The Pennsylvania State University
The Graduate School

DIFFERENTIALLY PRIVATE CONVEX OPTIMIZATION FOR EMPIRICAL RISK MINIMIZATION AND HIGH-DIMENSIONAL REGRESSION

A Dissertation in
Computer Science and Engineering
by
Abhradeep Guha Thakurta

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2013

The dissertation of Abhradeep Guha Thakurta was reviewed and approved* by the following:

Adam D. Smith
Associate Professor of Computer Science and Engineering
Chair of Committee and Dissertation Advisor

Aleksandra B. Slavković
Associate Professor of Statistics and Public Health Sciences

Daniel Kifer
Assistant Professor of Computer Science and Engineering

Sofya Raskhodnikova
Associate Professor of Computer Science and Engineering

Raj Acharya
Head of Department of Computer Science and Engineering

*Signatures are on file in the Graduate School.

Abstract

Learning systems are the backbone of most web-scale advertisement and recommendation systems. Such systems rely on past inputs from users to decide on a particular advertisement or recommendation to be displayed to a new user. The way these learning systems work is by first recording past user responses (collectively called the *training* data) and then learning a prediction model which decides on a relevant advertisement or recommendation for a new user. Often the training data sets contain sensitive information about users (e.g., sexual orientation or marital status). Recent research has shown that these large-scale learning systems can inadvertently leak sensitive information about individual users in the training data [Korolova, 2010; Calandrino et al., 2011]. This leads us to think about designing learning algorithms which do not leak "too much" information about any individual (user) in the training data.

In this dissertation we design learning algorithms with rigorous privacy guarantees. We adhere to a formal well-accepted notion of privacy, called *differential privacy* [Dwork et al., 2006b]. Differential privacy guarantees that an algorithm's output does not depend too much on the data of any individual in the data set. This is crucial in fields that handle sensitive data, such as genomics, collaborative filtering, and economics.

In our work we design differentially private algorithms for the following two sets of learning problems: i) convex optimization for empirical risk minimization (the most common use of convex optimization in machine learning), and ii) sparse regression (a broad class of high-dimensional problems). For both these problems, we design differentially private algorithms that are provably (almost) as accurate as the best non-private algorithm.

Table of Contents

List of Tables

Acknowledgments

Coming from an undergraduate background in electrical (power) engineering, theoretical computer science was mostly an unchartered territory for me. It took me significant time to appreciate the fundamental difference between *verifying* the correctness of a sequence of arguments leading to a theorem and *understanding* the main ideas behind a proof. It was Adam Smith, my academic advsior, who forced me to explain the main ideas in a research paper to him without writing equations on a white board. This was probably the single most important lesson I learnt during my PhD and I will be ever thankful to Adam for it. In terms of research, Adam has been a constant source of interesting problems and inspiring ideas. Most of the works in this dissertation are results of these ideas. I am really lucky to have him as my advisor.

I would like to thank Prateek Jain (my mentor and collaborator from Microsoft Research India), who was one of the driving forces towards me getting interested in machine learning. My collaboration with him has played a significant role in choosing the direction of my PhD research and eventually this dissertation. His constant efforts in explaining concepts from learning theory gave me insights which would have been much harder to get from any text book or a research paper. I would also like to thank my other collaborators, Raghav Bhaskar, Abhishek Bhowmick, Misha Bilenko, Kashyap Dixit, Vipul Goyal, Madhav Jha, Daniel Kifer, Pravesk Kothari, Prashanth Mohan, Srivatsan Laxman, Sofya Raskhodnikova, Elaine Shi, Dawn Song and Helen Wang without whom this PhD would not have been as enjoyable.

I would like to thank all the academic institutions and industry labs which both supported me financially and hosted me either as a visitor or as an intern. To that end, I am highly thankful to Microsoft for hosting me for as many as five internships/jobs (starting from my undergraduate internship in 2006). I have been either an intern or a fulltime employee in the following Microsoft offices: Microsoft India Development Center, Hyderabad, Microsoft Research India, Microsoft Research Redmond. I would also like to thank University of California at Berkeley for hosting me as a visiting student for the spring of 2011. Although I never worked for Yahoo Research, I am very thankful to them for honoring me with the Key Scientific Challenges award in 2011. Pennsylvania State University being my alma mater and my home for my entire PhD life, no amount of acknowledgement is sufficient to show my gratitude.

Some of the other people whom I would like to thank for being instrumental in

shaping my academic career are Joydeep Chakraborty (my high-school teacher), Vijayan Immanuel (my undergraduate advisor), Venkatesh Phadnavis (my undergraduate mentor at Microsoft) and Sitaram Raju (my manager at Microsoft). When I was coming for a PhD to United States, it was Sitaram who suggested that it might be worthwhile in my PhD to investigate privacy and security aspects of database systems.

Above all I would like to thank my parents for being constantly supportive in every decision I took in my academic and personal life and for protecting me from uncountable troubles and hardships I would have had to go through otherwise.

Dedication

To my parents and my family.

Chapter 1

Introduction

In the past few years, the web has transformed itself from being a collection of static web pages to being more interactive and "social". Since the arrival of social networking sites such as Facebook, recommendation systems such as Amazon, social search features of Google and Bing, the content on a webpage is largely dictated by its users. For example, the advertisements that are shown on the side bar of Facebook or the "You might also like..." box of Amazon are based on the choices made by a large pool of users with similar tastes and their responses to various advertisements or online recommendations.

Although these developments have significant benefits to the society, the websites' use of the data they store (with or without users' consent) poses significant privacy concerns. For example, a person who does not share his sexual orientation on the public Facebook profile may be seriously concerned if Facebook leaks this information. Even though most social networking sites or recommendation systems base their general decisions to display an advertisement or to recommend a product on partly anonymized data, past studies ([Korolova, 2010; Calandrino et al., 2011]) have shown that even with such anonymization, it is possible to infer specific information about a particular individual from various outputs like advertisements or recommendations. More generally, the need for preserving privacy of users (or individuals), while analyzing big data sets, spans many domains of science as evidenced by statistics released by National Institute of Health (NIH) [Homer et al., 2008] and by the US census bureau [Machanavajjhala et al., 2008].

> *In this dissertation, we develop new, computationally efficient algorithms that provide accurate analysis of large, sensitive data sets while satisfying rigorous privacy guarantees.*

In our work we adhere to the notion of *differential privacy* [Dwork et al., 2006b] which is one of the most well-accepted privacy definitions. Intuitively, differential privacy ensures that an attacker learns the same information about a user whether or not his data is present in the data set. Recent works, surveyed in [Dwork, 2006, 2008], have shown that for a large class of analyses one can guarantee differential privacy and yet incur very little degradation in accuracy.

Instead of analyzing privacy properties of specific advertisement or recommendation systems, such as Facebook and Amazon, we take a more unified approach. We study the privacy properties of a mathematical model called *convex optimization*, which is the primary building block for most of these systems. Convex optimization is a standard abstraction used in artificial intelligence and machine learning. The main advantage of the convex optimization formulation is that there exist algorithms for convex optimization that are computationally efficient even on large data sets. However, the outputs of these algorithms can leak a lot of information about users in the data set they are operating on [Chaudhuri et al., 2011]. In our work we design differentially private algorithms for convex optimization which enjoy (almost) the same utility guarantees as their non-private counterpart. Our results for private convex optimization are in two different regimes.

1. **The *classical* regime:** Here the number of entries in the underlying data set (n) is much larger than the number of parameters in the model (p). Only in this setting, most common optimization algorithms produce meaningful results. In this regime, we consider a special class of convex optimization problems called *empirical risk minimization* (ERM) problems. We provide algorithms for private ERM, for which the *excess* error (empirical risk) due to privacy scales roughly as $\tilde{O}(p^{2.5}/n)$.

2. **The *high-dimensional* regime:** Here the number of parameters in the model (p) is much larger than the number of entries in the underlying data set (n). With the growing complexity of the online systems, in many scenarios it is no longer true that the data set size is larger than the number of parameters in the model. In fact in typical advertisement systems, the number of parameters in the model can be orders of magnitude larger than the size of the data set. A relatively recent branch of statistics (called *high-dimensional regression*) allows one to obtain reasonable solutions to convex optimization problems even under this setting. In our work we design algorithms that guarantee differential privacy for a class of problems pertaining to high-dimensional regression, called *sparse regression*.

Our contribution in more detail: In the classical (*"small p,large n"*) regime, we extend the existing analysis of [Chaudhuri et al., 2011] to capture a larger class of ERM problems compared to [Chaudhuri et al., 2011]. Moreover, we provide stronger utility guarantees which have better dependence on the dimensionality (p) of the parameters in the model. For the special case of *linear regression* (a popular ERM problem), we improve the utility guarantees further.

We further study the ERM problem in the classical regime from an online learning perspective. In online learning, the general setting is that the data entries arrive online. Every time a data entry arrives, the learning algorithm makes a prediction. After the prediction is made, the algorithm suffers a loss based on the prediction. The objective is to incur "low" cumulative loss, which is usually formalized by the notion of *regret*. We use the tools of online learning, namely *online convex programming* (OCP), to design algorithms for private ERM in the classical regime. Although the algorithms obtained via online learning have worse utility guarantees, in terms of the assumptions necessary to guarantee differential privacy they seem to be more applicable in practical scenarios.

In the high-dimensional setting (*"large p, small n"*) regime, we design the first differentially private algorithms for sparse regression. Roughly speaking, in sparse regression the underlying model parameters are assumed to have very few non-zero entries (compared to the dimensionality p). For high-dimensional problems, guaranteeing privacy is more challenging since the systems that use high-dimensional algorithms tend to release a lot of information about the underlying data set. The difficulty of guaranteeing privacy for high-dimensional problems was highlighted by [Narayanan and Shmatikov, 2009] where they attacked such a system (namely, the Netflix prize data) to extract movie preferences of specific users in the data set. We show that for "typical" data sets, the output of our algorithms will have very little degradation of accuracy due to privacy.

Note: Some of the results mentioned in this dissertation have been published in the following three papers: [Jain et al., 2012; Kifer et al., 2012; Smith and Thakurta, 2012].

1.1 Structure of this Dissertation

Chapter 2 [Preliminaries]: We first provide a brief introduction to the concept of differential privacy and review two basic tools commonly used in the design of differentially private algorithms, namely, the *Laplace mechanism* [Dwork et al., 2006b] and the *exponential mechanism* [McSherry and Talwar, 2007].

All the algorithms in this dissertation are randomized. In most cases, they sample random variables from some well-known distributions. The utility guarantees of the algorithms depend on the tail behavior of these distributions. In this chapter we review some of the distributions used in this work and provide details about their tail behavior. Finally, since this dissertation is about convex ERM, we provide a very short introduction to convex functions.

Chapter 3 [Private ERM in the classical regime]: We significantly extend the analysis of the *objective perturbation* algorithm of [Chaudhuri et al., 2011] for convex ERM problems. We show that their method can be modified to use less noise (be more accurate), and to apply to problems with hard constraints and non-differentiable regularizers. We also give a tighter, data-dependent analysis of the additional error introduced by their method.

A key tool in our analysis is a new nontrivial limit theorem for differential privacy which is of independent interest: If a sequence of differentially private algorithms converges, in a *weak* sense, then the limit algorithm is also differentially private.

Chapter 4 [Private linear regression in the classical regime]: We instantiate the private ERM algorithms designed in Chapter 3 with the specific problem of linear regression. We further show that specifically for linear regression, one can design better differentially private algorithms compared to the generic private ERM algorithms. We leave the following as an open problem: *Can the algorithms specific to linear regression be extended to other ERMs?*

Chapter 5 [Private online learning and online learning based ERM]: We consider the problem of preserving privacy in the context of online learning. We study the problem in the framework of online convex programming (OCP)—a popular online learning setting with several theoretical and practical implications—while using differential privacy as the formal measure of privacy. For this problem, we provide a generic framework that can be used to convert any given OCP algorithm into a private OCP algorithm with provable privacy as well as utility (*regret*) guarantees. Finally, we show that our online learning framework can be used to provide differentially private algorithms for convex ERM problems in the classical regime. The assumptions for the privacy guarantees of these algorithms seem to be applicable in more practical scenarios compared to the algorithms in Chapter 3. However, there is a slight degradation in utility as compared to the algorithms from Chapter 3.

Chapter 6 [Private model selection and sparse regression]: We design differentially private algorithms for statistical model selection. Given a data set and a large, discrete collection of "models", each of which is a family of probability distributions, the goal is to determine the model that best "fits" the data. This is a basic problem in many areas of statistics and machine learning.

We consider settings in which there is a well-defined answer, in the following sense: Suppose that there is a *nonprivate* model selection procedure f, which is the reference to which we compare our performance. Our differentially private algorithms output the correct value $f(\mathcal{D})$ whenever f is *stable* on the input data set $\mathcal{D}$. We work with two notions, *perturbation* stability and *subsampling* stability.

We give two classes of results: generic ones, that apply to any function with discrete output set; and specific algorithms for the problem of sparse linear regression. The algorithms we describe are efficient and in some cases match the optimal *nonprivate* asymptotic sample complexity.

Our algorithms for sparse linear regression require analyzing the stability properties of the popular LASSO estimator. We give sufficient conditions for the LASSO estimator to be robust to small changes in the data set (i.e., addition or removal of a few entries in the data set), and show that these conditions hold with high probability under essentially the same stochastic assumptions that are used in the literature to analyze consistency of the LASSO.

Chapter 7 [Concluding remarks]: We conclude with some of the unanswered questions that came up along the course of this work and discuss possible approaches towards addressing them.

Chapter 2

Definitions and Preliminaries

2.1 Differential Privacy

Our algorithms satisfy differential privacy [Dwork et al., 2006b,a] which bounds the effect of any single record on the distribution of the released information. Intuitively, differential privacy ensures that an attacker learns the same information about an individual independent of his presence and absence in the data set. More formally, differential privacy ensures that on two neighboring data sets $\mathcal{D}$ and $\mathcal{D}'$ (which differ in exactly one data entry) the distributions induced by any randomized algorithm on the space of possible outputs are statistically close. The parameters (ϵ, δ), which are also called *privacy budgets*, measure the statistical closeness.

Definition 2.1 (Differential privacy [Dwork et al., 2006b,a])**.** *A randomized algorithm $\mathcal{A}$ is (ϵ, δ)-differentially private if for any two datasets $\mathcal{D}$ and $\mathcal{D}'$ drawn from a domain $\mathcal{T}^*$ with $|\mathcal{D} \Delta \mathcal{D}'| = 1$ (Δ being the symmetric difference), and for all (Borel) sets $\mathcal{O} \subseteq Range(\mathcal{A})$ the following holds:*

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

In the following we will discuss two basic tools commonly used in the design and analysis of differentially private algorithms. We frequently use these tools in this dissertation.

2.1.1 Laplace Mechanism

Differentially private algorithms must be randomized, since they must blur the distinction between two neighboring data sets $\mathcal{D}$ and $\mathcal{D}'$. A common technique to introduce randomness is the addition of Laplace noise to the outputs. Suppose that we would like to release (an approximation to) a vector of real-valued statistics. That is, for some function $f : \mathcal{T}^* \to \mathbb{R}^p$ and data set $\mathcal{D} \in \mathcal{T}^*$, we would like to release an approximation close to $f(\mathcal{D})$. [Dwork et al., 2006b] showed that it suffices to add noise proportional to

the *sensitivity* of the function f. Sensitivity measures the maximum possible change in the value of f when one entry from the data set is removed or added.

Definition 2.2 (Global sensitivity)**.** *The sensitivity of a function $f : \mathcal{T}^* \to \mathbb{R}^p$ is the smallest number such that for all data sets $\mathcal{D}$ and $\mathcal{D}'$ differing in exactly one data entry, $\|f(\mathcal{D}) - f(\mathcal{D}')\|_1 \leq \Delta f$.*

Consider the randomized algorithm $\mathcal{A}_f$ that computes $f(\mathcal{D})$ and releases $\tilde{f}(\mathcal{D}) = f(\mathcal{D}) + Lap\left(\frac{\Delta f}{\epsilon}\right)^p$, where $Lap(\lambda)$ denotes a vector of p i.i.d. samples from the Laplace distribution $Lap(\lambda)$. Recall that $Lap(\lambda)$ is the distribution on $\mathbb{R}$ with density at y given by $\frac{1}{\lambda}e^{-\frac{|y|}{\lambda}}$. [Dwork et al., 2006b] showed that $\mathcal{A}_f$ is $(\epsilon, 0)$ (or simply ϵ) differentially private. The standard deviation of $Lap(\lambda)$ is $\lambda\sqrt{2}$, so the algorithm adds noise proportional to $\frac{\Delta f}{\epsilon}$.

2.1.2 Exponential Mechanism

Exponential mechanism, initially proposed by [McSherry and Talwar, 2007], is a generic framework for designing differentially private algorithms. Unlike the *Laplace mechanism* above, the output of the function f can be non-numeric (e.g., categorical). The exponential mechanism is a family of algorithms, parameterized by a set $\mathcal{R}$ of possible outputs (called the *range*) and a real-valued function $q : \mathcal{T}^* \times \mathcal{R} \to \mathbb{R}$ that assigns each possible output $r \in \mathcal{R}$ a *score* $q(\mathcal{D}, r)$ based on the data set $\mathcal{D}$. Given $\mathcal{R}, q, \mathcal{D}$ and privacy parameter ϵ, the goal is to produce an output with as high a score as possible, while satisfying ϵ-differential privacy. To this end, the algorithm draws a single sample from the distribution on $\mathcal{R}$ which assigns each element $r \in \mathcal{R}$ probability mass proportional to $\exp\left(\frac{\epsilon q(\mathcal{D},r)}{2\Delta q}\right)$. Here Δq is the maximum of the global sensitivities of the functions $q(\cdot, r)$. Intuitively this mechanism is useful since it assigns high mass to elements with high scores. [McSherry and Talwar, 2007] showed that this algorithm is ϵ-differentially private.

2.2 Noise distributions

In this dissertation we sample noise from various distributions to introduce randomness in the algorithm, so that the output is differentially private. The three noise distributions we use are given below. Along with the noise distribution, we also briefly state the tail properties of these distributions.

Laplace distribution: The density function of a mean zero Laplace distribution in one dimension is given by $Lap(\lambda) = \frac{1}{2\lambda}e^{-\frac{|x|}{\lambda}}$, where $x \in \mathbb{R}$ and λ is the scaling parameter. In terms of its tail properties, with probability at least $1 - \gamma$, $|x| \leq \lambda \log\left(\frac{1}{\gamma}\right)$. Extending it to the multivariate case, $Lap(\lambda)^p$ represents a distribution over vectors of dimension p where each entry of the vector is drawn from $Lap(\lambda)$. For any vector $x \sim Lap(\lambda)^p$, by union bound, with probability $1 - \gamma$, $\|x\|_1 \leq \lambda p \log\left(\frac{p}{\gamma}\right)$.

Normal (Gaussian) distribution: The density function for symmetric mean zero Gaussian distribution in p-dimensions is denoted by $\mathcal{N}(0, \mathbb{I}_p\sigma^2)$, where the density at any vector $x \in \mathbb{R}^p$ is given by $\frac{1}{\sqrt{(2\pi)^p}} e^{-\frac{\|x\|_2^2}{2\sigma^2}}$. In terms of the tail bound, with probability at least $1-\gamma$, $\|x\|_2 \leq \sqrt{2p\sigma^2 \log\left(\frac{1}{\gamma}\right)}$. This bound follows from a result by [Dasgupta and Schulman, 2007]. In one dimension, a similar bound holds via the classic Mill's inequality.

Gamma distribution The density function for mean zero Gamma distribution at any point $x \in \mathbb{R}^p$ is given by $\frac{1}{\Gamma(p)\sigma^p} \|x\|_2^{p-1} e^{-\frac{\|x\|_2}{\sigma}}$. [Chaudhuri et al., 2011] provided a tail bound for the Gamma distribution, that is, with probability at least $1-\gamma$, $\|x\|_2 \leq p\sigma \log\left(\frac{p}{\gamma}\right)$.

2.3 Convex functions

In this dissertation all our results are with respect to convex functions. A function $f : \mathbb{R}^p \to \mathbb{R}$ is *convex* if for all $x, y \in \mathbb{R}^p$ and $0 \leq \alpha < 1$,

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y).$$

A function f is Ψ-strongly convex if for all $x, y \in \mathbb{R}^p$ and $0 \leq \alpha < 1$,

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y) - \frac{\Psi\alpha(1-\alpha)}{2}\|x-y\|_2^2,$$

i.e., the function f is bounded from below by a quadratic. A nice property of strongly convex function is that it has a unique minimum, and the minimum eigenvalue of its hessian is always greater than Ψ. We use this property of strongly convex function heavily in both our privacy and utility analysis of different algorithms. Notice that for a function f to be strongly convex, it need not be differentiable. When f is non-differentiable, the eigenvalue bound of the hessian refers to the fact that there exists a (twice-continuously) differentiable quadratic function which bounds f from below and has the minimum eigenvalue of its hessian at least Ψ.

Chapter 3

Differentially Private Convex Optimization

3.1 Introduction

Convex optimization problems arise in a large variety of statistical and learning settings, notably in the minimization of convex empirical risk measures (commonly known as empirical risk minimization (ERM)). Regularized convex loss functions with convex constraints arise both directly (e.g., because of a convex log-likelihood function) or as proxies for more complicated (non-convex) constraints such as sparsity or low matrix rank.

Generalization error for such problems is often parameterized by n (number of records in the data) and p (dimensionality). In the classical *low-dimensional* setting (or *small p, large n* setting), p grows slower than n. In many of these problems, the underlying data set is sensitive, *e.g.,* genomic data, financial data, user transaction data. Although there is substantial social benefit to publishing the results of an analysis over such data, there is a significant risk of inadvertently leaking information about the entries in the data set. The question we want to address in this chapter is "*how can we effectively solve convex optimization problems for empirical risk minimization without inadvertently leaking sensitive information?*"

We design algorithms that satisfy *differential privacy* [Dwork et al., 2006b; Dwork, 2006]. Intuitively, differential privacy requires that data sets differing in only one entry induce similar distributions on the output of a (randomized) algorithm. This implies that an attacker will draw essentially the same conclusions about an individual whether or not that individual's data was used – even if many records are known to the attacker [Dwork, 2006; Ganta et al., 2008; Kifer and Machanavajjhala, 2011].

The goal of this chapter is to design algorithms in the *low-dimensional setting* for a larger class of convex optimization problems than in prior work and to provide the option of improving utility at the cost of a slight relaxation in privacy (i.e., having a non-zero δ in the (ϵ, δ)-differential privacy guarantee).

In Chapter 5, we take different approach of online learning for this same problem of

low-dimensional convex optimization. In online leatning the entries in the data set are assumed to arrive online as opposed to the classical offline setting (which we consider in this chapter) where the complete data set is available at once. We defer the comparison to the online learning approach till Chapter 5.

3.1.1 Problem Setting

Given a data set $\mathcal{D} = (d_1, ..., d_n)$ of n individuals, where each observation d_i lies in a fixed domain $\mathcal{T}^*$, consider the following p-dimensional convex program:

$$\hat{\theta} \in \arg\min_{\theta \in \mathcal{C}} \tfrac{1}{n}(\textstyle\sum_{i=1}^{n} \ell(\theta; d_i) + r(\theta)), \tag{3.1}$$

where $\ell(\theta; d_i)$ is a real-valued function that is convex in the first parameter $\theta \in \mathbb{R}^p$ for every $d \in \mathcal{T}^*$, the *regularizer* r is an arbitrary convex function and the *constraint* $\mathcal{C} \subseteq \mathbb{R}^p$ is a closed convex set.

