celebA`)

(b) Demographic Parity (`celebA`)

(c) Equality of Opportunity (`celebA`)

(d) Accuracy Parity (`folktables`)

(e) Demographic Parity (`folktables`)

(f) Equality of Opportunity (`folktables`)

Figure D.3.2: Fairness and accuracy levels for optimal non-private model and random private ones as a function of privacy budget ϵ. For each value of ϵ, we sample 100 private models and take their minimum and maximum fairness/accuracy values to mark the area of attainable values. The solid blue line and the dashed one respectively give our guarantees, respectively from Theorem 6.5.1 with Lemma 6.5.1's bounds and with an empirical evaluation of $\|h^{\text{priv}} - h^*\|$.

Replacing Assumption 6.3.1 by this inequality in the proof of Theorem 6.4.1, we end up with the inequality

$$|\,\mathbb{P}\left(H(X) = Y \mid E\right) - \mathbb{P}\left(H'(X) = Y \mid E\right)| \leq \mathbb{P}\left(\frac{|\rho(h, X, Y)|}{L_{X,Y}^{h-h'}} \leq \|h - h'\|_{\mathcal{H}} \mid E\right) \quad ,$$

where the probability is over $(X, S, Y) \sim \mathcal{D}$. We obtained the same bound as Theorem 6.4.1, except with $L_{X,Y}^{h-h'}$ instead of $L_{X,Y}$. Note that even if this gives a much tighter bound, this can generally not be computed, as one of h or h' is typically not known.