This type of program captures a variety of empirical risk minimization (ERM) problems. For example, when $r = 0$, it can describe the MLE's (*maximum likelihood estimator*) for linear regression (where $\ell(\theta; d) = (y - \langle x, \theta \rangle)^2$ and $d = (x, y)$)) and logistic regression (where $\ell(\theta; d) = \log(1 + \exp(y\langle x, \theta \rangle))$). Another setting of r is $r(\theta) = \frac{\Delta}{2}\|\theta\|_2^2$. In the context of linear regression, this setting of r is commonly referred to as *ridge regression*. In our treatment of linear regression in the *small p, large n* setting, we will mainly focus on ridge regression. All the results in this chapter will be in the *low-dimensional* (or *small p, large n*) setting.

3.1.2 Notation

Let $\mathcal{D} = \langle d_1, \cdots, d_n \rangle$ be a data set drawn from domain $\mathcal{T}^*$. Let $\hat{\mathcal{L}}$ be an empirical loss defined as $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta; d_i)$, where $\ell(\theta; d_i)$ is a positive real valued convex function (in the first parameter $\theta \in \mathbb{R}^p$). Let $\mathcal{P}$ be a distribution over the domain $\mathcal{T}^*$. We define the stochastic loss for a parameter vector θ over the distribution $\mathcal{P}$ as follows: $\bar{\mathcal{L}}(\theta; \mathcal{P}) = \mathbb{E}_{d \sim \mathcal{P}}[\ell(\theta; d)]$. Now let $r : \mathbb{R}^p \to \mathbb{R}^+$ be a convex regularizer. We define the empirical objective function as $\hat{J}(\theta; \mathcal{D}) = \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{1}{n}r(\theta)$. Similarly, we define the stochastic objective function as $\bar{J}(\theta; \mathcal{P}) = \bar{\mathcal{L}}(\theta; \mathcal{P}) + \frac{1}{n}r(\theta)$. In the objective perturbation algorithm (see Sections 3.2.3 and 3.5.1) we add a "noisy" term $\frac{b^T\theta}{n}$ (where b is a noise vector drawn from some appropriate distribution) and an L_2 penalty $\frac{\Delta}{2n}\|\theta\|_2^2$ to the objective function $\hat{J}(\theta; \mathcal{D})$ in order to guarantee differential privacy. We denote such an objective function as $J^{\text{priv}}(\theta, b; \mathcal{D}) = \hat{J}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{1}{n}b^T\theta$. Since the term $\frac{\Delta}{2n}\|\theta\|_2^2$ becomes useful in our utility analysis too, we define $J^{\#}(\theta; \mathcal{D}) = \hat{J}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2$ to segregate the noise term. See Tables 3.1 and 3.2 for a summary.

Organization of the chapter: In Section 3.2 we review some of the prior works in differentially private empirical risk minimization. In Section 3.3 we provide an overview of our technical contributions in this chapter. In Section 3.4 we provide a generic tool (called *limit theorem for differential privacy*) for proving differential privacy properties

Description	Objective Function	Minimizer
Empirical loss + regularizer $\frac{1}{n}r$	$\hat{J}(\theta;\mathcal{D})$	$\hat{\theta}$
Expected empirical loss + regularizer $\frac{1}{n}r$	$\bar{J}(\theta;\mathcal{P})$	$\bar{\theta}$
$\hat{J}(\theta;\mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2$	$J^{\#}(\theta;\mathcal{D})$	$\theta^{\#}$
Private objective function ($J^{\#}(\theta;\mathcal{D}) + \frac{b^T\theta}{n}$)	$J^{\text{priv}}(\theta, b;\mathcal{D})$	θ^{priv}
$\bar{J}(\theta;\mathcal{P}) + \frac{\Delta}{2n}\|\theta\|_2^2$	$J^{\dagger}(\theta;\mathcal{P})$	$\theta^{\dagger}$

Table 3.1. Various objective functions and their corresponding minimizers over $\mathcal{C}$

Empirical objective functions	Stochastic (expected) objective functions
$\overbrace{\underbrace{\hat{\mathcal{L}}(\theta;\mathcal{D}) + \frac{1}{n}r(\theta)}_{\hat{J}(\theta;\mathcal{D})} + \frac{\Delta}{2n}\|\theta\|_2^2}^{J^{\#}(\theta;\mathcal{D})} + \frac{b^T\theta}{n}$ $\underbrace{\qquad\qquad\qquad\qquad}_{J^{\text{priv}}(\theta;\mathcal{D})}$	$\overbrace{\underbrace{\bar{\mathcal{L}}(\theta;\mathcal{P}) + \frac{1}{n}r(\theta)}_{\bar{J}(\theta;\mathcal{P})} + \frac{\Delta}{2n}\|\theta\|_2^2}^{J^{\dagger}(\theta;\mathcal{P})}$

Table 3.2. Schematic representation of various objective functions

of various algorithms. In Section 3.5 we apply the limit theorem in the context of convex optimization for ERM and show various improvements over the existing work.

3.2 Summary of Previous Work

The convex ERM setting considered here was explicitly studied by [Chaudhuri et al., 2011; Rubinstein et al., 2009], though variants and special cases had been considered previously. They considered two basic techniques: *output perturbation* (studied by both papers), where one releases the output $\hat{\theta}$ with additive noise, and *objective perturbation* (introduced by [Chaudhuri et al., 2011] and further studied by [Dwork et al., 2009]), where one releases the (exact) minimizer of a perturbed version of the objective function.

There also exist other techniques for specific convex optimization problems such as order statistics [Nissim et al., 2007; Dwork and Lei, 2009] and linear regression [Dwork and Lei, 2009]. The *sample-and-aggregate* framework [Nissim et al., 2007] is a generic technique for designing private algorithms, which can be instantiated in many different ways. [Smith, 2011] applied it to a class of statistical problems that includes low-dimensional ERM.

The existing analysis of output perturbation requires fewer assumptions than that of objective perturbation. Under the minimal set of assumptions that allow both techniques to apply, the worst-case theoretical guarantees on the two techniques' performance are very similar [Chaudhuri et al., 2011], and are better than the guarantees one gets for the techniques of [Dwork and Lei, 2009; Smith, 2011]. However, in experiments objective perturbation performed much better than objective perturbation. This phenomenon was partly explained by [Dwork et al., 2009], who showed that in logistic regression, objective perturbation distorts the minimizer much less than output perturbation on "nice" data.

3.2.1 Sample and Aggregate Framework

Sample and aggregate framework [Nissim et al., 2007; Smith, 2011] is a generic technique for converting an arbitrary algorithms with real valued outputs into its differentially private variant. The main idea in sample and aggregate framework (Figure 3.1) is the following. Let $\mathcal{T}^*$ be a domain of data sets and let $f : \mathcal{T}^* \to \mathbb{R}^p$ be a function to be evaluated on a data set $\mathcal{D} \in \mathcal{T}^*$ (of size n). First, randomly partition the data set $\mathcal{D}$ in to k data sets $\mathcal{D}_1, \cdots, \mathcal{D}_k$ of size n/k, where k is a parameter selected based on the exact problem. (See [Smith, 2011] for a generic setting of k.) Second, the function f is evaluated on all the data sets $\mathcal{D}_1, \cdots, \mathcal{D}_k$ to produce a vector $\langle f(\mathcal{D}_1), \cdots, f(\mathcal{D}_k)\rangle$. Finally, the output is produced as $A = \frac{1}{k}\sum_i f(\mathcal{D}_i) + Lap\left(\frac{\zeta p}{k\epsilon}\right)$, where ζ is a bound on the maximum width of the range of values each coordinate of the output of f is allowed to take and $Lap(\sigma)$ is a random variable sampled from the standard Laplace distribution. The aggregation function A using averaging was proposed by [Smith, 2011]. The initial sample and aggregate framework proposed in [Nissim et al., 2007] used a more complicated aggregation function.

In terms of privacy, the algorithm mentioned above is ϵ-differentially private. In order to provide good utility guarantees for the output, [Smith, 2011] stated a technique for effectively choosing the bound ζ using an algorithm called *widened winsorized mean*. In the context of empirical risk minimization (ERM) for MLE estimation, [Smith, 2011] showed that if the data points $(d_1, \cdots, d_n)$ in (3.1) are drawn i.i.d., then the generalization error for the differentially private estimator θ^{priv} will be at most $(1 + o(1))$-times the generalization error for the non-private estimator obtained via ERM on the complete data set $\mathcal{D}$ as long as the dimensionality p is $o(n^{1/6})$. In our results we will provide utility guarantees with sharper dependence on the dimensionality p.

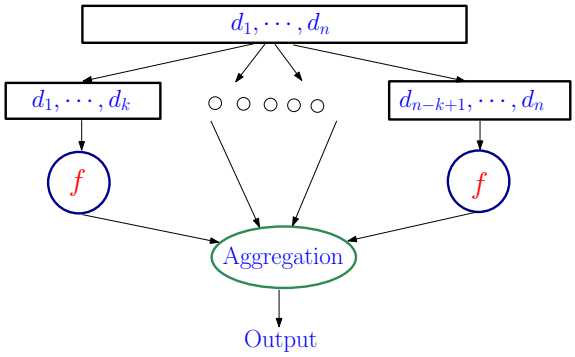

Figure 3.1. Sample and Aggregate Framework

3.2.2 Output Perturbation

The main idea in output perturbation is to first compute the minimizer $\hat{\theta}$ (see (3.1)) and add noise scaled according to the global sensitivity of $\hat{\theta}$ (see (3.1)). [Chaudhuri et al., 2011; Rubinstein et al., 2009] applied this approach to empirical risk minimization and analyzed the utility guarantees. In this section we summarize the output perturbation algorithm and state the privacy and utility guarantees.

Let $\mathcal{T}^*$ be a domain of data sets and let $f : \mathcal{T}^* \to \mathbb{R}^p$ be a function to be evaluated on a given data set $\mathcal{D} \in \mathcal{T}^*$. Let us define the term *global sensitivity* of the function f to be the quantity in (3.2). Here the operator Δ refers to the symmetric difference and $\|\cdot\|_q$ refers to the L_q-norm for a specific q.

$$\boldsymbol{GS}(f) = \max_{\mathcal{D},\mathcal{D}',|\mathcal{D}\Delta\mathcal{D}'|=1} \|f(\mathcal{D}) - f(\mathcal{D}')\|_q \tag{3.2}$$

Let $N \in \mathbb{R}^p$ be a random variable sampled from the noise distribution whose density is proportional to $e^{-\frac{\epsilon\|N\|_q}{2\boldsymbol{GS}(f)}}$. [Dwork et al., 2006b] in their *sensitivity method* showed that for a given data set $\mathcal{D}$, the algorithm which outputs $f(\mathcal{D}) + N$ is ϵ-differentially private. [Chaudhuri et al., 2011; Rubinstein et al., 2009] independently used the sensitivity method to output a differentially private version of $\hat{\theta}$, which is the minimizer in (3.1).

In their work, [Chaudhuri et al., 2011; Rubinstein et al., 2009] first compute the global sensitivity of the function $\hat{\theta}(\cdot)$ (where $\hat{\theta}(\mathcal{D})$ maps a data set $\mathcal{D}$ to the corresponding minimizer $\hat{\theta}$), and then adds noise based on the global sensitivity to produce $\theta^{\text{priv}}(\mathcal{D})$. [Chaudhuri et al., 2011] bounds the global sensitivity in the L_2-metric and [Rubinstein et al., 2009] bounds it in the L_1-metric. Both [Chaudhuri et al., 2011] and [Rubinstein et al., 2009] analyzes the ridge regression setting, i.e., when $r(\theta) = \frac{\Delta}{2}\|\theta\|_2^2$ in (3.1) with $\Delta > 0$. In order to bound the global sensitivity, the loss function $\ell(\theta; d)$ in (3.1) is assumed to be ζ-Lipschitz in the first parameter in the L_2 metric, i.e., for any $\theta_1, \theta_2 \in \mathcal{C}$, $|\ell(\theta_1; d) - \ell(\theta_2; d)| \leq \zeta\|\theta_1 - \theta_2\|_2$. In the following we provide a simpler analysis (compared to both [Chaudhuri et al., 2011; Rubinstein et al., 2009]) for computing the global sensitivity in the L_2-metric. Moreover, this analysis is strictly better than [Chaudhuri et al., 2011], since it does not require the loss function ℓ to be differentiable.

Lemma 3.1 (Global sensitivity of $\hat{\theta}$ in L_2-metric)**.** *For any pair of data sets $\mathcal{D}, \mathcal{D}' \in \mathcal{T}^*$ with $|\mathcal{D}\Delta\mathcal{D}'| = 1$, we have $\|\hat{\theta}(\mathcal{D}) - \hat{\theta}(\mathcal{D}')\|_2 \leq \frac{2\zeta}{\Delta}$, where n is the size of the data set $\mathcal{D}$.*

Proof. Without loss of generality assume that $\mathcal{D}'$ has one entry more than $\mathcal{D}$. Call this entry d_{n+1}. Recall that

$$\hat{\theta}(\mathcal{D}) = \arg\min_{\mathcal{C}} \underbrace{\sum_{i=1}^{n} \ell(\theta; d_i) + \frac{\Delta}{2}\|\theta\|_2^2}_{\hat{J}(\theta;\mathcal{D})}$$

and

$$\hat{\theta}(\mathcal{D}') = \arg\min_{\mathcal{C}} \hat{\mathcal{L}}(\theta; \mathcal{D}) + \ell(\theta; d_{n+1})$$

By strong convexity property we have,

$$\begin{aligned}
&\hat{J}(\hat{\theta}(\mathcal{D}'); \mathcal{D}) \geq \hat{J}(\hat{\theta}(\mathcal{D}); \mathcal{D}) + \frac{\Delta}{2}\|\hat{\theta}(\mathcal{D}') - \hat{\theta}(\mathcal{D})\|_2^2 \\
&\Leftrightarrow \hat{J}(\hat{\theta}(\mathcal{D}'); \mathcal{D}) + \ell(\hat{\theta}(\mathcal{D}'); d_{n+1}) + \ell(\hat{\theta}(\mathcal{D}); d_{n+1}) \\
&\qquad \geq \hat{J}(\hat{\theta}(\mathcal{D}); \mathcal{D}) + \ell(\hat{\theta}(\mathcal{D}'); d_{n+1}) + \ell(\hat{\theta}(\mathcal{D}); d_{n+1}) + \frac{\Delta}{2}\|\hat{\theta}(\mathcal{D}') - \hat{\theta}(\mathcal{D})\|_2^2 \\
&\Leftrightarrow \hat{J}(\hat{\theta}(\mathcal{D}'); \mathcal{D}') - \hat{J}(\hat{\theta}(\mathcal{D}); \mathcal{D}') + \ell(\hat{\theta}(\mathcal{D}); d_{n+1}) - \ell(\hat{\theta}(\mathcal{D}'); d_{n+1})
\end{aligned}$$

$$\geq \frac{\Delta}{2}\|\hat{\theta}(\mathcal{D}') - \hat{\theta}(\mathcal{D})\|_2^2 \tag{3.3}$$

Since $\hat{\theta}(\mathcal{D}')$ is the minimizer of $\hat{J}(\theta; \mathcal{D}')$, therefore $\hat{J}(\hat{\theta}(\mathcal{D}'); \mathcal{D}') - \hat{J}(\hat{\theta}(\mathcal{D}); \mathcal{D}') \leq 0$. Additionally, by Lipschitz property of ℓ we have, $\ell(\hat{\theta}(\mathcal{D}); d_{n+1}) - \ell(\hat{\theta}(\mathcal{D}'); d_{n+1}) \leq \zeta\|\hat{\theta}(\mathcal{D}) - \hat{\theta}(\mathcal{D}')\|_2$. Plugging in these bounds in (3.3), it follows that $\|\hat{\theta}(\mathcal{D}) - \hat{\theta}(\mathcal{D}')\|_2 \leq \frac{2\zeta}{\Delta}$. The L_1-sensitivity bound follows from the fact that L_1-norm is at most $\sqrt{p}$-times the L_2 norm. □

In terms of utility guarantee, from the analysis of [Chaudhuri et al., 2011] it can be concluded that if the loss function $\ell(\cdot;\cdot)$ is continuously differentiable with respect to the first parameter and has c-Lipschitz continuous gradient (i.e., for any two $\theta_1, \theta_2 \in \mathcal{C}$, $\|\bigtriangledown\ell(\theta_1; d) - \bigtriangledown\ell(\theta_2; d)\|_2 \leq \lambda\|\theta_1 - \theta_2\|_2$, then under suitable choice of the parameter Δ and assuming $\lambda > 1$, the *excess empirical risk* due to privacy scales as $O\left(\frac{\sqrt{\lambda}\zeta\|\theta^*\|_2^{4/3} p \log p}{\epsilon n^{2/3}}\right)$, where $\theta^* = \min_{\theta\in\mathcal{C}} \sum_{i=1}^n \ell(\theta; d_i)$. Here by excess empirical risk we mean the following:

$$\mathbb{E}_N\left[\frac{1}{n}\left(\sum_{i=1}^n \ell(\theta^{\text{priv}}(\mathcal{D}); d_i) - \min_{\theta\in\mathcal{C}} \sum_{i=1}^n \ell(\theta; d_i)\right)\right] \tag{3.4}$$

The bound of obtained by [Rubinstein et al., 2009] is similar but has a worse dependence on p.

Notice that the error bound obtained above is for the best possible choice of the parameter Δ. Next we discuss another algorithm (called *objective perturbation*[Chaudhuri et al., 2011]) which under the best possible choice of the regularization parameter Δ, has strictly better utility guarantee compared to output perturbation. However under similar settings that allow both these algorithms to work, the worst case theoretical error guarantees of output and objective perturbation are similar [Chaudhuri et al., 2011].

3.2.3 Objective Perturbation

In the previous section we saw an algorithm for private ERM, where first the minimizer $\hat{\theta}$ of the ERM is found via minimizing (3.1) and then $\hat{\theta}$ is appropriately perturbed to guarantee differential privacy. In this section we will discuss an algorithm where instead of perturbing the minimizer, the objective function in (3.1) is perturbed by a random linear term. This algorithm was first proposed by [Chaudhuri et al., 2011] and was found to outperform output perturbation in the experiments they conducted. In this section we summarize the objective perturbation algorithm of [Chaudhuri et al., 2011] and briefly state their utility and privacy guarantees.

Assume that the loss function $\ell(\cdot;\cdot)$ in (3.1) is twice-continuously differentiable and the regularizer $r(\theta) = \frac{\Delta}{2}\|\theta\|_2^2$. Also assume that the constraint set $\mathcal{C}$ is the complete real space, i.e., $\mathcal{C} = \mathbb{R}^p$. Let ζ be the upper bound on $\| \bigtriangledown \ell(\theta; d)\|_2$ and λ be the bound on the maximum eigenvalue of $\bigtriangledown^2\ell(\theta; d)$ for all $\theta \in \mathbb{R}^p$ and for all data points d. Under these assumptions [Chaudhuri et al., 2011] showed that the following algorithm (3.5) is ϵ-differentially private when the vector b is drawn from the Gamma distribution with

kernel $e^{-\frac{\epsilon\|b\|_2}{2\zeta}}$ and $\Delta \geq \frac{2\lambda}{\epsilon}$.

$$\theta^{\text{priv}} = \arg\min_{\theta\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n} \ell(\theta; d_i) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{\langle b, \theta\rangle}{n} \tag{3.5}$$

In terms of utility, [Chaudhuri et al., 2011] showed that under suitable choice of parameter Δ, the excess empirical risk due to privacy scales as $O\left(\frac{\zeta\|\theta^*\|_2 p \log p}{\epsilon n}\right)$, where $\theta^* = \min_{\theta\in\mathcal{C}} \sum_{i=1}^{n} \ell(\theta; d_i)$. For the definition of excess empirical risk see (3.4) in Section 3.2.2. Notice that the error bound obtained here is better than the one obtained for output perturbation in Section 3.2.2. However, the current algorithm is applicable only when the objective function is twice continuously differentiable and also the current bound is for the best possible choice of Δ. [Chaudhuri et al., 2011] reported that under similar settings which allow both output perturbation and objective perturbation to apply, the worst case theoretical guarantees of both these algorithms are similar. But in the experiments conducted by [Chaudhuri et al., 2011], objective perturbation always outperformed output perturbation. In our work, along with significantly extending the objective perturbation algorithm, we also provide a tighter utility analysis for objective perturbation (compared to [Chaudhuri et al., 2011]) for data sets satisfying some "niceness" properties. This tighter analysis provides a justification for the improved experimental behavior of objective perturbation.

3.3 Overview of Our Contribution

Our main contribution in this work is improving the objective perturbation technique (Section 3.2.3) to handle a much larger class of problems and provide tighter utility guarantees.

With the objective perturbation technique, instead of minimizing the empirical loss $\hat{J}(\theta; \mathcal{D}) = \frac{1}{n}(\sum_i \ell(\theta; d_i) + r(\theta))$, one considers a linear perturbation $J^{\text{priv}}(\theta; \mathcal{D}) = \hat{J}(\theta; \mathcal{D}) + \langle b, \theta\rangle$, where b is a random vector drawn according to a gamma distribution. The output of the algorithm is the minimizer of $J^{\text{priv}}(\cdot; \mathcal{D})$. We improve the treatment of [Chaudhuri et al., 2011] in several respects:

3.3.1 Generalized Privacy Analysis and a Limit Theorem for Differential Privacy

We show that objective perturbation (with either Gaussian or gamma perturbation) continues to be private even when the convex regularizer r is non-differentiable and when the parameter vector θ is constrained to a closed convex set $\mathcal{C}$. As mentioned above, the privacy proof of [Chaudhuri et al., 2011] required that r be differentiable and θ be unconstrained.

Our analysis greatly extends the range of problems to which objective perturbation applies. For example, it allows one to use objective perturbation for convex programs like the Lasso (where the regularizer r is the L_1 norm) and nuclear norm regularized minimization [Negahban et al., 2010], which was earlier not possible. The extension is

also critical for applying the objective perturbation technique to linear regression.

The main tool we use in the above analysis is a *limit theorem* for differential privacy. The theorem states that if a sequence of (ϵ, δ)-differentially private algorithms $\mathcal{A}_1, \mathcal{A}_2, \ldots$ converges in a weak sense, then the limiting algorithm $\mathcal{A} = \lim_{i\to\infty} \mathcal{A}_i$ is also (ϵ, δ)-differentially private. Note that the probabilistic behavior of $\mathcal{A}$ can be very different from any of the $\mathcal{A}_i$ (see Example 3.3). We feel this tool is likely to have other applications.

The idea behind our generalized analysis of objective perturbation is to approximate the constrained, nondifferentiable problem in (3.1) with a sequence of unconstrained, differentiable problems, and apply our limit theorem to the resulting sequence of algorithms. The difficulty is in ensuring that the resulting problems are all convex (so that the previous analysis applies) and converge in an appropriate sense to the original problem.

3.3.2 More Accurate Objective Perturbation

We show that drawing the perturbation b from a Gaussian (instead of gamma) distribution, leads to a $\tilde{\Omega}(\sqrt{p})$ improvement in the utility guarantees of the objective algorithm, at the cost of relaxing the privacy guarantee from $(\epsilon, 0)$- to (ϵ, δ)-differential privacy for negligible δ. When $\delta < 1/n^2$, the relaxed guarantee has very similar semantics to the original [Ganta et al., 2008]. This result parallels a similar improvement that is possible for output perturbation (see, *e.g.,* [Dwork et al., 2006a]), though the privacy and utility proofs are quite different.

3.3.3 Data-dependent Utility Analysis

Finally, we provide an improved, data-dependent utility analysis. Our approach is inspired by the analysis of [Dwork et al., 2009], which was specific to logistic regression. We show that for "nice" data, namely, data sets for which the loss function is *strongly* convex in a neighborhood of its minimizer, objective perturbation has much better error guarantees than in the worst case (roughly, a factor of $\sqrt{p}$ lower or a typical setting of parameters). The assumption of strong convexity is common in the optimization literature (*e.g.,* [Nocedal and Wright, 2000; Negahban et al., 2010]).

3.4 Limit Theorem for Differentially Private Algorithms

Establishing that an algorithm $\mathcal{A}$ satisfies differential privacy is often a difficult task. In this section we present a new proof technique for deriving the privacy properties of $\mathcal{A}$ from a sequence of differentially private algorithms $\mathcal{A}_i$. The power of this technique is that we only require a very weak form of convergence; in fact, the limiting probabilistic behavior of the A_i can be quite different from the behavior of $\mathcal{A}$. Our results are summarized in the following theorem. The proof of this theorem is given in Section 3.4.1.

Theorem 3.2 (Successive Approximation)**.** *Let b be a $\mathbb{R}^p$-valued random variable. Let $\mathcal{A}$ be a randomized algorithm induced by the random variable b and some deterministic function ϕ – that is, $\mathcal{A}(\mathcal{D}) \equiv \phi(\mathcal{D}, b)$. Let $\mathcal{A}_1, \mathcal{A}_2, \ldots$ be a sequence of randomized algorithms, where each $\mathcal{A}_i$ is induced by b and some deterministic function ϕ^i (i.e. $\mathcal{A}_i(\mathcal{D}) \equiv \phi^i(\mathcal{D}, b)$).*

If $\mathcal{A}_1, \mathcal{A}_2, \ldots$ are all (ϵ, δ)-differentially private and $\lim_{i\to\infty} \phi^i(\mathcal{D}, b) = \phi(\mathcal{D}, b)$ (i.e. pointwise convergence for all $\mathcal{D}$ and realized values of b), then $\mathcal{A}$ is also (ϵ, δ)-differentially private.

It is important to note that differential privacy is a condition on $\Pr[\phi^i(\mathcal{D}, b) \in \mathcal{O}]$ (which is the same as $P(\mathcal{A}_i(\mathcal{D}) \in \mathcal{O})$) yet the pointwise convergence $\lim_{i\to\infty} \phi^i(\mathcal{D}, b) \to \phi(\mathcal{D}, b)$ required by Theorem 3.2 is too weak to guarantee that $\Pr[\phi^i(\mathcal{D}, b) \in \mathcal{O}] \to \Pr[\phi(\mathcal{D}, b) \in \mathcal{O}]$. In fact, the limiting probabilistic behavior (if it exists) of $\phi^i(\mathcal{D}, b)$ can be quite different from the probabilistic behavior of $\phi(\mathcal{D}, b)$. Nevertheless, Theorem 3.2 establishes that $\mathcal{A}$ still inherits differential privacy properties from the $\mathcal{A}_i$'s. Consider the following example:

Example 3.3. *Let $\theta \in \mathbb{R}^p$ be a parameter vector and let $\hat{\mathcal{L}}(\theta; \mathcal{D})$ be a strongly convex, twice continuously differentiable loss function. Let $\phi(\mathcal{D}, b) \equiv \operatorname{argmin}_\theta \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{1}{n}(b^T\theta + \|\theta\|_1)$, which is an L_1-regularized minimization problem (with random perturbation $b^T\theta$). We can approximate it (see Section 3.5.2.2) with a sequence $\phi^i(\mathcal{D}, b) \equiv \operatorname{argmin}_\theta \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{1}{n}(b^T\theta + r_i(\theta))$ where r_i is an infinitely differentiable regularizer. If b has a continuous probability distribution, then for each fixed $\mathcal{D}$ the distribution of $\phi^i(\mathcal{D}, b)$ has a density ([Chaudhuri et al., 2011]) but $\phi(\mathcal{D}, b)$ does not. In fact, the subdifferentials of the L_1 regularizer ensure that for each $\mathcal{D}$, $\phi(\mathcal{D}, b)$ can take values in a lower-dimensional submanifold of $\mathbb{R}^p$ with positive probability (which is not possible for the $\phi^i(\mathcal{D}, b)$ because of their densities).*

3.4.1 Differential Privacy via Successive Approximation

Theorem 3.2 directly follows from the following lemma. In the following, we use the notation $\phi_\mathcal{D}$ and $\phi^i_\mathcal{D}$ in place of $\phi(\mathcal{D}, \cdot)$ and $\phi^i(\mathcal{D}, \cdot)$, respectively.

Lemma 3.4 (Weak Convergence Lemma)**.** *Let $\mathcal{D}$ and $\mathcal{D}'$ be two data sets. Let b be a random variable. Let $\phi^1_\mathcal{D}, \phi^2_\mathcal{D}, \ldots$ be a sequence of functions that converge pointwise to some function $\phi_\mathcal{D}$ (i.e., $\lim_{i\to\infty} \phi^i_\mathcal{D}(b) = \phi_\mathcal{D}(b)$ for all values of b). Similarly, let $\phi^1_{\mathcal{D}'}, \phi^2_{\mathcal{D}'}, \ldots$ be a sequence of functions that converge pointwise to some function $\phi_{\mathcal{D}'}$. Let $\mu^i_\mathcal{D}$ be the probability measure defined on any Borel set as $\mu^i_\mathcal{D}(E) \equiv \Pr[\phi^i_\mathcal{D}(b) \in E]$ and define $\mu^i_{\mathcal{D}'}$, $\mu_\mathcal{D}$, and $\mu_{\mathcal{D}'}$ similarly. For all $\epsilon, \delta \geq 0$, for all Borel sets E, if $\mu^i_\mathcal{D}(E) \leq e^\epsilon \mu^i_{\mathcal{D}'}(E) + \delta$ for all i, then $\mu_\mathcal{D}(E) \leq e^\epsilon \mu_{\mathcal{D}'}(E) + \delta$.*

Lemma 3.4 follows immediately from Claims 3.5, 3.6, and 3.7.

Claim 3.5. *Consider the σ-algebra of Borel subsets of $\mathbb{R}^p$. For any Borel set E, probability measure μ, and $\xi > 0$, there exists an open set A and a closed set B such that: $B \subseteq E \subseteq A$ and*

- $\mu(E) \leq \mu(A) \leq \mu(E) + \xi$
- $\mu(B) \leq \mu(E) \leq \mu(B) + \xi$

Proof. We first prove the first condition relating E and A for the cases when E is closed and then when E is a Borel set (the case when E is open is trivial). Then we prove the condition relating E and B by reducing it to previous results.

Part 1: Closed sets E.
Suppose E is a closed subset of $\mathbb{R}^p$. For each $i = 1, 2, \ldots$, define $A^{(i)} = \{y \ : \inf_{x \in E} \|x - y\|_2 < 1/i\}$. Each $A^{(i)}$ is open since it is the union of open balls of radius $1/i$ around each point of E. Clearly, $E \subseteq A^{(i)}$ and for all i and $A^{(1)} \supseteq A^{(2)} \supseteq \ldots$. Also $E = \bigcap_{i=1}^{\infty} A^{(i)}$ because if a point $x \notin E$ then, since E is closed, the distance between x and E is non-zero and so one of the $A^{(i)}$ does not contain x. The downward continuity property [Billingsley, 1995] of probability measures now ensures that $\lim_{i \to \infty} \mu(A^{(i)}) = \mu(E)$. Thus given $\xi > 0$, there exists an i such that $\mu(A^{(i)}) \leq \mu(E) + \xi$ and $\mu(A^{(i)}) \geq \mu(E)$ because $E \subseteq A^{(i)}$.

Part 2: Borel sets E. Consider the algebra $\mathcal{G}$ consisting of all subsets of $\mathbb{R}^p$ that are (1) open, or (2) closed, or (3) the intersection of an open and a closed set, or (4) the union of an open and closed set. Note that $\mathbb{R}^p \in \mathcal{G}$ and that $\mathcal{G}$ is closed under complementation, finite union, and finite intersection. Given the values of $\mu(C)$ for all $C \in \mathcal{G}$, we can define the outer measure [Billingsley, 1995] μ^* on <u>all</u> subsets $F \subseteq \mathbb{R}^p$ as follows:

$$\mu^*(F) = \inf_{\substack{\{C_1, C_2, \ldots\} \subseteq \mathcal{G} \\ F \subseteq \bigcup C_i}} \sum_i \mu(C_i)$$

where the infimum is taken over all finite and countable collections of sets from $\mathcal{G}$ whose union contains F. Caratheodory's Extension Theorem [Billingsley, 1995] guarantees that $\mu(E) = \mu^*(E)$ for all Borel sets E. Thus for any $\xi > 0$, there exists a finite or countable collection $C_1, C_2, \ldots$ of sets in $\mathcal{G}$ such that $E \subseteq \bigcup_i C_i$ and $\mu(E) \leq \sum_i \mu(C_i) \leq \mu(E) + \xi/2$.

We now replace the C_i with slightly bigger open sets A_i. If C_i is open, then set $A_i = C_i$. If C_i is closed, then use the previous result to find an open set $A_i \supset C_i$ such that $\mu(A_i) \leq \mu(C_i) + \xi/2^{i+1}$. If C_i is the intersection of an open set $\mathcal{O}$ and a closed set H, then replace H with an open set $H' \supset H$ such that $\mu(H') \leq \mu(H) + \xi/2^{i+1}$ and set $A_i = \mathcal{O} \cap H'$. Note that $C_i \subset A_i$ and $\mu(A_i) \leq \mu(C_i \cup (H' \setminus H)) \leq \mu(C_i) + \xi/2^{i+1}$. Finally, if C_i is the union of an open set $\mathcal{O}$ and a close set H, then replace H with an open set $H' \supset H$ such that $\mu(H') \leq \mu(H) + \xi/2^{i+1}$ and set $A_i = \mathcal{O} \cup H'$. Note that $C_i \subset A_i$ and $\mu(A_i) \leq \mu(C_i \cup (H' \setminus H)) \leq \mu(C_i) + \xi/2^{i+1}$.

Set $A = \bigcup_i A_i$. Note that A is open. Then, since $E \subseteq A$,

$$\begin{aligned} \mu(E) &\leq \mu(A) \leq \sum_i \mu(A_i) \leq \sum_i (\mu(C_i) + \xi 2^{-i-1}) \\ &\leq \xi/2 + \sum_i \mu(C_i) \leq \xi/2 + \mu(E) + \xi/2 \\ &= \mu(E) + \xi \end{aligned}$$

Part 3: Approximating E from below.
To prove the second part of the theorem, pick an $\xi > 0$ and choose an open set $A \supseteq E^c$ (the complement of E) such that $\mu(E^c) \leq \mu(A) \leq \mu(E^c) + \xi$. Set $B = A^c$. Then B is

closed, $B \subseteq E$, and

$$\begin{aligned}\mu(E^c) &\leq \mu(B^c) \leq \mu(E^c) + \xi \\ &\Rightarrow 1 - \mu(E) \leq 1 - \mu(B) \leq 1 - \mu(E) + \xi \\ &\Rightarrow \mu(B) \leq \mu(E) \leq \mu(B) + \xi\end{aligned}$$

□

The next result shows that pointwise convergence of the $\phi^i_{\mathcal{D}}$ allows us to upper bound $\Pr[\phi_{\mathcal{D}}(b) \in \mathcal{O}]$ when $\mathcal{O}$ is open and lower bound it when $\mathcal{O}$ is closed.

Claim 3.6. *Under the assumptions of Lemma 3.4, for every open set $A \subseteq \mathbb{R}^p$,*

$$\mu_{\mathcal{D}}(A) \leq \liminf_{i\to\infty} \mu^i_{\mathcal{D}}(A)\,.$$

For every closed set $B \subseteq \mathbb{R}^p$,

$$\mu_{\mathcal{D}}(B) \geq \limsup_{i\to\infty} \mu^i_{\mathcal{D}}(B)$$

Proof. For any set C, we use the notation $1_{\{\phi_{\mathcal{D}}(b)\in C\}}(b)$ to be the indicator function that is 1 when $\phi_{\mathcal{D}}(b) \in C$ and 0 otherwise, and similarly for $1_{\{\phi^i_{\mathcal{D}}(b)\in C\}}(b)$. Let A be any open set. For any b such that $\phi_{\mathcal{D}}(b) \in A$, there is a bounded open set $\mathcal{O}$ so that $\phi_{\mathcal{D}}(b) \in \mathcal{O} \subseteq A$. Since $\phi^i_{\mathcal{D}}(b)$ converges to $\phi_D(b)$, this means that eventually $\phi^i_{\mathcal{D}}(b) \in \mathcal{O}$ and so $\phi^i_{\mathcal{D}}(b) \in A$. This means that for any b such that $\phi_{\mathcal{D}}(b) \in A$, the indicators $1_{\{\phi^i_{\mathcal{D}}(b)\in A\}}(b)$ converge to $1_{\{\phi_{\mathcal{D}}(b)\in A\}}$ as $i \to \infty$. For b such that $\phi_{\mathcal{D}}(b) \notin A$, $1_{\{\phi^i_{\mathcal{D}}(b)\in A\}}(b) \geq 0 = 1_{\{\phi_{\mathcal{D}}(b)\in A\}}(b)$. Thus for all b, $\liminf_{i\to\infty} 1_{\{\phi^i_{\mathcal{D}}(b)\in A\}}(b) \geq 1_{\{\phi_{\mathcal{D}}(b)\in A\}}(b)$. By Fatou's Lemma [Billingsley, 1995],

$$\begin{aligned}\mu_{\mathcal{D}}(A) &= \int 1_{\{\phi_{\mathcal{D}}(b)\in A\}}(b)\; d\mu(b) \\ &\leq \int \liminf_{i\to\infty} 1_{\{\phi^i_{\mathcal{D}}(b)\in A\}}(b)\; d\mu(b) \\ &\leq \liminf_{i\to\infty} \int 1_{\{\phi^i_{\mathcal{D}}(b)\in A\}}(b)\; d\mu(b) \\ &= \liminf_{i\to\infty} \mu^i_{\mathcal{D}}(A)\end{aligned}$$

To show the second part, let B be a closed set. Consider its complement B^c, which is an open set. Using the previous result,

$$\begin{aligned}\mu_{\mathcal{D}}(B^c) &\leq \liminf_{i\to\infty} \mu^i_{\mathcal{D}}(B^c) \\ &\Rightarrow 1 - \mu_{\mathcal{D}}(B) \leq \liminf_{i\to\infty} (1 - \mu^i_{\mathcal{D}}(B)) \\ &\Rightarrow \mu_{\mathcal{D}}(B) \geq -\liminf_{i\to\infty} -\mu^i_{\mathcal{D}}(B) \\ &\Rightarrow \mu_{\mathcal{D}}(B) \geq \limsup_{i\to\infty} \mu^i_{\mathcal{D}}(B)\end{aligned}$$

□

The final result states that the upper bound and lower bound results of Claim 3.6 are all that we need.

Claim 3.7. *Let E be a Borel set and let $\mu_{\mathcal{D}}$, $\mu_{\mathcal{D}'}$, $\mu^i_{\mathcal{D}}$, and $\mu^i_{\mathcal{D}'}$ (for all i) be probability measures such that:*

1. *$\mu_{\mathcal{D}}(A) \leq \lim_{i\to\infty} \inf \mu^i_{\mathcal{D}}(A)$ for all open sets $A \subseteq \mathbb{R}^p$ and $\mu_{\mathcal{D}}(B) \geq \lim_{i\to\infty} \sup \mu^i_{\mathcal{D}}(B)$ for all closed sets $B \subseteq \mathbb{R}^p$.*
2. *$\mu_{\mathcal{D}'}(A) \leq \lim_{i\to\infty} \inf \mu^i_{\mathcal{D}'}(A)$ for all open sets $A \subseteq \mathbb{R}^p$ and $\mu_{\mathcal{D}'}(B) \geq \lim_{i\to\infty} \sup \mu^i_{\mathcal{D}'}(B)$ for all closed sets $B \subseteq \mathbb{R}^p$.*

For all $\epsilon, \delta \geq 0$, if $\mu^i_{\mathcal{D}}(E) \leq e^\epsilon \mu^i_{\mathcal{D}'}(E) + \delta$ for all i, then $\mu_{\mathcal{D}}(E) \leq e^\epsilon \mu_{\mathcal{D}'}(E) + \delta$.

Proof. **Part 1: Reduction to open sets.**
Let E be a Borel set and let $\mathcal{D}$ and $\mathcal{D}'$ be two data sets that differ by the addition or deletion of one tuple. Assume, by way of contradiction, that the (ϵ, δ)-differential privacy conditions do not hold so that $\mu_{\mathcal{D}}(E) \geq e^\epsilon \mu_{\mathcal{D}'}(E) + \delta + \alpha$ for some $\alpha > 0$. Using Claim 3.5, choose an open set $\mathcal{O} \supseteq E$ such that:

$$\mu_{\mathcal{D}'}(E) \quad \leq \quad \mu_{\mathcal{D}'}(\mathcal{O}) \leq \mu_{\mathcal{D}'}(E) + \frac{\alpha}{2e^\epsilon}$$

Therefore

$$\begin{aligned} \mu_{\mathcal{D}}(\mathcal{O}) &\geq \mu_{\mathcal{D}}(E) \geq e^\epsilon \mu_{D'}(E) + \delta + \alpha \\ &\geq e^\epsilon \left(\mu_{\mathcal{D}'}(\mathcal{O}) - \frac{\alpha}{2e^\epsilon}\right) + \delta + \alpha \\ &= e^\epsilon \mu_{D'}(\mathcal{O}) + \delta + \alpha/2 \end{aligned}$$

and so $\mathcal{O}$ also violates the (ϵ, δ)-differential privacy constraints. Therefore, without loss of generality, we can assume that E is actually an open set.

Part 2: Proof for open sets. Let E be an open set that violates the (ϵ, δ)-differential privacy conditions such that $\mu_{\mathcal{D}}(E) \geq e^\epsilon \mu_{\mathcal{D}'}(E) + \delta + \alpha$ for some $\alpha > 0$. We will approximate E from below using both open sets and closed sets as follows. First, note that $E \neq \emptyset$ because the set $\emptyset$ can never violate the differential privacy conditions. Consider the open sets A_i and closed set B_i defined as follows:

$$\begin{aligned} A_i &= \{\theta' \ : \inf_{\theta \in E^c} \|\theta - \theta'\|_2 < 1/i\} \\ B_i &= \{\theta' \ : \inf_{\theta \in E^c} \|\theta - \theta'\|_2 \leq 1/i\} \end{aligned}$$

Note that $B_i = \overline{A_i}$ (B_i is the closure of A_i) and E^c is a subset of A_i and B_i for all $i \geq 1$. Now define the open set $\mathcal{O}_i \equiv B_i^c$ and note that $\overline{\mathcal{O}_i} = A_i^c$ and that $\mathcal{O}_i$ and $\overline{\mathcal{O}_i}$ are subsets

of E. Finally, note that $\mathcal{O}_1 \subseteq \mathcal{O}_2 \subseteq \ldots$ and $\overline{\mathcal{O}_1} \subseteq \overline{\mathcal{O}_2} \subseteq \ldots$ and

$$\bigcup_{i=1}^{\infty} \mathcal{O}_i = E = \bigcup_{i=1}^{\infty} \overline{\mathcal{O}_i}$$

Now, by the upward continuity property of probability measures [Billingsley, 1995], there exists an i_0 such that for all $i \geq i_o$

$$\begin{aligned} \mu_{\mathcal{D}}(\mathcal{O}_i) &\leq \mu_{\mathcal{D}}(E) \leq \mu_{\mathcal{D}}(\mathcal{O}_i) + \frac{\alpha}{3} \\ \mu_{\mathcal{D}'}(\overline{\mathcal{O}_i}) &\leq \mu_{\mathcal{D}'}(E) \end{aligned}$$

Thus

$$\begin{aligned} \mu_{\mathcal{D}}(E) &\geq e^{\epsilon}\mu_{\mathcal{D}'}(E) + \delta + \alpha \\ &\Rightarrow \mu_{\mathcal{D}}(\mathcal{O}_i) + \frac{\alpha}{3} \geq e^{\epsilon}\mu_{\mathcal{D}'}(\overline{\mathcal{O}_i}) + \delta + \alpha \\ &\Rightarrow \mu_{\mathcal{D}}(\mathcal{O}_i) \geq e^{\epsilon}\mu_{\mathcal{D}'}(\overline{\mathcal{O}_i}) + \delta + \frac{2\alpha}{3} \end{aligned}$$

Then, using the lim inf conditions on open sets and lim sup conditions on closed sets,

$$\begin{aligned} \liminf_{j\to\infty} \mu^j_{\mathcal{D}}(\mathcal{O}_i) &\geq \mu_{\mathcal{D}}(\mathcal{O}_i) \\ &\geq e^{\epsilon}\mu_{\mathcal{D}'}(\overline{\mathcal{O}_i}) + \delta + \frac{2\alpha}{3} \\ &\geq \limsup_{j\to\infty} e^{\epsilon}\mu^j_{\mathcal{D}'}(\overline{\mathcal{O}_i}) + \delta + \frac{2\alpha}{3} \end{aligned}$$

Now, since $\mu^j_{\mathcal{D}}(\mathcal{O}_i) \leq \mu^j_{\mathcal{D}}(\overline{\mathcal{O}_i})$:

$$\liminf_{j\to\infty} \mu^j_{\mathcal{D}}(\overline{\mathcal{O}_i}) \geq \limsup_{j\to\infty} e^{\epsilon}\mu^j_{\mathcal{D}'}(\overline{\mathcal{O}_i}) + \delta + \frac{2\alpha}{3}$$

and so for some j

$$\mu^j_{\mathcal{D}}(\overline{\mathcal{O}_i}) \geq e^{\epsilon}\mu^j_{\mathcal{D}'}(\overline{\mathcal{O}_i}) + \delta + \frac{\alpha}{3}$$

However, this contradicts the fact that the pair of measures $\mu^j_{\mathcal{D}}$, $\mu^j_{\mathcal{D}'}$ satisfy the (ϵ, δ)-differential privacy conditions ($\mu^j_{\mathcal{D}}(\overline{\mathcal{O}_i}) \leq e^{\epsilon}\mu^j_{\mathcal{D}'}(\overline{\mathcal{O}_i}) + \delta$). Therefore E cannot violate the (ϵ, δ)-differential privacy conditions for the measures $\mu_{\mathcal{D}}$ and $\mu_{\mathcal{D}'}$. □

Algorithm 3.1 Generalized Objective Perturbation Mechanism (Obj-Pert)

Require: data set $\mathcal{D} = \{d_1, \ldots, d_n\}$, privacy parameters ϵ and δ ($\delta = 0$ for ϵ-differential privacy), convex regularizer r, a convex domain $\mathcal{C} \subseteq \mathbb{R}^p$, convex loss function $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta; d_i)$ with continuous Hessian, $\| \bigtriangledown \ell(\theta; d)\|_2 \leq \zeta$ (for all $d \in \mathcal{P}$ and $\theta \in \mathcal{C}$), and upper bound λ on the eigenvalues of $\bigtriangledown^2 \ell(\theta; d)$ (for all d and for all $\theta \in \mathcal{C}$).

1: Set $\Delta \geq \frac{2\lambda}{\epsilon}$.
2: **if** require ϵ-differential privacy **then**
3: sample $b \in \mathbb{R}^p$ from the Gamma distribution with density $\nu_1(b; \epsilon, \zeta) \propto e^{-\epsilon \frac{\|b\|_2}{2\zeta}}$
4: **else if** require (ϵ, δ)-differential privacy **then**
5: sample $b \in \mathbb{R}^p$ from $\nu_2(b; \epsilon, \delta, \zeta) = \mathcal{N}\left(0, \frac{\zeta^2(8 \log \frac{2}{\delta} + 4\epsilon)}{\epsilon^2} \mathbb{I}_{p \times p}\right)$.
6: **end if**
7: **return** $\theta^{\text{priv}} \equiv \arg\min_{\theta \in \mathcal{C}} \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{1}{n} r(\theta) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{b^T \theta}{n}$.

3.5 Application: Private Convex Optimization for ERM

3.5.1 Private Constrained Optimization for ERM

We use Theorem 3.2 to extend the applicability of the differentially private empirical risk minimization framework of [Chaudhuri et al., 2011] to allow hard convex constraints and non-differentiable regularizers. Consider the convex program: $\text{argmin}_{\theta \in \mathcal{C}} \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{1}{n} r(\theta)$, where $\mathcal{C} \subseteq \mathbb{R}^p$ is a closed convex set, $\mathcal{D} = \{d_1, \ldots, d_n\}$ is a data set, $\hat{\mathcal{L}}$ is a twice-continuously differentiable convex loss function of the form $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta; d_i)$ and r is any (possibly non-differentiable) convex regularizer. When the objective function is γ/n-strongly convex ($\gamma \geq 0$) for all data sets of size n, one adds a quadratic term $\frac{(\Delta - \gamma)^+}{2n}\|\theta\|_2^2$, where Δ depends on the largest possible eigenvalue of the Hessian of $\ell(\theta, d_i)$. This ensures that the objective function is Δ/n-strongly convex and reduces the influence of any single data point. For privacy, a random linear perturbation term $\frac{b^T\theta}{n}$ is then added to the objective function. The full mechanism is described in Algorithm 4.1. Note that to simplify the discussion, we can w.l.o.g. assume $\gamma = 0$ (i.e.,the initial objective function is not strongly convex).

Theorem 3.8 (Private Convex Optimization via Objective Perturbation)**.** *Let $\mathcal{C}$ be a closed convex subset of $\mathbb{R}^p$. Let $\mathcal{D} = \{d_1, \ldots, d_n\}$ be a data set, let $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta; d_i)$ be a convex loss function with continuous Hessian, let ζ be the upper bound on $\|\bigtriangledown \ell(\theta; d)\|_2$ and let λ be an upper bound on the eigenvalues of $\bigtriangledown^2 \ell(\theta; d)$ (for all d and for all $\theta \in \mathcal{C}$), and let r be a convex function. Assume that for all $\theta \in \mathcal{C}$ and for all d the rank of $\bigtriangledown^2 \ell(\theta; d)$ is at most one.*

Then Algorithm 4.1 is $(\epsilon, 0)$-differentially private when b has gamma density ν_1 and (ϵ, δ)-differentially private when b has Gaussian density ν_2.

We provide the detailed proof in the following section (Section 3.5.2). The main idea is to use Theorem 3.2 twice. We first consider unconstrained optimization and convolve the regularizer r with a sequence $K_1, K_2, \ldots$ of infinitely differentiable kernels. This

results in a sequence of smooth optimization problems that can be solved differentially privately by the results of [Chaudhuri et al., 2011]. We prove pointwise convergence of their differentially private solutions and then invoke Theorem 3.2. For constrained optimization, we replace the hard constraint $\theta \in \mathcal{C}$ with a sequence of soft constraints by adding penalties for $\theta \notin \mathcal{C}$ that depend on the distance from θ to $\mathcal{C}$. We again show pointwise convergence and invoke Theorem 3.2.

3.5.2 Proof of Theorem 4.1 (Private Convex Optimization via Objective Perturbation)

In order to prove Theorem 4.1, the starting point is Lemma 3.9 (Section 3.5.2.1) which proves differential privacy for the special cases of Algorithm 4.1 where the regularizer r is twice continuously differentiable and the convex set $\mathcal{C}$ over which we optimize is the entire real space $\mathbb{R}^p$. Afterwards, we will use our *successive approximation* technique to remove these assumptions one-by-one (Sections 3.5.2.2 and 3.5.2.3).

3.5.2.1 Private Smooth Unconstrained Optimization

Lemma 3.9 (Differentially Private Smooth Unconstrained Objective Perturbation)**.** *Under the conditions of Theorem 4.1, if we assume that the convex regularizer r is twice-continuously differentiable, the convex set $\mathcal{C}$ is the entire real space $\mathbb{R}^p$, then*

1. *[Chaudhuri et al., 2011] with Gamma density ν_1 in Algorithm 4.1 (Algorithm* Obj-Pert *) guarantees ϵ-differential privacy.*
2. *(This work) with Gaussian density ν_2 in Algorithm 4.1 (Algorithm* Obj-Pert *) guarantees (ϵ, δ)-differential privacy.*

Proof. The first part of Lemma 3.9 follows directly from [Chaudhuri et al., 2011] and hence omitted here. The proof of the second part of Lemma 3.9 is as follows.

If we want to prove that Algorithm 4.1 satisfies (ϵ, δ)-privacy, it suffices to show that for all $\alpha \in \mathbb{R}^p$ the following is true.

$$e^{-\epsilon}(\text{pdf}(\theta^{\text{priv}} = \alpha; \mathcal{D}') - \delta) \leq \text{pdf}(\theta^{\text{priv}} = \alpha; \mathcal{D}) \leq e^{\epsilon}\text{pdf}(\theta^{\text{priv}} = \alpha; \mathcal{D}') + \delta \tag{3.6}$$

First consider an $\alpha \in \mathbb{R}^p$. If we have $\theta^{\text{priv}} = \alpha$, then it means that $\alpha = \arg\min_{\theta \in \mathbb{R}^p} n\hat{\mathcal{L}}(\theta; \mathcal{D}) + r(\theta) + \frac{\Delta}{2}\|\theta\|_2^2 + b^T\theta$. Setting the gradient of the objective function to zero we get the following.

$$b(\alpha; \mathcal{D}) = -\left(n \bigtriangledown \hat{\mathcal{L}}(\alpha; \mathcal{D}) + \bigtriangledown r(\alpha) + \Delta\alpha\right) \tag{3.7}$$

We have $\frac{\text{pdf}_{\mathcal{D}}(\theta^{\text{priv}}=\alpha)}{\text{pdf}_{\mathcal{D}'}(\theta^{\text{priv}}=\alpha)} = \frac{\nu_2(b(\alpha;\mathcal{D});\epsilon,\delta,\zeta)}{\nu_2(b(\alpha;\mathcal{D}');\epsilon,\delta,\zeta)} \frac{|\det(\bigtriangledown b(\alpha;\mathcal{D}'))|}{|\det(\bigtriangledown b(\alpha;\mathcal{D}))|}$. We bound the ratios of the densities ν_2 and the determinants separately.

First, we show that for all $\alpha \in \mathbb{R}^p$, $e^{-\epsilon} \leq \frac{|\det(\bigtriangledown b(\alpha;\mathcal{D}'))|}{|\det(\bigtriangledown b(\alpha;\mathcal{D}))|} \leq e^{\epsilon}$. The following lemma would be helpful in bounding the ratio.

Lemma 3.10 ([Chaudhuri et al., 2011])**.** *If A is a full-rank matrix and if E is matrix with rank at most 2, then,*

$$\frac{det(A+E)-\det(A)}{\det(A)} = \lambda_1(A^{-1}E) + \lambda_2(A^{-1}E) + \lambda_1(A^{-1}E)\lambda_2(A^{-1}E)$$

where $\lambda_i(Z)$ is the i-th highest eigenvalue of matrix Z.

Let $A = \bigtriangledown b(\alpha;\mathcal{D}) = -(n \bigtriangledown^2 \hat{\mathcal{L}}(\alpha;\mathcal{D}) + \bigtriangledown^2 r(\alpha) + \Delta \mathbb{I}_p)$, where $\mathbb{I}_p$ is an identity matrix of $p \times p$ dimensions. W.l.o.g. assume that $\mathcal{D}'$ has one entry more as compared to $\mathcal{D}$, and $\mathcal{D}$ has n entries. Let $E = \bigtriangledown^2 \ell(\alpha; d_{n+1})$. Therefore, $|\det(\bigtriangledown b(\alpha;\mathcal{D}'))| = \det(A+E)$. Since $n \bigtriangledown^2 \hat{\mathcal{L}}(\alpha;\mathcal{D}) + \bigtriangledown^2 r(\alpha)$ is positive semi-definite (as both $\hat{\mathcal{L}}$ and r are convex), the smallest eigenvalue of A is Δ. Since E is a positive semi-definite matrix of rank at most one, $A^{-1}E$ has at most one non-zero eigenvalue. Additionally, it follows that $\lambda_1(A^{-1}E) \leq \frac{\lambda_1(E)}{\Delta}$. Applying Lemma 3.10, we have $\frac{det(A+E)}{det(A)} \leq 1 + \frac{\psi}{\Delta}$, since $\lambda_1(E) \leq \psi$ by assumption. Replacing the value of Δ we get $\frac{|\det(\bigtriangledown b(\alpha;\mathcal{D}'))|}{|\det(\bigtriangledown b(\alpha;\mathcal{D}))|} \leq e^{\frac{\epsilon}{2}}$.

To bound $\frac{\nu_2(b(\alpha;\mathcal{D});\epsilon,\delta,\zeta)}{\nu_2(b(\alpha;\mathcal{D}');\epsilon,\delta,\zeta)}$, recall that the noise vector b is drawn from the Gaussian distribution $\mathcal{N}(0,\beta^2\mathbb{I}_p)$, where $\beta = \frac{\zeta\sqrt{8\log\frac{2}{\delta}+4\epsilon}}{\epsilon}$ is the standard deviation. Let us assume $\Gamma = b(\alpha;\mathcal{D}) - b(\alpha;\mathcal{D}')$. With this we have the following:

$$\begin{aligned}
\frac{\nu_2(b(\alpha;\mathcal{D});\epsilon,\delta,\zeta)}{\nu_2(b(\alpha;\mathcal{D}');\epsilon,\delta,\zeta)} &= \frac{e^{-\frac{\|b(\alpha;\mathcal{D})\|_2^2}{2\beta^2}}}{e^{-\frac{\|b(\alpha;\mathcal{D}')\|_2^2}{2\beta^2}}} \\
&= e^{\frac{1}{2\beta^2}\left|\|b(\alpha;\mathcal{D})\|_2^2 - \|b(\alpha;\mathcal{D}')\|_2^2\right|} \\
&= e^{\frac{1}{2\beta^2}\left|\|b(\alpha;\mathcal{D})\|_2^2 - \|b(\alpha;\mathcal{D})-\Gamma\|_2^2\right|} \\
&= e^{\frac{1}{2\beta^2}\left|2\langle b(\alpha;\mathcal{D}),\Gamma\rangle - \|\Gamma\|_2^2\right|}
\end{aligned}$$

Since $\| \bigtriangledown \ell(\theta;\dot{)}\|_2 \leq \zeta$ for all $\theta \in \mathbb{R}^p$ and for all $d \in \mathcal{T}^*$, therefore $\|\Gamma\|_2 \leq \zeta$. Hence the following is true.

$$e^{\frac{1}{2\beta^2}\left|2\langle b(\alpha;\mathcal{D}),\Gamma\rangle - \|\Gamma\|_2^2\right|} \leq e^{\frac{1}{2\beta^2}\left(|2\langle b(\alpha;\mathcal{D}),\Gamma\rangle| + \|\Gamma\|_2^2\right)} \leq e^{\frac{1}{2\beta^2}\left(|2\langle b(\alpha;\mathcal{D}),\Gamma\rangle| + \zeta^2\right)} \tag{3.8}$$

The following two lemmas will be useful in bounding $|\langle b(\alpha;\mathcal{D}),\Gamma\rangle|$. Both of them follow from basic probability theory and hence we skip their proofs.

Lemma 3.11. *Let $Z \sim \mathcal{N}(0,\mathbb{I}_p)$ and $v \in \mathbb{R}^p$ be a fixed vector. Then*

$$\langle Z, v\rangle \sim \mathcal{N}(0, \|v\|_2^2)$$

Note that Γ is independent of the noise vector. Therefore using Lemma 3.11, we get $\langle b(\alpha;\mathcal{D}),\Gamma\rangle \sim \mathcal{N}(, \|\Gamma\|_2^2\beta^2)$. The following lemma provides a tail bound for normal distribution which we use to bound the probability that the noise vector $b(\alpha;\mathcal{D})$ is not in the set **good** .

Lemma 3.12. *Let $Z \sim \mathcal{N}(0,1)$, then for all $t > 1$, we have*

$$\Pr[|Z| > t] \leq e^{-t^2/2}$$

Using this lemma and the fact that $\|\Gamma\|_2 \leq \zeta$, we get $\Pr[|\langle b(\alpha;\mathcal{D}), \Gamma\rangle| \geq \zeta\beta t] \leq e^{-\frac{t^2}{2}}$, where $t > 1$. Let **good** be the set $\{a \in \mathbb{R}^p | \langle a, \Gamma\rangle| \geq \zeta\beta t\}$. We want the noise vector $b(\alpha;\mathcal{D})$ to be in the set **good** w.p. at least $1-\delta$. Setting $t = \sqrt{2\log\frac{2}{\delta}}$ implies that $2e^{-\frac{t^2}{2}} = \delta$. To make sure $t \geq 1$, we need to have $\delta \leq \frac{2}{\sqrt{e}}$. This always true for any non trivial δ. Replacing $t = \sqrt{2\log\frac{2}{\delta}}$ in $\zeta\beta t$, we get from (3.8) that $\frac{\nu_2(b(\alpha;\mathcal{D});\epsilon,\delta,\zeta)}{\nu_2(b(\alpha;\mathcal{D}');\epsilon,\delta,\zeta)} \leq e^{\frac{1}{2\beta^2}\left(\beta\zeta\sqrt{8\log\frac{2}{\delta}}+\zeta^2\right)}$. Solving for β we get $\beta \geq \frac{\zeta\sqrt{8\log\frac{2}{\delta}+4\epsilon}}{\epsilon}$. To complete the argument, we show the following:

$$\begin{aligned}
&\mathrm{pdf}(\theta^{\mathrm{priv}} = \alpha;\mathcal{D}) = \Pr[b \in \mathbf{good}\,]\mathrm{pdf}(\theta^{\mathrm{priv}} = \alpha | b \in \mathbf{good}\,;\mathcal{D}) \\
&+ \Pr[b \in \overline{\mathbf{good}}\,]\mathrm{pdf}(\theta^{\mathrm{priv}} = \alpha | b \in \overline{\mathbf{good}}\,;\mathcal{D}) \\
&\leq \Pr[b \in \mathbf{good}\,]\mathrm{pdf}(\theta^{\mathrm{priv}} = \alpha | b \in \mathbf{good}\,;\mathcal{D}) + \delta \\
&\leq e^{\epsilon}\Pr[b \in \mathbf{good}\,]\mathrm{pdf}(\theta^{\mathrm{priv}} = \alpha | b \in \mathbf{good}\,;\mathcal{D}') + \delta \\
&\leq e^{\epsilon}\mathrm{pdf}(\theta^{\mathrm{priv}} = \alpha;\mathcal{D}') + \delta
\end{aligned}$$

where b is the noise vector in Algorithm 4.1. This concludes the proof of Lemma 3.9. □

3.5.2.2 Extension to non-differentiable regularizers via Successive Approximation

Our first goal is to remove the differentiability assumptions on the regularizer r. To do this, we use the *bump function* [Zemanian, 1987] $\Psi(x) : \mathbb{R} \to \mathbb{R}$ and a sequence of kernel functions $K_i(\theta) : \mathbb{R}^p \to \mathbb{R}$ (for $i = 1, 2, \dots$) defined as:

$$\begin{aligned}
\Psi(x) &= \begin{cases} \exp(-\frac{1}{1-x^2}) & \text{if } |x| < 1 \\ 0 & \text{if } |x| \geq 1 \end{cases} \\
K_i(\theta) &= \frac{\Psi(i\|\theta\|_2^2)}{\int\limits_{\theta' \in \mathbb{R}^p} \Psi(i\|\theta'\|_2^2)\, d\theta'}
\end{aligned} \tag{3.9}$$

The bump function Ψ is infinitely differentiable and all of its derivatives vanish outside the interval $(-1, 1)$ [Zemanian, 1987]. Therefore the kernels K_i are also infinitely differentiable and their support (and that of their derivatives) is $\left\{\theta \ : \ \|\theta\|_2^2 < 1/i\right\}$. Now, if r is a convex regularizer (but not necessarily differentiable), then consider the regularizer r_i defined as the convolution of r and K_i:

$$r_i(\theta) = [r * K_i](\theta) \equiv \int\limits_{y \in \mathbb{R}^p} r(\theta - y)K_i(y)dy$$

By the elementary properties of convolution and the smoothness of K_i, the regularizer r_i is infinitely differentiable. Since convolution with K_i is the same as an (infinite) positive linear combination of translations of r, the regularizer r_i is also convex. Thus we will approximate the objective function $J^{\text{priv}}(\theta, b; \mathcal{D}) = \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{1}{n}(b^T\theta + r(\theta))$ with $J^{\text{priv}^i}(\theta, b; \mathcal{D}) = \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{1}{n}(b^T\theta + r_i(\theta))$, which has a smooth regularizer and to which Lemma 3.9 can be applied. In the next lemma we show that the minimizers of J^{priv} and J^{priv^i} converge pointwise. This will enable us to invoke the successive approximations proof technique for guaranteeing privacy via Lemma 3.4.

Lemma 3.13 (Unconstrained Pointwise Convergence)**.** *Let $\hat{\mathcal{L}}$ be a convex function and r a convex regularizer. Define the kernel function K_i as in* (3.9) *and let $r_i(\theta) = [r * K_i](\theta)$ be the convolution between r and K_i. Define the objective function $J^{priv}(\theta, b; \mathcal{D}) = \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{1}{n}(b^T\theta + r(\theta))$ and $J^{priv^i}(\theta, b; \mathcal{D}) = \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{1}{n}(b^T\theta + r^i(\theta))$. Define the unconstrained minimizers, for each b, as $\phi_{\mathcal{D}}(b) = \operatorname{argmin}_{\theta \in \mathbb{R}^p} J^{priv}(\theta, b; \mathcal{D})$ and $\phi^i_{\mathcal{D}}(b) = \operatorname{argmin}_{\theta \in \mathbb{R}^p} J^{priv^i}(\theta, b; \mathcal{D})$. Then for every $b \in \mathbb{R}^p$, $\lim_{i \to \infty} \phi^i_{\mathcal{D}}(b) = \phi_{\mathcal{D}}(b)$.*

Proof. In order to prove the pointwise convergence of the sequence of functions $\phi^i_{\mathcal{D}}$ to $\phi_{\mathcal{D}}$, we first prove the following claim.

Claim 3.14. *Let $\mathcal{I} \subseteq \mathbb{R}^p$ be a bounded set and let $B \subseteq \mathbb{R}^p$ be any set. The functions $n \cdot J^{priv^i}$ converge uniformly to $n \cdot J^{priv}$ on $\mathcal{I} \times B$ as $i \to \infty$.*

Proof. Choose a $\xi > 0$. Let $\mathcal{I}' = \{y \ : \inf_{\vec{x} \in \mathcal{I}} ||y - \vec{x}||_2 \leq 1\}$ be the set of all points whose distance to $\mathcal{I}$ is at most 1. Note that $\mathcal{I}'$ is closed, bounded, and hence compact. Since $r(\theta)$ is a continuous function defined over the compact set $\mathcal{I}'$, it is then also uniformly continuous on $\mathcal{I}'$. This means that there exists an η (depending only on ξ) such that $|r(\theta_1) - r(\theta_2)| \leq \xi$ whenever $\theta_1, \theta_2 \in \mathcal{I}'$ and $||\theta_1 - \theta_2||_2 \leq \eta$. Now, for any $i > 1/\xi$ and any $\theta \in \mathcal{I}$ and $b \in B$,

$$
\begin{aligned}
n|J^{\text{priv}^i}(\theta, b; \mathcal{D}) - J^{\text{priv}}(\theta, b; \mathcal{D})| &= |r_i(\theta) - r(\theta)| \\
&= \left| \int r(\theta - y) K_i(y)\, dy - r(\theta) \right| \\
&= \left| \int [r(\theta - y) - r(\theta)] K_i(y)\, dy \right| \quad (\because \text{integral of } K_i \text{ is 1}) \\
&\leq \int \left| r(\theta - y) - r(\theta) \right| K_i(y)\, dy
\end{aligned}
$$

$$
\begin{aligned}
&= \int_{\{y \,:\, ||y||_2^2 \leq 1/i\}} \left| r(\theta - y) - r(\theta) \right| K_i(y)\, dy \quad (\text{The support of } K_i) \\
&\leq \int_{\{y \,:\, ||y||_2^2 \leq 1/i\}} \xi K_i(y)\, dy \quad (\text{Since } \theta \in \mathcal{I},\ \theta - y \in \mathcal{I}', \text{ and } ||y||_2 \leq 1/i \leq \xi) \\
&= \xi \quad (\text{Integral of } K_i \text{ over its support is 1})
\end{aligned}
$$

Thus $n \cdot {J^{\text{priv}}}^i$ converge uniformly to $n \cdot J^{\text{priv}}$ on $\mathcal{I} \times B$. □

Now with the uniform conververgence of the objective function in hand we use the following steps to complete the proof for Lemma 3.13.

Step 1: Properties of J^{priv}

In order to prove pointwise convergence, we first establish some simple properties of $J^{\text{priv}}(\theta, b; \mathcal{D})$, which is $\frac{\Delta}{n}$-strongly convex in θ for each fixed b. Recall that for each b, $\phi_{\mathcal{D}}(b)$ returns the unique θ that minimizes $J^{\text{priv}}(\theta, b; \mathcal{D})$ over $\mathbb{R}^p$ (uniqueness is guaranteed by strong convexity). By definition of Δ-strong convexity,

$$J^{\text{priv}}(t\theta_1 + (1-t)\theta_2, b; \mathcal{D}) \leq tJ^{\text{priv}}(\theta_1, b; \mathcal{D}) + (1-t)J^{\text{priv}}(\theta_2, b; \mathcal{D}) - \frac{\Delta}{2n}t(1-t)\|\theta_1 - \theta_2\|_2^2$$

So for all θ and $t \in (0,1)$, recalling that $\phi_{\mathcal{D}}(b)$ is the minimizer of $J^{\text{priv}}(\cdot, b; \mathcal{D})$ over $\mathbb{R}^p$,

$$\begin{aligned}
J^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D}) &\leq J^{\text{priv}}(t\phi_{\mathcal{D}}(b) + (1-t)\theta, b; \mathcal{D}) \\
&\leq tJ^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D}) + (1-t)J^{\text{priv}}(\theta, b; \mathcal{D}) - \frac{\Delta}{2n}t(1-t)\|\phi_{\mathcal{D}}(b) - \theta\|_2^2 \\
(1-t)J^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D}) &\leq (1-t)J^{\text{priv}}(\theta, b; \mathcal{D}) - \frac{\Delta}{2n}t(1-t)\|\phi_{\mathcal{D}}(b) - \theta\|_2^2 \\
\frac{\Delta}{2n}t\|\phi_{\mathcal{D}}(b) - \theta\|_2^2 &\leq J^{\text{priv}}(\theta, b; \mathcal{D}) - J^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D}) \\
\frac{\Delta}{2n}\|\phi_{\mathcal{D}}(b) - \theta\|_2^2 &\leq J^{\text{priv}}(\theta, b; \mathcal{D}) - J^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D}) \qquad (3.10)
\end{aligned}$$

Where the last inequality follows by taking limits as $t \to 1$.

Step 2: Choosing Parameters

Choose any b. Now choose a small ξ such that $\Delta/2 > \xi > 0$. Define $\mathcal{I} = \{\theta : \|\theta - \phi_{\mathcal{D}}(b)\|_2 \leq 1\}$ and the corresponding set $B = \{b' : \phi_{\mathcal{D}}(b') \in \mathcal{I}\}$. Since $\mathcal{I}$ is bounded, we can use the uniform convergence of the $n \cdot {J^{\text{priv}}}^i$ to $n \cdot J^{\text{priv}}$ over $\mathcal{I} \times B$ (from Claim 3.14). Choose an i_ξ depending only on ξ such that for all $i \geq i_\xi$, $\theta \in \mathcal{I}$, and $b' \in B$ the inequality $n|J^{\text{priv}}(\theta, b'; \mathcal{D}) - {J^{\text{priv}}}^i(\theta, b'; \mathcal{D})| \leq \frac{\xi}{3}$ holds.

Step 3: Pointwise Convergence

We now show that $\|\phi^i_{\mathcal{D}}(b) - \phi_{\mathcal{D}}(b)\|_2 \leq \sqrt{4\xi/\Delta}$ for all $i \geq i_\xi$. Assume, by way of contradiction, that $\|\phi^i_{\mathcal{D}}(b) - \phi_{\mathcal{D}}(b)\|_2 > \sqrt{4\xi/\Delta}$ for some $i \geq i_\xi$ and $b \in B$. Then, by the strong convexity of ${J^{\text{priv}}}^i$ (in terms of the parameter θ), there is a θ' along the line from $\phi^i_{\mathcal{D}}(b)$ to $\phi_{\mathcal{D}}(b)$ such that

$${J^{\text{priv}}}^i(\phi^i_{\mathcal{D}}(b), b; \mathcal{D}) < {J^{\text{priv}}}^i(\theta', b; \mathcal{D}) < {J^{\text{priv}}}^i(\phi_{\mathcal{D}}(b), b; \mathcal{D}) \qquad (3.11)$$

$$\text{and} \quad \|\theta' - \phi_{\mathcal{D}}(b)\|_2 = \sqrt{2\xi/\Delta} < 1 \quad (\text{since we chose } \xi < \Delta/2) \qquad (3.12)$$

Since $\theta' \in \mathbb{R}^p$, so by (3.12), $\theta' \in \mathcal{I}$. Now, by (3.10),

$$\xi = \frac{\Delta}{2}\|\phi_{\mathcal{D}}(b) - \theta'\|_2^2 \leq n|{J^{\text{priv}}}_{\mathcal{D}}(\theta', b) - {J^{\text{priv}}}_{\mathcal{D}}(\phi_{\mathcal{D}}(b), b)|$$

$$\leq \quad n \cdot {J^{\text{priv}}}^i_{\mathcal{D}}(\theta', b) + \frac{\xi}{3} - n \cdot {J^{\text{priv}}}^i_{\mathcal{D}}(\phi_{\mathcal{D}}(b), b) + \frac{\xi}{3} \quad \text{(By uniform convergence on } \mathcal{I} \times B\text{)}$$

$$\Rightarrow \quad n \cdot {J^{\text{priv}}}^i(\theta', b; \mathcal{D}) \geq n \cdot {J^{\text{priv}}}^i(\phi_{\mathcal{D}}(b), b; \mathcal{D}) + \frac{\xi}{3}$$

This contradicts the fact that θ' was chosen to satisfy ${J^{\text{priv}}}^i(\theta', b; \mathcal{D}) < {J^{\text{priv}}}^i(\phi_{\mathcal{D}}(b), b; \mathcal{D})$. Thus $\|\phi^i_{\mathcal{D}}(b) - \phi_{\mathcal{D}}(b)\|_2 \leq \sqrt{4\xi/\Delta}$ for all $i \geq i_\xi$ and therefore $\phi^i_{\mathcal{D}}(b) \to \phi_{\mathcal{D}}(b)$ as $i \to \infty$. □

Now invoking Lemmas 3.4 and 3.9 we directly get the following.

Lemma 3.15 (Differentially Private Unconstrained Objective Perturbation)**.** *Under the conditions of Theorem 4.1, if we assume that the convex set $\mathcal{C}$ is the entire real space $\mathbb{R}^p$, then*

- *using Gamma density ν_1, Algorithm 4.1 (Algorithm* Obj-Pert *) guarantees ϵ-differential privacy.*
- *using Gaussian density ν_2, Algorithm 4.1 (Algorithm* Obj-Pert *) guarantees (ϵ, δ)-differential privacy.*

3.5.2.3 Extension to Hard Convex Constraints via Successive Approximation.

In order to extend Lemma 3.15 to Theorem 4.1, we need to show the same (as in Lemma 3.15) when $\mathcal{C}$ is a closed convex subset of $\mathbb{R}^p$. To show this we will again invoke our successive approximations technique.

Consider the function $f(\theta) = \min_{y \in \mathcal{C}} \|\theta - y\|_2$. This function is zero if $\theta \in \mathcal{C}$ and is increasing as θ goes farther away from $\mathcal{C}$. Also notice that f is a convex function. Now, consider the following unconstrained optimization problem.

$$\phi^i_{\mathcal{D}}(b) = \arg\min_{\theta \in \mathbb{R}^p} {J^{\text{priv}}}^i(\theta, b; \mathcal{D}) = \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{1}{n}(r(\theta) + b^T\theta + if(\theta)) \tag{3.13}$$

Correspondingly consider the following optimization problem whose privacy we care about.

$$\phi_{\mathcal{D}}(b) = \arg\min_{\theta \in \mathcal{C}} J^{\text{priv}}(\theta, b; \mathcal{D}) = \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{1}{n}(r(\theta) + b^T\theta) \tag{3.14}$$

Similar to Lemma 3.13, the following Lemma shows the pointwise convergence of $\phi^i_{\mathcal{D}}$ and $\phi_{\mathcal{D}}$.

Lemma 3.16 (Constrained Pointwise Convergence)**.** *Let $\hat{\mathcal{L}}$ be a convex function and r a convex regularizer. For a given closed convex set $\mathcal{C} \subset \mathbb{R}^p$, define the function $f(\theta) = \min_{y \in \mathcal{C}} \|\theta - y\|_2$. Define the objective function $J^{priv}(\theta, b; \mathcal{D})$ and ${J^{priv}}^i(\theta, b; \mathcal{D})$ as in Equations 3.13 and 3.14. Define the minimizers, for each b, as $\phi_{\mathcal{D}}(b) = \operatorname{argmin}_{\theta \in \mathcal{C}} J^{priv}(\theta, b; \mathcal{D})$ and $\phi^i_{\mathcal{D}}(b) = \operatorname{argmin}_{\theta \in \mathbb{R}^p} {J^{priv}}^i(\theta, b; \mathcal{D})$. Then for every $b \in \mathbb{R}^p$, $\lim_{i \to \infty} \phi^i_{\mathcal{D}}(b) = \phi_{\mathcal{D}}(b)$.*

Proof. Before we prove Lemma 3.16, we will state some simple properties about strongly convex functions. These properties will be needed in the argument of the proof of Lemma 3.16.

Claim 3.17. *Let g be a Δ-strongly convex function and let $\hat{\theta}$ be the minimizer of g over a convex set M. Then $g(\theta) - g(\hat{\theta}) \geq \frac{\Delta}{2}\|\theta - \hat{\theta}\|_2$ for all $\theta \in M$.*

Claim 3.18. *Let g be a convex function and let $\theta_1, \theta_2 \in \mathbb{R}^p$ and let $s \geq 1$. Then*

$$\frac{g(\theta_1 + s\theta_2) - g(\theta_1)}{\|s\theta_2\|_2} \geq \frac{g(\theta_1 + \theta_2) - g(\theta_1)}{\|\theta_2\|_2}$$

With these two claims in hand we complete the proof of Lemma 3.16.

In the following set of arguments we will show that for any $b \in \mathbb{R}^p$, there exists an i_0 s.t. $\forall i > i_0$, $\phi^i_{\mathcal{D}}(b) = \phi_{\mathcal{D}}(b)$. This will then directly imply that ϕ^i converges pointwise to $\phi_{\mathcal{D}}$. Consider any $b \in \mathbb{R}^p$. First note that $\phi^0_{\mathcal{D}}(b)$ is the unconstrained minimizer. By strong convexity, the unconstrained minimizer $\phi^0_{\mathcal{D}}(b)$ exists and the constrained minimizer $\phi_{\mathcal{D}}(b)$ also exists since $\mathcal{C}$ is closed and convex. If $\phi^0_{\mathcal{D}}(b) = \phi_{\mathcal{D}}(b)$, then we are done. If $\phi^0_{\mathcal{D}}(b) \in \mathcal{C}$, then $\phi^0_{\mathcal{D}}(b) = \phi_{\mathcal{D}}(b)$ by strong convexity (since $\phi_{\mathcal{D}}(b)$ is a minimizer over $\mathcal{C}$) and we are also done. Thus, we may assume $\phi^0_{\mathcal{D}}(b) \neq \phi_{\mathcal{D}}(b)$ and $\phi^0_{\mathcal{D}}(b) \notin \mathcal{C}$.

For any θ, let $\theta_{\mathcal{C}}$ be the point in $\mathcal{C}$ that is closest to θ (existence is guaranteed because $\mathcal{C}$ is closed and uniqueness is guaranteed because $\mathcal{C}$ is convex) so that $\|\theta - \theta_{\mathcal{C}}\|_2 = f(\theta)$.

Now consider the set of points $H \equiv \{\theta \ : \ J(\theta, b; \mathcal{D}) \leq J^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D})\}$. Clearly $\phi^i_{\mathcal{D}}(b) \in H$ for all i. Let $d = \sqrt{n \cdot J^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D}) - n \cdot J^{\text{priv}}(\phi^0_{\mathcal{D}}(b), b; \mathcal{D})}$. By Claim 3.17, H lies in the closed ball B with center $\phi^0_{\mathcal{D}}(b)$ and radius d. Since $\phi_{\mathcal{D}}(b) \in H \subseteq B$, the farthest distance from any point $\theta \in B$ to $\mathcal{C}$ is $\|\theta - \theta_F\|_2 \leq \|\theta - \phi_{\mathcal{D}}(b)\|_2 \leq 2d$.

Consider the function $\kappa : B \times \{v \in \mathbb{R}^p \ : \ \|v\|_2 = 2d\}$ defined as $\kappa(\theta, v) = \frac{n \cdot J^{\text{priv}}(\theta + v, b; \mathcal{D}) - n \cdot J^{\text{priv}}(\theta, b; \mathcal{D})}{\|v\|_2}$. Let m be the supremum of κ (it is finite since κ is a continuous function over a compact set). Then for any $\theta \in H$, using Claim 3.18 (with $\theta_1 \equiv \theta$, $\theta_2 \equiv \theta_{\mathcal{C}} - \theta$, and $s \equiv \frac{2d}{\|\theta_{\mathcal{C}} - \theta\|_2}$), we have $\frac{n \cdot J^{\text{priv}}(\theta_{\mathcal{C}}, b; \mathcal{D}) - f(\theta)}{\|\theta_{\mathcal{C}} - \theta\|_2} \leq m$.

Now set $\alpha = 2m$. Then for any $\theta \in H \subseteq B$ with $\theta \notin \mathcal{C}$,

$$\begin{aligned}
n \cdot J^{\text{priv}}(\theta, b; \mathcal{D}) + if(\theta) &= n \cdot J^{\text{priv}}(\theta_{\mathcal{C}}, b; \mathcal{D}) + if(\theta) \\
&\quad - \frac{n \cdot J^{\text{priv}}(\theta_{\mathcal{C}}, b; \mathcal{D}) - n \cdot J^{\text{priv}}(\theta, b; \mathcal{D})}{\|\theta_{\mathcal{C}} - \theta\|_2}\|\theta_{\mathcal{C}} - \theta\|_2 \\
&\geq n \cdot J^{\text{priv}}(\theta_{\mathcal{C}}, b; \mathcal{D}) + if(\theta) - m\|\theta_{\mathcal{C}} - \theta\|_2 \\
&= n \cdot J^{\text{priv}}(\theta_{\mathcal{C}}, b; \mathcal{D}) + 2mf(\theta) - mf(\theta) \\
&\geq n \cdot J^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D}) + mf(\theta) \\
&> n \cdot J^{\text{priv}}(\phi_{\mathcal{D}}(b), b; \mathcal{D}) \quad \text{(since } \theta \notin \mathcal{C} \text{ and } \mathcal{C} \text{ is closed)}
\end{aligned}$$

Since $\phi^i_{\mathcal{D}}(b) \in H$, this means $\phi^i_{\mathcal{D}}(b) = \phi_{\mathcal{D}}(b)$, contradicting the assumption that $\phi^i_{\mathcal{D}}(b) \notin \mathcal{C}$.

This completes the proof of Lemma 3.16. □

To complete the proof of Private Convex Optimization theorem i.e.,Theorem 4.1,

notice that in the successive approximations in (3.13), differential privacy condition holds on any subset of $\mathcal{C}$. We know that for the original constrained optimization, the probability measure on any set outside the constraint set $\mathcal{C}$ is zero. So, it suffices to argue about differential privacy condition only on subsets of $\mathcal{C}$. Using the results of Lemmas 3.15 and 3.16, and invoking Lemma 3.4 on all subsets of $\mathcal{C}$, we complete the proof of Private Convex Optimization theorem, i.e.,Theorem 4.1.

3.5.3 Utility Analysis (Empirical Risk and Generalization Error)

Empirical Risk: The following lemma bounds the empirical risk (i.e.,$\hat{J}(\theta^{\text{priv}};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D})$) of Algorithm 4.1 (Algorithm Obj-Pert).

Lemma 3.19 (Empirical risk)**.** *Let $\hat{\theta}$ be the minimizer of the empirical objective function $\hat{J}(\theta;\mathcal{D})$ over the closed convex set $\mathcal{C}$ and let θ^{priv} be the output of Algorithm 4.1. We have $\hat{J}(\theta^{priv};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D}) \leq \frac{2\|b\|_2^2}{\Delta n} + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2$.*

We defer the proof of this lemma till Section 3.5.4. Using the tail bounds for the noise distributions used in Algorithm Obj-Pert , we obtain the following theorem as a corollary of the above lemma. A detailed proof of this theorem is given in Section 3.5.4.

Theorem 3.20 (Theorem 3.24, special case)**.** *Assume that $\| \bigtriangledown \ell(\theta;d)\|_2 \leq \zeta$ (for all $\theta \in \mathcal{C}$ and for all $d \in \mathcal{P}$). Let λ be the maximum eigenvalue bound on $\bigtriangledown^2 \ell$.*

1. *[Chaudhuri et al., 2011] With Gamma density ν_1, setting $\Delta = \Theta\left(\frac{\zeta p \log p}{\epsilon\|\hat{\theta}\|_2}\right)$ and assuming $\Delta \geq \frac{\lambda}{2\epsilon}$, we have $\mathbb{E}\left[\hat{J}(\theta^{priv};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D})\right] = O\left(\frac{\zeta\|\hat{\theta}\|_2 p \log p}{\epsilon n}\right)$.*
2. *(This work) With Gaussian density ν_2, setting $\Delta = \Theta\left(\frac{\sqrt{\zeta^2 p \log(1/\delta)}}{\epsilon\|\hat{\theta}\|_2}\right)$ and assuming $\Delta \geq \frac{\lambda}{2\epsilon}$, we have $\mathbb{E}\left[\hat{J}(\theta^{priv};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D})\right] = O\left(\frac{\zeta\|\hat{\theta}\|_2\sqrt{p\log(1/\delta)}}{\epsilon n}\right)$.*

Note that the empirical risk bounds in Theorem 3.20 are for the ideal choices of Δ. Optimal Δ depends on the L_2 norm of the true minimizer ($\hat{\theta}$) of $\hat{J}$. In practice, if the exact bound on $\|\hat{\theta}\|_2$ is not known, then one can replace it with a loose upper bound, *e.g.,* a bound on the diameter of the convex set $\mathcal{C}$.

The main takeaway from Theorem 3.20 is that, ignoring the privacy parameters (ϵ, δ), the empirical risk bound for the Gamma distribution (ν_1) is at least $\sqrt{p}$ times higher than for Gaussian distribution (ν_2). Intuitively, this gap arises from the fact that the vectors drawn from ν_2 are more tightly concentrated around the mean as compared to ν_1. For an application of Theorem 3.20 above to linear regression, see Section 4.2 in Chapter 4.

Generalizaion Error: In our presentation of generalization error we restrict ourselves to *Generalized Linear Models* (GLM). In GLM, each data entry d in the data set is of the form (y, x), where $y \in \mathbb{R}$ and $x \in \mathbb{R}^p$. The loss function $\ell(\theta;d)$ is of the form $\ell_{\text{GLM}}(x^T\theta; y)$, where $d = (y, x)$.

Following is the generalization error bound we obtain as a corollary to Theorem 3.20 by using [Shalev-Shwartz et al., 2009, Theorem 2] to convert from empirical risk to generalization error. In the rest of this chapter, we only concentrate on empirical risk as one can easily convert it to generalization error via the result discussed above. For the simplicity of exposition, in this section we assume that the regularizer $\frac{1}{n}r(\theta)$ (in the loss $\bar{J}(\theta;\mathcal{D})$) is zero for all θ.

Theorem 3.21. *Consider that for any data entry $d = (y, x)$ (where $y \in \mathbb{R}$ and $x \in \mathbb{R}^p$), $\|x\|_2 \leq R$. Also assume that $|\ell'_{GLM}(u; y)| \leq L$ and $|\ell''_{GLM}(u; y)| \leq (\epsilon\Delta)/(2R^2)$, where $u \in \mathbb{R}$ and $y \in \mathbb{R}$, and the derivatives are w.r.t. u. When using Gaussian density ν_2 for the noise vector b and setting $\Delta = \Theta\left(\frac{\sqrt{n(RL)^2 p\log(1/\delta)}}{\epsilon\|\bar{\theta}\|_2}\right)$, we have*

$$\mathbb{E}_b\left[\bar{J}(\theta^{priv};\mathcal{P}) - \bar{J}(\bar{\theta};\mathcal{P})\right] = O\left(\frac{(RL)\sqrt{p\log(1/\delta)}\|\bar{\theta}\|_2}{\epsilon n^{1/2}}\right).$$

The main takeaway from the above theorem is that the generalization error for the private version is worse by a factor of $\sqrt{p}$ compared to the emprical risk minimizer.

In all utility guarantees in this chapter, using the *Gamma* noise distribution results in an $\sqrt{p}$ increase in the error. So in the rest of our discussion, we will only concentrate on *Gaussian* noise distribution and hence guarantee (ϵ, δ)-differential privacy with $\delta > 0$.

3.5.4 Estimating Empirical Risk: Proofs of Lemma 3.19 and Theorem 3.20

To bound the empirical risk (Lemma 3.19 and Theorem 3.20) mentioned in Section 3.5.3, we need the following helper lemma.

Lemma 3.22. *Let $\mathcal{D} = \{d_1, \ldots, d_n\}$ be a data set, and let $\hat{J}(\theta;\mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n}\ell(\theta; d_i) + \frac{r(\theta)}{n}$. Let $\theta^{\#} = \arg\min_{\theta\in\mathcal{C}}\hat{J}(\theta;\mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2$ and let θ^{priv} be the output of Algorithm 4.1 (Algorithm* Obj-Pert *), where $\mathcal{C} \subseteq \mathbb{R}^p$ is a closed convex set. Then*

$$\|\theta^{\#} - \theta^{priv}\|_2 \leq \frac{2\|b\|_2}{\Delta}$$

where b is the noise vector in Algorithm 4.1.

Proof. We have $\theta^{\text{priv}} = \arg\min_{\theta\in\mathcal{C}} \underbrace{J^{\#}(\theta;\mathcal{D}) + \frac{b^T\theta}{n}}_{J^{\text{priv}}(\theta;\mathcal{D})}$, where $J^{\#}(\theta;\mathcal{D}) = \hat{J}(\theta;\mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2$.

Similarly, $\theta^{\#} = \arg\min_{\theta\in\mathcal{C}} J^{\#}(\theta;\mathcal{D})$.

Since θ^{priv} is the minimizer of $J^{\text{priv}}(\theta;\mathcal{D})$ and J^{priv} is $\frac{\Delta}{n}$ strongly convex in θ, we have the following (from Claim 3.17):

$$J^{\text{priv}}(\theta^{\#};\mathcal{D}) \geq J^{\text{priv}}(\theta^{\text{priv}};\mathcal{D}) + \frac{\Delta}{2n}\|\theta^{\#} - \theta^{\text{priv}}\|_2^2 \tag{3.15}$$

$$\Rightarrow J^{\#}(\theta^{\#};\mathcal{D}) + \frac{b^T\theta^{\#}}{n} \geq J^{\#}(\theta^{\text{priv}};\mathcal{D}) + \frac{b^T\theta^{\text{priv}}}{n} + \frac{\Delta}{2n}\|\theta^{\#} - \theta^{\text{priv}}\|_2^2 \quad (3.16)$$

Notice that $J^{\#}(\theta^{\#};\mathcal{D}) \leq J^{\#}(\theta^{\text{priv}};\mathcal{D})$, since $\theta^{\#}$ is the minimizer of $J^{\#}(\theta;\mathcal{D})$. Therefore, we have the following:

$$\begin{aligned}
& b^T\theta^{\#} \geq b^T\theta^{\text{priv}} + \frac{\Delta}{2}\|\theta^{\#} - \theta^{\text{priv}}\|_2^2 \\
& \Rightarrow b^T(\theta^{\#} - \theta^{\text{priv}}) \geq \frac{\Delta}{2}\|\theta^{\#} - \theta^{\text{priv}}\|_2^2 \\
& \Rightarrow \|b\|_2\|\theta^{\#} - \theta^{\text{priv}}\|_2 \geq \frac{\Delta}{2}\|\theta^{\#} - \theta^{\text{priv}}\|_2^2 \\
& \Rightarrow \|\theta^{\#} - \theta^{\text{priv}}\|_2 \leq \frac{2\|b\|_2}{\Delta}
\end{aligned}$$

Hence proved. □

The following corollary bounds the difference in the values of the objective function $J^{\#}$ at θ^{priv} and $\theta^{\#}$. The gap is due to the noise variable b.

Corollary 3.23. *Let $\theta^{\#} = \arg\min_{\theta\in\mathcal{C}} \hat{J}(\theta;\mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2$ and let θ^{priv} be the output of Algorithm 4.1 (Algorithm* Obj-Pert *), where $\mathcal{C} \subseteq \mathbb{R}^p$ is a closed convex set. Then*

$$J^{\#}(\theta^{priv};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D}) \leq \frac{2\|b\|_2^2}{\Delta n}$$

where b is the noise vector in Algorithm 4.1.

Proof. From Equation 3.16 of the previous lemma, we have

$$\begin{aligned}
& J^{\#}(\theta^{\#};\mathcal{D}) + \frac{b^T\theta^{\#}}{n} \geq J^{\#}(\theta^{\text{priv}};\mathcal{D}) + \frac{b^T\theta^{\text{priv}}}{n} + \frac{\Delta}{2n}\|\theta^{\#} - \theta^{\text{priv}}\|_2^2 \\
& \Rightarrow J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D}) \leq \frac{b^T(\theta^{\#} - \theta^{\text{priv}})}{n} - \frac{\Delta}{2n}\|\theta^{\#} - \theta^{\text{priv}}\|_2^2 \\
& \Rightarrow J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D}) \leq \frac{\|b\|_2\|\theta^{\#} - \theta^{\text{priv}}\|_2}{n} \\
& \Rightarrow J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D}) \leq \frac{2\|b\|_2^2}{n\Delta}
\end{aligned}$$

The last inequality follows from Lemma 3.22. This completes the proof. □

3.5.4.1 Proof of Lemma 3.19

Proof. We have

$$\begin{aligned}
\hat{J}(\theta^{\text{priv}};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D}) \quad = \quad & (J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D})) + (J^{\#}(\theta^{\#};\mathcal{D}) - J^{\#}(\hat{\theta};\mathcal{D})) \\
& + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2 - \frac{\Delta}{2n}\|\theta^{\text{priv}}\|_2^2
\end{aligned}$$

Notice that $(J^{\#}(\theta^{\#};\mathcal{D}) - J^{\#}(\hat{\theta};\mathcal{D})) \leq 0$. Also from Corollary 3.23 we have $(J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D})) \leq \frac{2\|b\|_2^2}{n\Delta}$. Hence, we have

$$\hat{J}(\theta^{\text{priv}};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D}) \leq \frac{2\|b\|_2^2}{n\Delta} + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2$$

This completes the proof. □

3.5.4.2 Proof of Theorem 3.20

In order to prove Theorem 3.20, we prove the following which is a slightly generalized version. Replacing $\Delta = \Theta\left(\frac{\zeta p \log p}{\epsilon\|\hat{\theta}\|_2}\right)$ in the first part of Theorem 3.24 and $\Delta = \Theta\left(\frac{\sqrt{\zeta^2 p \log\frac{1}{\delta}}}{\epsilon\|\hat{\theta}\|_2}\right)$ in the second part of Theorem 3.24, we obtain Theorem 3.20. Since, we are looking at expected error in Theorem 3.20, we ignore the term γ.

Theorem 3.24. *Assuming that $\| \bigtriangledown \ell(\theta; d)\|_2 \leq \zeta$ (for all $d \in \mathcal{P}$ and for all $\theta \in \mathcal{C}$), the following are true.*

1. *With Gamma density ν_1, w.p. $\geq 1 - \gamma$*

$$\hat{J}(\theta^{priv};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D}) \leq \frac{8\zeta^2 p^2 \log^2\frac{p}{\gamma}}{n\epsilon^2\Delta} + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2$$

2. *With Gaussian density ν_2, w.p. $\geq 1 - \gamma$*

$$\hat{J}(\theta^{priv};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D}) \leq \frac{4p\zeta^2(8\log\frac{2}{\delta} + 4\epsilon)\log(1/\gamma)}{n\epsilon^2\Delta} + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2$$

Proof. The proof essentially goes via bounding $\|b\|_2$ under the two distributions ν_1 and ν_2 used in Algorithm Obj-Pert (Algorithm 4.1) and plugging it in Lemma 3.19.

Recall that distribution $\nu_1(b;\epsilon,\zeta) \propto e^{-\frac{\|b\|_2}{2\zeta}}$. Thus, under the distribution ν_1 for b, we have $\|b\|_2 \sim \Gamma(p, \frac{2\zeta}{\epsilon})$. The following lemma from [Chaudhuri et al., 2011] provides a tail bound for *Gamma* distribution.

Lemma 3.25 (Lemma 4 from [Chaudhuri et al., 2011])**.** *Let X be a random variable drawn from the distribution $\Gamma(p,\theta)$, where p is a positive integer. Then,*

$$\Pr\left[X \geq p\theta\log\frac{p}{\gamma}\right] \leq \gamma$$

Using Lemma 3.25, w.p. $\geq 1 - \gamma$ we have the following:

$$\|b\|_2 \leq \frac{2p\zeta\log\frac{p}{\gamma}}{\epsilon}$$

Plugging in the value of $\|b\|_2$ from above into Lemma 3.19, we have w.p. $\geq 1-\gamma$

$$\hat{J}(\theta^{\text{priv}};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D}) \leq \frac{8\zeta^2 p^2 \log^2 \frac{p}{\gamma}}{n\epsilon^2\Delta} + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2$$

This completes the proof of first part of the theorem.

For the second part, we need to bound $\|b\|_2$ when $b \sim \mathcal{N}\left(0, \mathbb{I}_p \frac{\zeta^2\left(8\log\frac{2}{\delta}+4\epsilon\right)}{\epsilon^2}\right)$. We use the following lemma from [Dasgupta and Schulman, 2007].

Lemma 3.26 (Lemma 2 from [Dasgupta and Schulman, 2007])**.** *Pick X from the distribution $\mathcal{N}(0,\mathbb{I}_p)$. Then for any $\phi \geq 1$, we have*

$$\Pr[\|X\|_2 \geq \sqrt{\phi p}] \leq e^{-\frac{p}{2}(\phi - 1 - \log\phi)}$$

In the above lemma, in order to set $e^{-\frac{p}{2}(\phi-1-\log\phi)} \leq \gamma$, we need $1 + \frac{2}{p}\log\frac{1}{\gamma} \leq \frac{\phi}{2}$. Therefore, setting ϕ as above, we have w.p. $\geq 1-\gamma$,

$$\|X\|_2 \leq \sqrt{2 + \frac{2}{p}\log\frac{1}{\gamma}}\sqrt{p}$$
$$\Rightarrow \|X\|_2 \leq \sqrt{2p\log\frac{1}{\gamma}}$$

Using the above bound we have w.p. $\geq 1-\gamma$,

$$\|b\|_2 \leq \sqrt{\frac{2p\zeta^2\left(8\log\frac{2}{\delta}+4\epsilon\right)\log\frac{1}{\gamma}}{\epsilon^2}}$$

Plugging in the value of $\|b\|_2$ from above into Lemma 3.19, we have w.p. $\geq 1-\gamma$,

$$\hat{J}(\theta^{\text{priv}};\mathcal{D}) - \hat{J}(\hat{\theta};\mathcal{D}) \leq \frac{4p\zeta^2(8\log\frac{2}{\delta}+4\epsilon)\log(1/\gamma)}{n\epsilon^2\Delta} + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2$$

This completes the proof of second part of the theorem. □

3.5.4.3 Refined Utility Guarantees Under Stronger Assumptions

In this section we provide refined utility guarantees for Algorithm Obj-Pert (Algorithm 4.1) based on stronger assumptions on the underlying data set. Our analysis is inspired by the work of [Dwork et al., 2009] which specifically analyzes logistic regression under such a setting.

For the simplicity of exposition, assume the empirical objective function $\hat{J}(\theta;\mathcal{D})$ equals the empirical loss function $\hat{\mathcal{L}}(\theta;\mathcal{D})$, i.e.,the regularizer $r(\theta)$ is set to zero. Suppose the empirical loss function $\hat{\mathcal{L}}$ is (η/n)-strongly convex within a ball of radius ψ (which will be fixed later) around $\hat{\theta}$, where $\hat{\theta}$ is the minimizer of $\hat{\mathcal{L}}$.

Theorem 3.28 bounds the empirical risk based on this stronger assumption on the loss function . In order to make the result more informative, we state a special case of

Theorem 3.30 (Section 3.5.4.4.2) below. In the following theorem we have assumed the following.

Assumption 3.27. *Assume: i)* $\eta = \Omega(\lambda n/p)$, *ii)* $\psi \geq \frac{p^{3/2}\zeta\sqrt{\log(1/\delta)}}{\lambda n\epsilon^2} + \sqrt{\frac{p}{n}}\|\hat{\theta}\|_2$, *iii)* $n \geq p^2$.

Intuitively, the assumption on η makes sense because if each of $\bigtriangledown^2\ell(\theta; d_i)$ is a rank-one matrix with an eigenvalue $\lambda > 0$ and the eigenvalues of $\bigtriangledown^2\hat{\mathcal{L}}$ are spread out across all dimensions, then we would expect $\sum_{i=1}^n \ell(\theta; d_i)$ to have minimum eigenvalue of Σ to be $\Omega(\lambda n/p)$ (since there are p dimensions).

Theorem 3.28 (Theorem 3.30, special case)**.** *Let* $\Delta = 2\lambda/\epsilon$ *(where* λ *is the bound on the maximum eigenvalue of* $\bigtriangledown^2\ell$*). Under Assumption 3.27, using Gaussian density* ν_2*, we have* $\mathbb{E}\left[\hat{\mathcal{L}}(\theta^{priv}; \mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta}; \mathcal{D})\right] = O\left(\frac{1}{n\epsilon}\left(\frac{p^2\zeta^2\log(1/\delta)}{\lambda n\epsilon} + \lambda\|\hat{\theta}\|_2^2\right)\right)$.

The proof is given in Section 3.5.4.4.2 below.

3.5.4.4 Proofs for Refined Utility Guarantees Under Stronger Assumptions

3.5.4.4.1 Parameter Estimation Error bounds

Theorem 3.29 (Parameter estimation error)**.** *Under the assumption that* $\|\bigtriangledown\ell(\theta; d)\|_2 \leq \zeta$ *(for all* $d \in \mathcal{P}$ *and for all* $\theta \in \mathcal{C}$*), the following are true for Algorithm 4.1.*

1. *When using the Gaussian density* ν_2 *and* $\psi \geq \frac{\sqrt{32p}\zeta\sqrt{(8\log\frac{2}{\delta}+4\epsilon)\log(1/\gamma)}}{\epsilon(\Delta+\eta)}$ $+2\sqrt{\frac{\Delta}{(\Delta+\eta)}}\|\hat{\theta}\|_2$*, then w.p.* $\geq 1-\gamma$ *the following are true.*

   (a) $\|\theta^{priv} - \theta^{\#}\|_2 \leq \frac{\sqrt{2p}\zeta\sqrt{(8\log\frac{2}{\delta}+4\epsilon)\log(1/\gamma)}}{\epsilon(\Delta+\eta)}$

   (b) $\|\theta^{priv} - \hat{\theta}\|_2 \leq \frac{\sqrt{2p}\zeta\sqrt{(8\log\frac{2}{\delta}+4\epsilon)\log(1/\gamma)}}{\epsilon(\Delta+\eta)} + \sqrt{\frac{\Delta}{(\Delta+\eta)}}\|\hat{\theta}\|_2$

Here ψ *is the radius of the ball around* $\hat{\theta}$ *where* $\hat{\mathcal{L}}$ *is* $\frac{\eta}{n}$*-strongly convex.*

Proof. By assumption we have $\hat{\mathcal{L}}(\theta; \mathcal{D})$ is η/n-strongly convex in a ball of radius ψ around $\hat{\theta}$, where $\hat{\theta} = \arg\min_{\theta\in\mathcal{C}} \hat{\mathcal{L}}(\theta; \mathcal{D})$. We will fix the value of ψ later.

Recall that

$$\theta^{\text{priv}} = \arg\min_{\theta\in\mathcal{C}} \underbrace{\hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{b^T\theta}{n}}_{J^{\text{priv}}(\theta;\mathcal{D})}$$

and

$$\theta^{\#} = \arg\min_{\theta\in\mathcal{C}} \underbrace{\hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\Delta}{2n}\|\theta\|_2^2}_{J^{\#}(\theta;\mathcal{D})}$$

where b is the noise vector used in Algorithm Obj-Pert (Algorithm 4.1).

Assume for now that $\psi \geq 2\left(\|\theta^{\text{priv}} - \theta^{\#}\|_2 + \|\theta^{\#} - \hat{\theta}\|_2\right)$. We will remove this assumption as we move along. The above assumption implies the following:

1. $\hat{\mathcal{L}}(\theta;\mathcal{D})$ is $\frac{\eta}{n}$-strongly convex in a ball of radius $\|\theta^{\#}-\hat{\theta}\|_2$ around $\theta^{\#}$.

2. $\hat{\mathcal{L}}(\theta;\mathcal{D})$ is $\frac{\eta}{n}$-strongly convex in a ball of radius $\|\theta^{\text{priv}}-\theta^{\#}\|_2$ around θ^{priv}.

In order to bound $\|\theta^{\text{priv}}-\hat{\theta}\|_2$, we first bound $\|\theta^{\text{priv}}-\theta^{\#}\|_2$ and $\|\theta^{\#}-\hat{\theta}\|_2$ individually. Since $\|\theta^{\text{priv}}-\hat{\theta}\|_2 \leq \|\theta^{\text{priv}}-\theta^{\#}\|_2 + \|\theta^{\#}-\hat{\theta}\|_2$, we obtain the required bound.

Since θ^{priv} is the minimizer of $J^{\text{priv}}(\theta;\mathcal{D})$, the following is true from the definition of J^{priv}.

$$J^{\text{priv}}(\theta^{\#};\mathcal{D}) \geq J^{\text{priv}}(\theta^{\text{priv}};\mathcal{D}) + \frac{\Delta+\eta}{2n}\|\theta^{\#}-\theta^{\text{priv}}\|_2^2 \tag{3.17}$$

$$\Rightarrow J^{\#}(\theta^{\#};\mathcal{D}) + \frac{b^T\theta^{\#}}{n} \geq J^{\#}(\theta^{\text{priv}};\mathcal{D}) + \frac{b^T\theta^{\text{priv}}}{n} + \frac{\Delta+\eta}{2n}\|\theta^{\#}-\theta^{\text{priv}}\|_2^2 \tag{3.18}$$

Notice that $J^{\#}(\theta^{\#};\mathcal{D}) \leq J^{\#}(\theta^{\text{priv}};\mathcal{D})$, since $\theta^{\#}$ is the minimizer of $J^{\#}(\theta;\mathcal{D})$. Therefore we have the following:

$$b^T\theta^{\#} \geq b^T\theta^{\text{priv}} + \frac{\Delta+\eta}{2}\|\theta^{\#}-\theta^{\text{priv}}\|_2^2 \tag{3.19}$$

$$\Rightarrow b^T(\theta^{\#}-\theta^{\text{priv}}) \geq \frac{\Delta+\eta}{2}\|\theta^{\#}-\theta^{\text{priv}}\|_2^2 \tag{3.20}$$

$$\Rightarrow \|b\|_2\|\theta^{\#}-\theta^{\text{priv}}\|_2 \geq \frac{1}{2}\|\theta^{\#}-\theta^{\text{priv}}\|_2^2(\Delta+\eta) \tag{3.21}$$

$$\Rightarrow \|\theta^{\#}-\theta^{\text{priv}}\|_2 \leq \frac{2\|b\|_2}{(\Delta+\eta)} \tag{3.22}$$

Here, η is the local strong convexity parameter.

In order to bound $\|\theta^{\#}-\hat{\theta}\|_2$, we first notice the following:

$$\hat{\mathcal{L}}(\hat{\theta}) + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2 \geq \hat{\mathcal{L}}(\theta^{\#}) + \frac{\Delta}{2n}\|\theta^{\#}\|_2^2 + \frac{\Delta+\eta}{2n}\|\theta^{\#}-\theta^{\text{priv}}\|_2^2 \tag{3.23}$$

$$\Rightarrow \Delta\|\hat{\theta}\|_2^2 \geq (\Delta+\eta)\|\hat{\theta}-\theta^{\#}\|_2^2 \tag{3.24}$$

$$\Rightarrow \|\hat{\theta}-\theta^{\#}\|_2 \leq \sqrt{\frac{\Delta}{\Delta+\eta}}\|\hat{\theta}\|_2 \tag{3.25}$$

From Equations 3.22 and 3.25, it follows that

$$\|\theta^{\text{priv}}-\hat{\theta}\|_2 \leq \frac{2\|b\|_2}{(\Delta+\eta)} + \sqrt{\frac{\Delta}{\Delta+\eta}}\|\hat{\theta}\|_2$$

Equations 3.22 and 3.25 also imply a bound on the radius ψ, so that the initial assumption $\psi \geq 2\left(\|\theta^{\text{priv}}-\theta^{\#}\|_2 + \|\theta^{\#}-\hat{\theta}\|_2\right)$ is true. We set $\psi \geq 2\left(\frac{2\|b\|_2}{(\Delta+\eta)} + \sqrt{\frac{\Delta}{\Delta+\eta}}\|\hat{\theta}\|_2\right)$ to satisfy the above assumption.

From the tail bound calculations for $\|b\|_2$ in Section 3.5.4.2, w.p. $\geq 1-\gamma$ we have the following.

- In Algorithm Obj-Pert (Algorithm 4.1) when the noise distribution is ν_2, we have

$$\|b\|_2 \leq \sqrt{\frac{2p\zeta^2\left(8\log\frac{2}{\delta}+4\epsilon\right)\log\frac{1}{\gamma}}{\epsilon^2}}$$

Plugging in these bounds for $\|b\|_2$, completes the proof. □

3.5.4.4.2 Proof of Theorem 3.28

Proof. In order to prove Theorem 3.28, we prove the following slightly generalized version. Substituting the parameters from Assumption 3.27 in Theorem 3.30 and setting $\Delta = 2\lambda/\epsilon$, we obtain Theorem 3.28. Note that in Theorem 3.28 we ignore the term γ, since there we are dealing with expected error.

Theorem 3.30. *When using the Gaussian distribution function ν_2 and $\psi \geq \frac{\sqrt{p}\zeta\sqrt{\log(1/\delta)\log(1/\gamma)}}{\epsilon(\Delta+\eta)} + \sqrt{\frac{\Delta}{(\Delta+\eta)}}\|\hat{\theta}\|_2$, then w.p. $\geq 1-\gamma$ the following is true.*

$$\hat{\mathcal{L}}(\theta^{priv};\mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta};\mathcal{D}) = O\left(\frac{p\zeta^2\log(1/\delta)\log(1/\gamma)}{n\epsilon^2(\Delta+\eta)} + \frac{\Delta}{n}\|\hat{\theta}\|_2^2\right)$$

Here ψ is the radius of the ball around $\hat{\theta}$ where $\hat{\mathcal{L}}$ is $\frac{\eta}{n}$-strongly convex.

Proof. Using Equation 3.18 from Section 3.5.4.4.1 we have

$$J^{\#}(\theta^{\#};\mathcal{D}) + \frac{b^T\theta^{\#}}{n} \geq J^{\#}(\theta^{\text{priv}};\mathcal{D}) + \frac{b^T\theta^{\text{priv}}}{n} + \frac{\Delta+\eta}{2n}\|\theta^{\#}-\theta^{\text{priv}}\|_2^2 \tag{3.26}$$

$$\Rightarrow J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D}) \leq \frac{b^T(\theta^{\#}-\theta^{\text{priv}})}{n} - \frac{\Delta+\eta}{2n}\|\theta^{\#}-\theta^{\text{priv}}\|_2^2 \tag{3.27}$$

$$\Rightarrow J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D}) \leq \frac{\|b\|_2\|\theta^{\#}-\theta^{\text{priv}}\|_2}{n} \tag{3.28}$$

The last step follows from Cauchy-Schwarz inequality. Recall that

$$\begin{aligned}\hat{\mathcal{L}}(\theta^{\text{priv}};\mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta};\mathcal{D}) &= (J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D})) + (J^{\#}(\theta^{\#};\mathcal{D}) - J^{\#}(\hat{\theta};\mathcal{D})) \\ &\quad + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2 - \frac{\Delta}{2n}\|\theta^{\text{priv}}\|_2^2\end{aligned}$$

Notice that $J^{\#}(\theta^{\#};\mathcal{D}) - J^{\#}(\hat{\theta};\mathcal{D}) \leq 0$. Using Equation 3.28 we have the following.

$$\hat{\mathcal{L}}(\theta^{\text{priv}};\mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta};\mathcal{D}) \leq \frac{\|b\|_2\|\theta^{\#}-\theta^{\text{priv}}\|_2}{n} + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2$$

The theorem follows from using the tail bounds for $\|b\|_2$ under the distributions ν_1 and ν_2 (see Section 3.5.4.4.1) and Theorem 3.29. □

□

Chapter 4

Case Study: Differentially Private Linear Regression

4.1 Introduction to Private Linear Regression

Consider the linear system in (4.1),

$$y = X\theta^* + w \tag{4.1}$$

where the design matrix $X \in \mathbb{R}^{n\times p}$, output vector $y \in \mathbb{R}^{n\times 1}$, parameter vector $\theta^* \in \mathbb{R}^p$, and $w \in \mathbb{R}^{n\times 1}$ is a noise vector. We define the loss function for any given θ as $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{2n}\sum_{i=1}^{n}(y_i - \langle X_i, \theta\rangle)^2$, where y_i is the i-th entry in the vector y and X_i is the i-th row of the matrix X. The corresponding regression problem for estimating θ^* is in (4.2).

$$\theta^{\#} = \arg\min_{\theta\in\mathcal{C}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Delta}{2n}\|\theta\|_2^2 \tag{4.2}$$

The setting we are interested in is where each row of the design matrix X has L_2 norm at most $\sqrt{p}$ and the parameter vector θ^* has L_∞ norm at most 1. Notice that since we assume θ^* and the rows of X have norm at most $\sqrt{p}$, so truncating y into $[-p, p]$ will not hurt the utility guarantees. Therefore, w.l.o.g. we assume that $y \in [-p, p]$. Also, since θ^* is assumed to have L_∞ norm at most 1, we assume that the convex set over which the optimization is performed has L_∞ norm at most 1, i.e., $\mathcal{C} = \{\theta \in \mathbb{R}^p : \|\theta\|_\infty \le 1\}$.

In order to use the results from Section 3.5.1, we want to bound the gradient and hessian of the loss function $\hat{\mathcal{L}}$. Under the above setting, we bound the gradient of $\frac{1}{2}(y_i - \langle X_i, \theta\rangle)^2$ by ζ for any $\theta \in \mathcal{C}$. It is easy to see that the gradient is $-X_i^T(y_i - \langle X_i, \theta\rangle)$. Therefore, under the choice of parameters in our setting, we have $\zeta = 2p^{3/2}$.

Similarly, to bound the maximum eigenvalue of $\frac{1}{2}\bigtriangledown^2 (y_i - \langle X_i, \theta\rangle)^2$ by λ, we first notice that the hessian is $X_i^T X_i$. Since $\|X_i\|_2 \le \sqrt{p}$, the maximum eigenvalue of the matrix is p. Hence, we can set $\lambda = p$.

In terms of privacy, we want to solve the regression problem in (4.2) while guaranteeing (ϵ, δ)-differential privacy to the data set $\mathcal{D} = (y, X)$. In this chapter we study

two different algorithms for differentially private linear regression. The first one is an instantiation of the objective perturbation algorithm from Chapter 3, and the second one is based on a new idea of data dependent regularization. We will do a comparative study of both these algorithms.

Organization of the chapter: In Section 4.2 we discuss the implications of the utility guarantees from Chapter 3 in the context of linear regression. In Section 4.3 we tighten the utility guarantee (compared to the ones from Chapter 3) via a quadratic formulation of the linear regression objective function. In Section 4.4 we provide a new modification to the objective perturbation algorithm where the L_2-regularization parameter is now dependent on the data set. This modification results in a quadratic improvement over the utility guarantee in Section 4.4. Finally, in Section 4.5, we compare our algorithms for private linear regression to another algorithm for private linear regression existent in the literature. This algorithm is based on the Propose Test and Release framework of [Dwork and Lei, 2009].

4.2 Implications of Utility Guarantees for Algorithm Obj-Pert from Section 3.5

In Chapter 3 we designed the constrained objective perturbation algorithm (Algorithm 4.1 (Algorithm Obj-Pert)) for empirical risk minimization problems with differential privacy guarantees. We provided three different utility guarantees depending on the noise model used and the properties of the data. Theorem 3.20 (Part 1) referred to the utility guarantee with Gamma noise distribution (where the corresponding version of Algorithm Obj-Pert satisfied ϵ-differential privacy), Theorem 3.20 (Part 2) referred to the utility guarantee with Gaussian noise distribution (where the corresponding version of Algorithm Obj-Pert satisfied (ϵ, δ)-differential privacy), and Theorem 3.28 for data sets $\mathcal{D}$ whose corresponding loss functions $\hat{\mathcal{L}}(\theta; \mathcal{D})$ have the least eigenvalue of their hessians bounded from below.

Under the setting of linear regression in Section 4.1, we obtain the following empirical risk bounds (Table 4.1) corresponding to various utility guarantees for Algorithm Obj-Pert .

Section	Theorem	Empirical risk (ignoring ϵ and δ)
Section 3.5.3	Theorem 3.20 (Part 1)	$\tilde{O}(p^3/n)$
Section 3.5.3	Theorem 3.20 (Part 2)	$\tilde{O}(p^{5/2}/n)$
Section 3.5.4.3	Theorem 3.28	$\tilde{O}(p^2/n)$

Table 4.1. Empirical risk bounds for linear regression in the "small p, large n" regime

4.3 Tighter Utility Analysis via Quadratic Form

In this section we provide even tighter utility analysis (under stronger assumptions) compared to the analysis in Section 3.5.4.3 for Algorithm Obj-Pert (Algorithm 4.1). We

would like to mention that the current utility analysis is specific to the problem of linear regression and it is an interesting problem to extend it to other empirical risk minimization problems. In the following section we first revisit the utility analysis from Section 3.5.4.3.

4.3.1 Background: Private Linear Regression using Algorithm 4.1 and Utility Guarantees from Section 3.5.4.3

In this section we revisit Algorithm Obj-Pert (specific to linear regression) and state the privacy and utility guarantees. We state the utility guarantee under the assumption that the hessian X^TX of the loss function has minimum eigenvalue of $\eta = \Omega(n)$. Later in the sparse regression setting (Chapter 6) we discuss that such an assumption is commonly made in the literature to ensure consistent sparse estimation. Moreover, if X is a random Gaussian matrix, then it satisfies the eigenvalue lower bound with high probability.

Algorithm 4.1 Differentially Private Linear Regression via Objective Perturbation

Require: data set $\mathcal{D} = (y, X)$ (with $y \in \mathbb{R}^p$ and $X \in \mathbb{R}^{n\times p}$), privacy parameters ϵ and δ, L_2-regularization parameter $\Delta \geq \frac{2p}{\epsilon}$, a convex set $\mathcal{C} = \{\theta : \|\theta\|_\infty \leq 1\}$.

1: Sample $b \in \mathbb{R}^p$ from $\nu(b; \epsilon, \delta) = \mathcal{N}\left(0,\ \frac{p^3(32\log\frac{2}{\delta}+16\epsilon)}{\epsilon^2}\mathbb{I}_p\right)$.

2: **return** $\theta^{\text{priv}} = \arg\min_{\theta\in\mathcal{C}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Delta}{2n}\|\theta\|_2^2 + \frac{\langle b,\theta\rangle}{n}$.

Theorem 4.1 (Privacy guarantee (from Chapter 3))**.** *Algorithm 4.1 is* (ϵ, δ) *differentially private.*

Theorem 4.2 (Utility guarantee (from Chapter 3))**.** *If the minimum eigenvalue of* X^TX *is at least* η*, then*

$$\mathbb{E}_b\left[\hat{\mathcal{L}}(\theta^{priv}; \mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta}; \mathcal{D})\right] = O\left(\frac{p^4\log(1/\delta)}{n\epsilon^2(\Delta+\eta)} + \frac{\Delta}{n}\|\hat{\theta}\|_2^2\right)$$

Setting $\Delta = \Theta\left(\frac{p}{\epsilon}\right)$ and assuming $\eta = \Omega(n)$ we have the error scale as $\tilde{O}\left(\frac{p^2}{n\epsilon}\right)$. Now consider the following spread of the eigenvalues of X^TX: the smallest eigenvalue of X^TX is a constant and all other eigenvalues are $\Omega(n)$. In such a situation, setting $\Delta = \left(\frac{p^{3/2}}{\epsilon}\right)$ optimally, we get the error to scale as $\tilde{O}\left(\frac{p^{2.5}}{n\epsilon^2}\right)$. In this case we obtain a worse convergence rate (in terms of dimensionality p) compared to the first setting. Here the main reason for the worsening of utility guarantee is that Theorem 4.2 ignores the spread of the eigenvalues of X^TX and only considers the minimum eigenvalue. In the next section we provide a tighter utility analysis which takes into account the spread of the eigenvalues.

4.3.2 Tighter utility analysis of Algorithm 4.1

In this section we provide a tighter utility analysis for Algorithm 4.1 under the assumption that the minimizers θ^{priv} (See Algorithm 4.1) and $\theta^{\#} = \arg\min_{\theta\in\mathcal{C}} \frac{1}{2n}\|y-X\theta\|_2^2 + \frac{\Delta}{n}\|\theta\|_2^2$

lie in the interior of the convex set $\mathcal{C}$. Compared to Theorem 4.2, the new utility analysis bounds the empirical risk in terms of the trace of the matrix $(X^TX + \Delta\mathbb{I}_p)^{-1}$ instead of its minimum eigenvalue. Later we discuss the advantages of this error bound over the one provided in Theorem 4.2 in different scenarios.

We obtain the tighter utility guarantee via a different proof technique (compared to the ones in Chapter 3). Here we view the objective function as a quadratic form (Equation 4.3 below) and first obtain a closed form expression for θ^{priv}. Then we plug in this value back into to the quadratic form and finally bound the expected empirical risk in terms the expected value of $b^T(X^TX + \Delta\mathbb{I}_p)^{-1}b$, where b is the noise vector.

Theorem 4.3 (Tighter utility guarantee)**.** *If the eigenvalues of X^TX are as $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_p$ and the minimizer $\theta^{\#} = \arg\min_{\theta\in\mathcal{C}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Delta}{n}\|\theta\|_2^2$ and θ^{priv} are in the interior of the convex set $\mathcal{C}$, then*

$$\mathbb{E}_b\left[\hat{\mathcal{L}}(\theta^{priv};\mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta};\mathcal{D})\right] = O\left(\frac{p^3\log(1/\delta)}{n\epsilon^2}\sum_{i=1}^{p}\left(\frac{1}{\Delta+\lambda_i}\right) + \frac{\Delta}{n}\|\hat{\theta}\|_2^2\right)$$

Setting $\Delta = \Theta\left(\frac{p}{\epsilon}\right)$ and assuming $\lambda_1 = \Omega(n)$ we can bound the error as $\tilde{O}\left(\frac{p^2}{n\epsilon}\right)$ (which is exactly the bound we obtained in the previous section). Now consider the following spread of the eigenvalues of X^TX (discussed in the previous section): the smallest eigenvalue of X^TX is a constant and all other eigenvalues are $\eta = \Omega(n)$. In such a situation, setting $\Delta = \left(\frac{p}{\epsilon}\right)$ optimally, we get the error to scale as $\tilde{O}\left(\frac{p^2}{n\epsilon^2}\right)$. Notice that the bound we obtain here is $\sqrt{p}$-times better than the one obtained via Theorem 4.2 in the previous section. The tighter error bound is due to the fact that Theorem 4.3 takes explicitly into account the spread of the eigenvalues of X^TX. It is not clear if the analysis of Theorem 4.2 can be directly extended to obtain guarantees of the form of Theorem 4.3.

Proof of Theorem 4.3. Since $\theta^{\#}$ is the unconstrained minimizer of $J^{\#}(\theta;\mathcal{D})$, we can write the objective function $J^{\text{priv}}(\theta;\mathcal{D})$ in the following quadratic form.

$$J^{\text{priv}}(\theta;\mathcal{D}) = \frac{1}{2n}(\theta-\theta^{\#})^T(X^TX + \Delta\mathbb{I}_p)(\theta-\theta^{\#}) + J^{\#}(\theta^{\#};\mathcal{D}) + \frac{\langle b,\theta\rangle}{n} \tag{4.3}$$

In the above expression let us represent $A = X^TX + \Delta\mathbb{I}_p$. Since $\hat{\theta}$ is the unconstrained minimizer of $J^{\text{priv}}(\theta;\mathcal{D})$, we have

$$\begin{aligned} A(\theta^{\text{priv}} - \theta^{\#}) + b &= 0 \\ \Leftrightarrow \theta^{\text{priv}} &= \theta^{\#} - A^{-1}b \end{aligned}$$

Similar to Equation 4.3, we can write $J^{\#}$ also in the following quadratic form.

$$J^{\#}(\theta;\mathcal{D}) = \frac{1}{2n}(\theta-\theta^{\#})^T(X^TX + \Delta\mathbb{I}_p)(\theta-\theta^{\#}) + J^{\#}(\theta^{\#};\mathcal{D})$$

Plugging in the value of θ^{priv} in the above expression we have

$$\begin{aligned} J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D}) &= \frac{1}{2n}(\theta^{\#} - A^{-1}b - \theta^{\#})^T A(\theta^{\#} - A^{-1}b - \theta^{\#}) \\ &= \frac{1}{2n} b^T (A^{-1}) b \end{aligned}$$

Therefore, from the above expression, we have
$\mathbb{E}_b[J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D})] = \frac{1}{2n}\mathbb{E}[b^T(A^{-1})b]$, where $b \sim \mathcal{N}\left(0,\ \frac{p^3(32\log\frac{2}{\delta}+16\epsilon)}{\epsilon^2}\mathbb{I}_p\right)$. By the property of Gaussian random vectors, we have the following. Here we have $\sigma^2 = \frac{p^3(32\log\frac{2}{\delta}+16\epsilon)}{\epsilon^2}$.

$$\begin{aligned} \mathbb{E}_b[J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D})] &= \frac{1}{2n}\mathbb{E}[b^T(A^{-1})b] \\ &= \frac{\sigma^2}{2n} tr(A^{-1}) \end{aligned} \tag{4.4}$$

In order to complete the proof, we need to bound $\mathbb{E}_b[\hat{\mathcal{L}}(\theta^{\text{priv}};\mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta};\mathcal{D})]$. We have,

$$\begin{aligned} \mathbb{E}_b[\hat{\mathcal{L}}(\theta^{\text{priv}};\mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta};\mathcal{D})] &\leq \mathbb{E}_b[J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D})] + \mathbb{E}_b[J^{\#}(\theta^{\#};\mathcal{D}) - J^{\#}(\hat{\theta};\mathcal{D})] \\ &\quad + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2 - \frac{\Delta}{2n}\|\theta^{\text{priv}}\|_2^2 \\ &\quad (\text{where } J^{\#}(\theta^{\#};\mathcal{D}) - J^{\#}(\hat{\theta};\mathcal{D}) \text{ is always} \leq 0) \\ &\leq \mathbb{E}_b[J^{\#}(\theta^{\text{priv}};\mathcal{D}) - J^{\#}(\theta^{\#};\mathcal{D})] + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2 \\ &\leq \frac{\sigma^2}{2n} tr(A^{-1}) + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2 \end{aligned}$$

The last inequality follows from Equation 4.4. Plugging in the value of σ and expanding $tr(A^{-1})$ in terms of the eigenvalues we get the required bound. □

4.4 New Algorithm: Objective Perturbation with Data-dependent Regularization

In the previous section the utility guarantees (Theorems 4.2 and 4.3) were best if the L_2-regularization Δ was fixed to its minimum permissible value allowed (due to (ϵ, δ)-differential privacy guarantee) (i.e. $\Delta = 2p/\epsilon$). In fact, the utility guarantees improve as Δ become smaller. In this section we provide an algorithm with data dependent regularization (i.e., the L_2 regularization depends on the eigenvalues of X^TX, where X is the design matrix). This new way of regularizing the objective function ensures that if the loss function for linear regression is sufficiently strongly convex (i.e., the minimum eigenvalue of X^TX is at least $2p/\epsilon$), then the regularization term is zero and hence we get better utility guarantee. In case the minimum eigenvalue bound on X^TX does not hold, then the utility guarantee is same (up to constant factors) as that of Theorem 4.3 as long as we assume the privacy parameter ϵ to be a small constant. The details of the

algorithm is given in Algorithm 4.2.

Algorithm 4.2 Differentially Private Linear Regression with data-dependent regularization

Require: data set $\mathcal{D} = (y, X)$ (with $y \in \mathbb{R}^p$ and $X \in \mathbb{R}^{n \times p}$), privacy parameters ϵ and δ, L_2-regularization parameter $\Delta = \frac{2p}{\epsilon}$, a convex set $\mathcal{C} = \{\theta : \|\theta\|_\infty \leq 1\}$.

1: Let $U\Sigma U^T$ be the eigen-decomposition of $X^T X$, where $\forall i \in [p], \Sigma(i)$ is the i-th eigenvalue of Σ.

2: Let $\hat{\Sigma}$ be a diagonal matrix with $\hat{\Sigma}(i) = \max(0, \Delta - \Sigma(i))$ for all $i \in [p]$.

3: Sample $b \in \mathbb{R}^p$ from $\nu(b; \epsilon, \delta) = \mathcal{N}\left(0,\ \frac{p^3(512 \log \frac{2}{\delta} + 256\epsilon)}{\epsilon^4} \mathbb{I}_p\right)$.

4: **return** $\theta^{\text{priv}} = \arg\min_{\theta \in \mathcal{C}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\theta^T \hat{\Sigma} \theta}{2n} + \frac{\langle b, \theta \rangle}{n}$.

Privacy analysis: Since in Algorithm 4.2 the regularization $\theta^T \hat{\Sigma} \theta$ is dependent on the design matrix $X^T X$, the proof of Theorem 4.1 does not directly hold anymore for the privacy theorem (Theorem 4.4 below). In our privacy analysis where we first bound the additive change in the gradient and multiplicative change in the hessian of the objective function in Algorithm 4.2 at any $\theta \in \mathcal{C}$ when the underlying data set $\mathcal{D}$ is changed by one entry. Then we invoke a modified version of Theorem 4.1 which ensures that bounding the change in the gradient and the hessian is enough for privacy.

Theorem 4.4 (Privacy Guarantee)**.** *Algorithm 4.2 is (ϵ, δ)-differentially private.*

Proof. The proof of privacy closely follows the proof of Theorem 4.1 from Chapter 3. For the ease of notation, let us denote $\tilde{J}^{\text{priv}}(\theta; \mathcal{D}) = \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\theta^T \hat{\Sigma} \theta}{2n} + \frac{\langle b, \theta \rangle}{n}$ and $\tilde{J}^{\#}(\theta; \mathcal{D}) = \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\theta^T \hat{\Sigma} \theta}{2n}$. In order to use the proof idea from Theorem 4.1, we need to bound the following two terms for any two data sets $\mathcal{D} = (y, X)$ and $\mathcal{D}' = (y', X')$ (where w.l.o.g. we assume that $\mathcal{D}'$ has one entry (y_{new}, x_{new}) more than $\mathcal{D}$).

1. **Change in hessian:** For any given $\theta \in \mathcal{C}$, bound $\frac{det(\bigtriangledown^2 \tilde{J}^{\#}(\theta; \mathcal{D}'))}{det(\bigtriangledown^2 \tilde{J}^{\#}(\theta; \mathcal{D}))}$.

2. **Change in gradient:** For any given $\theta \in \mathcal{C}$, bound $|\bigtriangledown \tilde{J}^{\#}(\theta; \mathcal{D}') - \bigtriangledown \tilde{J}^{\#}(\theta; \mathcal{D})|$.

We bound the change in the hessian via the following lemma.

Lemma 4.5. *For any $\theta \in \mathcal{C}$ and $\Delta = \frac{2p}{\epsilon}$, $\frac{det(\bigtriangledown^2 \tilde{J}^{\#}(\theta; \mathcal{D}'))}{det(\bigtriangledown^2 \tilde{J}^{\#}(\theta; \mathcal{D}))}$ is bounded by $e^{\epsilon/2}$.*

Proof. For a data set $\mathcal{D} = (y, X)$, the hessian $\bigtriangledown^2 \tilde{J}^{\#}(\theta; \mathcal{D}) = X^T X + \hat{\Sigma}$. For the data set $\mathcal{D}'$ which has one entry more than $\mathcal{D}$, we represent the hessian $\bigtriangledown^2 \tilde{J}^{\#}(\theta; \mathcal{D}') = X^T X + x_{new} x_{new}^T + \hat{\Sigma}'$, where Σ' is the new regularization matrix (for data set $\mathcal{D}'$) in Algorithm 4.2. Let $\lambda_1' \leq \cdots \leq \lambda_p'$ be the eigenvalues of $X'^T X'$ and $\lambda_1 \leq \cdots \leq \lambda_p$ be the eigenvalues of $X^T X$. Since A is a positive semi-definite matrix, by Weyl's inequality for all $i \in [p]$, $\lambda_i' \geq \lambda_i$ Additionally, since the maximum eigenvalue of $x_{new} x_{new}^T$ is at most p (because

$\|x_{new}\|_2 \leq \sqrt{p}$), it follows that for all $i \in [p]$, $\lambda'_i - \lambda_i \leq p$. Now,

$$\begin{aligned}\frac{det(\bigtriangledown^2 \tilde{J}^{\#}(\theta;\mathcal{D}'))}{det(\bigtriangledown^2 \tilde{J}^{\#}(\theta;\mathcal{D}))} &= \frac{det(X'^T X' + \hat{\Sigma}')}{det(X^T X + \hat{\Sigma})} \\ &= \frac{\prod \max(\lambda'_i, \Delta)}{\prod \max(\lambda_i, \Delta)} = \prod \frac{\max(\lambda'_i, \Delta)}{\max(\lambda_i, \Delta)} \\ &= \prod \left(1 + \frac{\max(\lambda'_i, \Delta) - \max(\lambda_i, \Delta)}{\max(\lambda_i, \Delta)}\right)\end{aligned}$$

$$\begin{aligned}&\leq \prod \left(1 + \frac{\lambda'_i - \lambda_i}{\max(\lambda_i, \Delta)}\right) \leq \prod \left(1 + \frac{\lambda'_i - \lambda_i}{\Delta}\right) \\ &\text{(Follows from the fact that } \lambda'_i \geq \lambda_i \text{ for all } i) \\ &= 1 + \frac{1}{\Delta}\sum_i (\lambda'_i - \lambda_i) + \frac{1}{\Delta^2}\sum_{i \neq j} (\lambda'_i - \lambda_i)(\lambda'_j - \lambda_j) \\ &\leq 1 + \frac{p}{\Delta} + \frac{p^2}{\Delta^2} + \cdots\end{aligned}$$

Since $\Delta = 2p/\epsilon$ and $\epsilon < 1$,we have the above expression upper bounded by $\frac{\Delta}{\Delta - p}$. In order to bound $\frac{\Delta}{\Delta - p}$ to at most $e^{\epsilon/2}$, we need $\Delta \geq \frac{p}{1 - e^{-\epsilon/2}} \geq \frac{2p}{\epsilon}$. Now with the choice of $\Delta = 2p/\epsilon$ completes the proof. □

We bound the change in the gradient via the following lemma.

Lemma 4.6. *For any given $\theta \in \mathcal{C}$ and $\Delta = \frac{2p}{\epsilon}$, $\| \bigtriangledown \tilde{J}^{\#}(\theta;\mathcal{D}') - \bigtriangledown \tilde{J}^{\#}(\theta;\mathcal{D})\|_2$ is at most $\frac{4p^{3/2}}{\epsilon}$.*

Proof. We have the following.

$$\begin{aligned}| \bigtriangledown \tilde{J}^{\#}(\theta;\mathcal{D}') - \bigtriangledown \tilde{J}^{\#}(\theta;\mathcal{D})| &\leq \|(X'^T X' - X^T X) + (\hat{\Sigma}' - \hat{\Sigma})\|_2 \|\theta\|_2 + \|X'^T y' - X^T y\|_2 \\ &\leq \sqrt{p} \underbrace{\|(X'^T X' - X^T X) + (\hat{\Sigma}' - \hat{\Sigma})\|_2}_{A} + \underbrace{\|X'^T y' - X^T y\|_2}_{B}\end{aligned} \tag{4.5}$$

We bound the terms A and B individually. First to bound term B, notice $B = \|y_{new} x_{new}\|_2$. Now, by assumption $y_{new} \in [-p, p]$ and $\|x_{new}\|_2 \leq \sqrt{p}$. Therefore, B is upper bounded by $p^{3/2}$. In the following we upper bound the term A by $4p/\epsilon$.

Notice that $A \leq \|X'^T X' - X^T X\|_2 + \|\hat{\Sigma}' - \hat{\Sigma}\|_2$. To bound the spectral norm of $X'^T X' - X^T X$, it suffices to show that for any unit vector z, $|z^T (X'^T X' - X^T X) z| \leq p$. Now, $|z^T (X'^T X' - X^T X) z| = |z^T x_{new} x_{new}^T z| = \langle z, x_{new} \rangle^2 \leq p$. We will use the same technique for $\hat{\Sigma}' - \hat{\Sigma}$. Recall that both $\hat{\Sigma}$ and $\hat{\Sigma}'$ are positive semi-definite matrices. Therefore,

$$|z^T (\hat{\Sigma}' - \hat{\Sigma}) z| \leq \max\{z^T \hat{\Sigma} z, z^T \hat{\Sigma}' z\} \leq \Delta = \frac{2p}{\epsilon}$$

Therefore, $A \leq 3p/\epsilon$. Plugging in the values of A and B in Equation 4.5 we get the required bound. □

With Lemmas 4.5 and 4.6 in hand, we now invoke the following (modified) privacy theorem from Chapter 3 to conclude the proof of Theorem 4.4.

Theorem 4.7 (Private Objective Perturbation (Modified Theorem 4.1))**.** *Let $\mathcal{C}$ be a closed convex subset of $\mathbb{R}^p$. Let $\mathcal{D} = \{d_1, \ldots, d_n\}$ be a data set, let $\mathcal{L}(\theta; \mathcal{D})$ be a convex function with continuous hessian. For any data set $\mathcal{D}'$ differing in one entry from $\mathcal{D}$ (i.e, has one entry more or less compared to $\mathcal{D}$), let ζ be the upper bound on $\| \bigtriangledown \mathcal{L}(\theta; \mathcal{D}) - \bigtriangledown \mathcal{L}(\theta; \mathcal{D}')\|_2$ and let $e^{\epsilon/2}$ and $e^{-\epsilon/2}$ be an upper and lower bound on the ratio between the determinants of $\bigtriangledown^2 \mathcal{L}(\theta; \mathcal{D})$ and $\bigtriangledown^2 \mathcal{L}(\theta; \mathcal{D}')$ (for all $\theta \in \mathcal{C}$). Then, with the noise vector $b \sim \mathcal{N}\left(0, \frac{\zeta^2(8\log(2/\delta)+4\epsilon)}{\epsilon^2}\mathbb{I}_p\right)$, the minimizer $\theta^{priv} = \arg\min_{\theta \in \mathcal{C}} \mathcal{L}(\theta; \mathcal{D}) + \langle b, \theta \rangle$ is (ϵ, δ)-differentially private.*

□

Utility analysis: In this section we show that if the minimum eigenvalue of $X^T X$ is sufficiently large (i.e., at least $\Omega(p/\epsilon)$), then the error does not have dependence on the regularization parameter. As a result we get a quadratic improvement in the utility guarantee when then least eigenvalue of $X^T X$ is at least $\Omega(n)$. It is important to note that such a lower bound on the eigenvalue of $X^T X$ hold with high probability when X is a random Gaussian matrix. In case the minimum eigenvalue of $X^T X$ is $o(n)$, then utility guarantee of Theorem 4.8 is same as that of Theorem 4.3 in Section 4.3 (up to $1/\epsilon^2$ factors). It is not clear if the additional $1/\epsilon^2$ factor is necessary. A tighter utility analysis can possibly remove this factor.

Theorem 4.8 (Utility Guarantee)**.** *If the eigenvalues of $X^T X$ are $\frac{p}{\epsilon} \leq \lambda_1 \leq \lambda_2 \cdots \leq \lambda_p$ and the and the minimizers $\theta^{\#} = \arg\min_{\theta \in \mathcal{C}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\theta^T \hat{\Sigma} \theta}{n}$ and θ^{priv} are in the interior of the convex set $\mathcal{C}$, then*

$$\mathbb{E}_b\left[\hat{\mathcal{L}}(\theta^{priv}; \mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta}; \mathcal{D})\right] = O\left(\frac{p^3 \log(1/\delta)}{n\epsilon^4} \sum_{i=1}^{p} \left(\frac{1}{\lambda_i}\right)\right)$$

If the eigenvalues are lower bounded by zero (i.e, $0 \leq \lambda_1 \leq \lambda_2 \cdots \leq \lambda_p$), then

$$\mathbb{E}_b\left[\hat{\mathcal{L}}(\theta^{priv}; \mathcal{D}) - \hat{\mathcal{L}}(\hat{\theta}; \mathcal{D})\right] = O\left(\frac{p^3 \log(1/\delta)}{n\epsilon^4} \sum_{i=1}^{p} \left(\frac{1}{\max(\lambda_i, \Delta)}\right) + \frac{\Delta}{2n}\|\hat{\theta}\|_2^2\right)$$

The proof Theorem 4.8 is similar to Theorem 4.3 (in Section 4.3.2) and hence omitted here. Assuming $\lambda_1 = \Omega(n)$ we can bound the error as $\tilde{O}\left(\left(\frac{p^2}{n\epsilon^2}\right)^2\right)$. Compared to the bound we obtained in Sections 4.3.1 and 4.3.2, the current error bound is better by a quadratic factor (except an additional $1/\epsilon^2$ multiplicative dependence).

4.5 Comparison to Propose-Test-Release (PTR) based Linear Regression

[Dwork and Lei, 2009] proposed a general framework called Propose-Test-Release (PTR) framework for designing differentially private algorithms for various statistical estimators. At a high level, for a given function $f : \mathcal{T}^* \to \mathbb{R}^p$ (where $\mathcal{T}^*$ is the domain of the data sets) and a given data set $\mathcal{D}$, PTR framework first computes (while preserving differential privacy) the number of entries in $\mathcal{D}$ that needs to be changed so that $f(\mathcal{D})$ changes significantly. If the answer is small (i.e., $poly \log n$, where n is the size of the data set $\mathcal{D}$), then it adds "small" amount of Laplace noise to $f(\mathcal{D})$ and outputs, otherwise it outputs a $\perp$. In terms of privacy, PTR framework is always guaranteed to produce an output which is (ϵ, δ)-differentially private. [Dwork and Lei, 2009] analyzed the utility guarantee of PTR framework in the context of linear regression. They showed that as long as the dimensionality of the problem (p) in (4.1) is constant, if the rows of X are drawn i.i.d. from some underlying distribution, and the noise vector w is independent of X and has i.i.d. entries from some (symmetric, mean zero) distribution, then the linear regression algorithm under the PTR framework outputs a θ^{priv} which converges in distribution to θ^* in (4.1) as the size (n) of the data set $\mathcal{D} = (y, X)$ goes to infinity.

In contrast, in our work, under the assumption that each row of the design matrix X has norm at most $\sqrt{p}$, the response vector y is in $[-p, p]^n$, and $\|\theta^*\|_\infty \leq 1$, we achieve a convergence rate of the L_2 distance between the estimated parameter θ^{priv} and true parameter θ^* as $\tilde{O}\left(\frac{p^{2.5}}{n}\right)$. Compared to convergence guarantee obtained by PTR framework, our bounds are sharper (by an exponential factor in the dimensionality). Also notice that for convergence, as long as p grows slower than $n^{2/5}$, θ^{priv} converges to θ^*. So, in particular, in our algorithm we do not need to restrict the dimensionality p to be a constant.

Chapter 5

Differentially Private Online Learning

5.1 Introduction

With the continuous increase in the amount of computational resources available, modern websites and online systems can now, in real time, process large amounts of potentially sensitive information gathered from their customers. Although in most cases, these websites intend to just leverage real-time learning using their customers' data, it might actually compromise their customers' privacy.

For example, consider the following scenario in the context of sponsored search. Sponsored search advertisements (ads) are served with organic search results and form a major source of revenue for search engines. To serve these ads effectively, search engines attempt to learn the relevance of an ad for a user. For this purpose, search engines typically store users' profile information, e.g., gender of the user. Now, suppose a male user clicks an ad, through which the search engine learns the rule "males like this ad". This rule is directly observable by the corresponding advertiser whose ad was clicked. To do this, the advertiser makes two profiles, one that reports the gender as male and the other one as female. He now compares the rank of his ad presented by the search engine to each of his profile and observes that the ad is presented at the top of other ads for the male profile. Also, the advertiser can obtain the IP address of the user, as the user clicked the ad. Thus, he can make a direct association between the user and his gender, compromising the user's privacy. Similar examples can be constructed for several other online learning domains such as, *online portfolio management* [Kalai and Vempala, 2005], *online linear prediction* [Hazan et al., 2007] etc.

In this chapter, we address privacy concerns in online learning scenarios similar to the examples mentioned above. Specifically, we provide a generic framework for privacy preserving online learning. We use *differential privacy* [Dwork et al., 2006b] as the formal notion of privacy, and use *online convex programming* (OCP) [Zinkevich, 2003] as the formal online learning model. Note that OCP is a popular online learning paradigm and includes several online learning problems faced by real-life systems. Examples include

online logistic regression, online linear regression etc.

Most of the existing results in differentially private learning have focused on the offline setting only, where all the training data is available beforehand. Hence, both privacy and utility need to be argued only over one final output. In contrast, in the online learning setting, data arrives online[1] (e.g. user queries and clicks) and the algorithm has to provide an output (e.g. relevant ads) at each step. Hence, the number of outputs produced is the same as the size of the entire dataset. To guarantee differential privacy, one has to analyze the privacy of the *complete* sequence of outputs produced, thereby making privacy preservation a significantly harder problem. For utility, we need to show that asymptotically the algorithm is at least as good as the optimal offline solution, i.e., the algorithm has sub-linear *regret*.

In this work, we study both privacy and utility aspects of privacy preserving online learning in the online convex programming (OCP) model. The goal is to provide differentially private OCP algorithms with sub-linear regret. To this end, we provide a generic framework to convert any OCP algorithm into a differentially private algorithm with sub-linear regret, provided that the algorithm satisfies two criteria: a) linearly decreasing sensitivity (see Definition 5.3), b) sub-linear regret.

Next, we instantiate our generic framework with two popular OCP algorithms: Implicit Gradient Descent (IGD) by [Crammer et al., 2006; Kulis and Bartlett, 2010] and Generalized Infinitesimal Gradient Ascent (GIGA) by [Zinkevich, 2003]. Our algorithms guarantee differential privacy as well as $\tilde{O}(\sqrt{T})$ regret for a fairly general class of strongly convex functions with Lipschitz continuous gradients. In fact, we show that IGD can be used with our framework for non-differentiable functions as well.

Finally, our generic framework can be used to obtain privacy preserving algorithms for a large class of *offline* learning problems as well. In particular, we show that our private OCP framework can be used to obtain generalization error bounds for various offline learning problems using techniques of [Kakade and Tewari, 2008] (Section 5.8). Our differentially private offline learning framework provide better error bounds and is more practical than the output perturbation methods of [Chaudhuri et al., 2011; Rubinstein et al., 2009] (Section 3.2.2). Moreover, the assumptions for the privacy guarantees of the algorithms in this chapter seem to be applicable in more practical scenarios compared to the algorithms in Chapter 3.

Organization of the chapter: In Section 5.2 we review some of the earlier works in private online learning and differentially private learning in general. In Section 5.3 we summarize our contributions in this chapter. In Section 5.5 we introduce our differentially private online convex programming framework (POCP) and, in Sections 5.6 and 5.7 we instantiate the POCP framework two specific online learning algorithms (*implicit gradient descent* and *generalized infinitesimal gradient ascent* respectively). In Section 5.8 we use the POCP framework for the standard empirical risk minimization problem considered in Chapter 3.

[1] At each time step one data entry arrives.

5.2 Summary of Previous Work

Recently, several private algorithms have been developed for learning problems [Blum et al., 2008; Chaudhuri et al., 2011; Williams and McSherry, 2010; Pathak et al., 2010; Rubinstein et al., 2009]. Among these, the works by [Chaudhuri et al., 2011; Rubinstein et al., 2009; Williams and McSherry, 2010] are the most related as they consider a large class of offline learning problems that can be written as regularized empirical risk minimization (ERM) problems with convex loss functions. In particular, [Chaudhuri et al., 2011; Rubinstein et al., 2009] proposed a differentially private method that ensures privacy by either adding noise to the optima (output perturbation (Section 3.2.2)) of the corresponding ERM or by perturbing the ERM itself (objective perturbation (Section 3.2.3)). In both these cases, the privacy guarantees depend on the promise that the *exact* minimum to the underlying optimization problem is obtained, which is unlikely for several practical problems. In contrast, [Williams and McSherry, 2010] proposed a *noisy* gradient descent method to optimize ERM. Although their method maintains differential privacy at each gradient descent step, it fails to provide any convergence guarantees. Interestingly, our online learning techniques can be applied to this offline learning problem as well. In fact, our method provides better error bounds and is more practical than the existing methods (see Section 5.8).

As mentioned earlier, most of the existing work in differentially private learning has been in the offline setting. One notable exception is the work of [Dwork et al., 2010a], that considers the problem of preserving privacy in the experts setting. In particular, they provide a differentially private algorithm for experts framework that has $\tilde{O}(\sqrt{T})$ regret. However, their results are restricted to the experts setting only, and it is not clear how their techniques can be generalized to the general class of OCP problems.

In a related line of work, there have been a few results that use online learning techniques to obtain differentially private algorithms [Hardt and Rothblum, 2010; Dwork et al., 2010b; Gupta et al., 2012]. In particular, [Hardt and Rothblum, 2010; Gupta et al., 2012] used the experts framework to obtain a differentially private algorithm for answering *adaptive* counting queries on a dataset. We stress that although these methods use online learning techniques, they are designed to handle offline problems only, where the dataset is fixed and is known in advance.

5.3 Overview of Our Contributions

In the following we summarize our main contributions in this chapter.

1. We formalize the problem of differentially private OCP, and provide a generic framework for the same with provable privacy and utility (regret) guarantees. (see Section 5.5).

2. We instantiate our framework with two popular OCP algorithms: Implicit Gradient Descent (IGD) and Generalized Infinitesimal Gradient Ascent (GIGA). For both the algorithms we provide privacy guarantees and $\tilde{O}(\sqrt{T})$ regret. To guarantee privacy, we need to show that the effect of any data entry on the output of any of the algorithms (at time step t) *decreases linearly in* t. This stability bound is of

independent interest and has implications for connections between online learning and stability. (See [Ross and Bagnell, 2011; Poggio et al., 2011] for more discussions on this topic.)

3. In Section 5.8 we show that our differentially private framework for OCP can be used to solve a large class of offline learning problems as well, for which our method provides better guarantees than the existing state-of-the-art results ([Chaudhuri et al., 2011] and [Rubinstein et al., 2009]).

5.4 Preliminaries

5.4.1 Online Convex Programming

Online convex programming (OCP) is one of the most popular and powerful paradigms in the online learning setting. OCP can be thought of as a game between a player and an adversary. At each step t, the player selects a point $\theta_t \in \mathbb{R}^p$ from a convex set $\mathcal{C}$. Then, the adversary selects a convex cost function $f_t : \mathbb{R}^p \to \mathbb{R}$ and the player has to pay a cost of $f_t(\theta_t)$. An OCP algorithm $\mathcal{A}$ maps a function sequence $F = \langle f_1, f_2, \ldots, f_T \rangle$ to a sequence of points $\Theta = \langle \theta_2, \theta_3, \ldots, \theta_{T+1} \rangle \in \mathcal{C}^T$, i.e., $\mathcal{A}(F) = \Theta$. Now, the goal of the player (or the algorithm) is to minimize *regret*, i.e., the total cost incurred when compared to the optimal offline solution θ^* selected in hindsight, i.e., when all the functions have already been provided. Formally,

Definition 5.1 (Regret)**.** *Let $\mathcal{A}$ be an online convex programming algorithm that selects a point $\theta_t \in \mathcal{C}$ at the $(t-1)$-th iteration. Let $f_t : \mathbb{R}^d \to \mathbb{R}$ be a convex cost function served at the t-th iteration. Then, the regret $\mathcal{R}_\mathcal{A}$ of $\mathcal{A}$ over T iterations is given by:* $\mathcal{R}_\mathcal{A}(T) = \sum_{t=1}^{T} f_t(\theta_t) - \min_{\theta^* \in \mathcal{C}} \sum_{t=1}^{T} f_t(\theta^*)$.

Several OCP algorithms exist in the literature that guarantee $O(\sqrt{T})$ regret for bounded Lipschitz functions f_t and $O(\log T)$ regret for *strongly* convex functions f_t [Hazan et al., 2007; Kulis and Bartlett, 2010; Zinkevich, 2003; Kakade and Shalev-Shwartz, 2008].

5.4.2 Differential Privacy

We now formally define the notion of differential privacy in the context of our problem.

Definition 5.2 ((ϵ, δ)-differential privacy [Dwork et al., 2006b,a])**.** *Let $F = \langle f_1, f_2, \ldots, f_T \rangle$ be a sequence of convex functions. Let $\mathcal{A}(F) = \Theta$, where $\Theta = \langle \theta_1, \theta_2, \ldots, \theta_T \rangle \in \mathcal{C}^T$ be T outputs of the OCP algorithm $\mathcal{A}$ when applied to F. A randomized OCP algorithm $\mathcal{A}$ is (ϵ, δ)-differentially private if for any two function sequences F and F' that differ in at most one function entry, and for all $\mathcal{S} \subset \mathcal{C}^T$ the following holds:*

$$\Pr[\mathcal{A}(F) \in \mathcal{S}] \leq e^{\epsilon} \Pr[\mathcal{A}(F') \in \mathcal{S}] + \delta$$

Intuitively, the above definition means that changing any $f_t \in F$, $t \leq T$ to some other function f'_t will not modify the output sequence X by a large amount. If we consider

each f_t to be some information or data point associated with an individual, then the definition above states that the presence or absence of that individual's entry in the data set will not affect the output by too much. Hence, the output of algorithm $\mathcal{A}$ will not reveal any extra information about the individual. Privacy parameters (ϵ, δ) decide the extent to which an individual's entry affects the output; lower values of ϵ and δ imply higher level of privacy.

5.4.3 Notation

$F = \langle f_1, f_2, \ldots, f_T \rangle$ denotes the function sequence given to an OCP algorithm $\mathcal{A}$ and $\mathcal{A}(F) = \Theta$ s.t. $\Theta = \langle \theta_2, \theta_3, \ldots, \theta_{T+1} \rangle \in \mathcal{C}^T$ represents output sequence when $\mathcal{A}$ is applied to F. We denote the subsequence of functions F till the t-th step as $F_t = \langle f_1, \ldots, f_t \rangle$. $\mathcal{C} \subseteq \mathbb{R}^p$, denotes a convex set in d dimensions. Matrices are represented by capital letters (e.g., M). $x^T y$ denotes the inner product between $x \in \mathbb{R}^p$ and $y \in \mathbb{R}^p$. $\|M\|_2$ denotes the spectral norm of matrix M and is the largest eigenvalue of M.

Typically, Ψ is the minimum strong convexity parameter of any $f_t \in F$. Similarly, L is the largest Lipschitz constant of any $f_t \in F$ and L_G is the largest Lipschitz constant of the gradient of any $f_t \in F$. Recall that a function $f : \mathcal{C} \to \mathbb{R}$ is Ψ-strongly convex, if $\forall\ x, y \in \mathcal{C}$ the following holds:

$$f(\gamma x + (1-\gamma)y) \leq \gamma f(x) + (1-\gamma) f(y) - \frac{\Psi \gamma (1-\gamma)}{2} \|x - y\|_2^2, 0 \leq \gamma \leq 1.$$

Also recall that a function f is L-Lipschitz, if $\forall\ x, y \in \mathcal{C}$ the following holds: $|f(x) - f(y)| \leq L\|x - y\|_2$. Function f is L_G-Lipschitz continuous gradient if $\|\bigtriangledown f(x) - \bigtriangledown f(y)\|_2 \leq L_G \|x - y\|_2$, $\forall x, y \in \mathcal{C}$.

At time-step t, non-private and private versions of any OCP algorithm output θ_{t+1} and $\hat{\theta}_{t+1}$, respectively. θ^* denotes the optimal offline solution, i.e.,
$\theta^* = \arg\min_{\theta \in \mathcal{C}} \sum_{t=1}^{T} f_t(\theta)$. $\mathcal{R}_{\mathcal{A}}(T)$ denotes the regret of an OCP algorithm $\mathcal{A}$ when applied for T steps.

5.5 Differentially Private Online Convex Programming

In this section we first present our differentially private framework for solving OCP problems, and provide privacy as well as regret guarantees for our framework. Then, in Section 5.6, we instantiate our framework with the *Implicit Gradient Descent* (IGD) [Kulis and Bartlett, 2010] algorithm and provide regret, privacy guarantees for the same. We also instantiate our framework with the *Generalized Infinitesimal Gradient Ascent* (GIGA) [Zinkevich, 2003] algorithm (Section 5.7).

Recall that a differentially private OCP algorithm should not produce *significantly* different sequences of outputs ($\Theta = \langle \theta_2, \cdots, \theta_{T+1} \rangle$) for input function sequences F and F', where F and F' differ in exactly one cost function. A trivial way to ensure it is by selecting output sequence Θ independently of the input cost functions F. However, such an "algorithm" can have $O(T)$ regret.

To discard such bad solutions, we require a differentially private OCP algorithm

to have both: a) Privacy: (ϵ, δ)-differential privacy, and b) Utility: sub-linear regret. Our generic framework can transform any given OCP algorithm, $\mathcal{A}$, into a differentially private OCP algorithm that satisfies the above given requirements. However, we require $\mathcal{A}$ to have sub-linear regret and low sensitivity, i.e., $\mathcal{A}$ should not be very sensitive to any particular cost function in the input sequence. We now formalize this notion of sensitivity:

Definition 5.3 (L_2-sensitivity [Dwork et al., 2006b; Chaudhuri et al., 2011])**.** *Let F, F' be two function sequences differing in at most one entry, i.e., at most one function is different in the two sequences. Let $\theta_{t+1} = \mathcal{A}(F)_t$ be the t-th output of $\mathcal{A}$ when supplied F, and similarly, $\theta_{t+1} = \mathcal{A}(F')_t$ is the t-th output of $\mathcal{A}$ for input sequence F'. Then sensitivity of the algorithm $\mathcal{A} : F \to \mathcal{C}^T$, at the t-th time-step is given by:*

$$\mathcal{S}(\mathcal{A}, t) = \sup_{F, F'} \|\mathcal{A}(F)_t - \mathcal{A}(F')_t\|_2.$$

Using this definition of sensitivity, we now state the assumptions that $\mathcal{A}$ should satisfy:

- L_2-sensitivity: The L_2-sensitivity of the algorithm $\mathcal{A}$ should decay linearly with time, i.e.,
$$\mathcal{S}(\mathcal{A}, t) \leq \frac{\lambda_{\mathcal{A}}}{t}, \tag{5.1}$$
where $\lambda_{\mathcal{A}} > 0$ is a constant depending only on $\mathcal{A}$, L and α, i.e., the Lipschitz constant and the strong convexity parameter of the functions in F.
- Regret bound $\mathcal{R}_{\mathcal{A}}(T)$: Regret of $\mathcal{A}$ is assumed to be sub-linear in T, i.e.,
$$\mathcal{R}_{\mathcal{A}}(T) = \sum_{t=1}^{T} f_t(\theta_t) - \min_{\theta^* \in \mathcal{C}} \sum_{t=1}^{T} f_t(\theta^*) = o(T). \tag{5.2}$$

A natural question to ask is whether there exists an OCP algorithm $\mathcal{A}$ that satisfies both the conditions above. In Sections 5.6 and 5.7 we show that both IGD and GIGA satisfy these conditions. In fact, recent results by [Ross and Bagnell, 2011; Poggio et al., 2011] seem to indicate a close connection between sensitivity and regret for online learning algorithms . We leave further investigation of the interplay between sensitivity and regret as a topic for future research.

Now, given $\mathcal{A}$ that satisfies both (5.1) and (5.2), we transform it into a private algorithm by perturbing θ_{t+1} (output of $\mathcal{A}$ at t-th step) by a small amount of noise, whose magnitude is fixed by the *noise parameter* β in Algorithm 5.1. Let $\tilde{\theta}_{t+1}$ be the perturbed output which might lie outside the convex set $\mathcal{C}$. As OCP requires each output to lie in $\mathcal{C}$, we project $\tilde{\theta}_{t+1}$ back to $\mathcal{C}$ and output the projection $\hat{\theta}_{t+1}$. Note that our Private OCP (POCP) algorithm also stores the "uncorrupted" iterate θ_{t+1}, which is used in the next step. See Algorithm 5.1 for a pseudo-code of our method.

Now, using the above two assumptions along with concentration bounds for Gaussian noise, we obtain privacy and regret guarantees for our Private OCP algorithm. Using

the sensitivity assumption, it is easy to prove differential privacy for the output of any *fixed* time step. However, we need to provide differential privacy jointly over *all* time steps, which is the main technical novelty of our work. See Sections 5.5.1 and 5.5.2 for a detailed analysis of privacy and regret guarantees, respectively.

Algorithm 5.1 Private OCP Method (POCP)

Input: OCP algorithm $\mathcal{A}$, cost function sequence $F = \langle f_1, \cdots, f_T \rangle$ and the convex set $\mathcal{C}$
Parameter: privacy parameters (ϵ, δ)
Choose θ_1 and $\hat{\theta}_1$ randomly from $\mathcal{C}$
for $t = 1$ to $T - 1$ **do**
  Cost: $L_t(\hat{\theta}_t) = f_t(\hat{\theta}_t)$
  OCP Update: $\theta_{t+1} \leftarrow \mathcal{A}(\langle f_1, \ldots, f_t \rangle, \langle \theta_1, \ldots, \theta_t \rangle, \mathcal{C})$
  Noise Addition: $\tilde{\theta}_{t+1} \leftarrow \theta_{t+1} + b_{t+1}$, $b_{t+1} \sim \mathcal{N}(0^p, \frac{\beta^2}{t^2}\mathbb{I}^p)$,
    where $\beta = \lambda_{\mathcal{A}} T^{0.5+c} \sqrt{\frac{2}{\epsilon}\left(\log\frac{T}{\delta} + \frac{\sqrt{\epsilon}}{T^{0.5+c}}\right)}$ and $c = \frac{\log\frac{1}{2\epsilon}\log(2/\delta)}{2\log T}$
  Output $\hat{\theta}_{t+1} = \operatorname{argmin}_{\theta \in \mathcal{C}} \left(\|\theta - \tilde{\theta}_{t+1}\|_2^2 \right)$
end for

5.5.1 Privacy Analysis for POCP

Under Assumption (5.1), changing one function in the cost function sequence F can lead to a change of at most $\lambda_{\mathcal{A}}/t$ in the t-th output of $\mathcal{A}$. Intuitively, adding a noise of the same order should make the output of Algorithm 5.1 at the t-th step "almost independent" of any fixed cost function and hence, differentially private. We make this idea precise in the following lemma.

Lemma 5.4. *Let $\mathcal{A}$ be an OCP algorithm that satisfies the sensitivity assumption* (5.1) *and let $\lambda_{\mathcal{A}}$ be the sensitivity parameter. Fix the noise parameter in Algorithm 5.1 as $\beta = \lambda_{\mathcal{A}} T^{0.5+c} \sqrt{\frac{2}{\epsilon}\left(\log\frac{T}{\delta} + \frac{\sqrt{\epsilon}}{T^{0.5+c}}\right)}$ for $c > 0$. Then, the output at the t-th step of the Algorithm 5.1, $\hat{\theta}_{t+1}$, is $(\frac{\sqrt{\epsilon}}{T^{0.5+c}}, \frac{\delta}{T})$-differentially private.*

We defer the proof of the above lemma to Section 5.5.3.1. The above lemma shows that the output at each step of Algorithm 5.1 is $(\frac{\sqrt{\epsilon}}{T^{0.5+c}}, \frac{\delta}{T})$-differentially private. Hence, a simple composition argument guarantees $(T^{0.5-c}\sqrt{\epsilon}, \delta)$-differential privacy for all the steps [Dwork and Lei, 2009]. Thus, setting $c = 0.5$ makes the outputs at every time step (ϵ, δ)-differentially private. However, this requires that a noise of variance $\sim O(T/t)$ be added at each step. This means that the noise added to any fixed output is much larger than the effect of incoming function f_t and thus can lead to an arbitrarily high regret.

To avoid this problem and obtain better regret bounds, we exploit the interdependence between the iterates (and outputs) of our algorithm. For this purpose, we use a lemma by [Dwork et al., 2010b, Lemma III.2] that bounds the relative entropy of two random variables in terms of the L_∞ norm of the ratio of their probability densities and

also a proof technique developed by [Dwork et al., 2010b; Hardt and Rothblum, 2010] for the problem of releasing differentially private data sets.

Now we state the privacy guarantee for Algorithm 5.1 over *all* T iterations. For clarity purposes, we defer the proof of the above theorem till Section 5.5.3.2.

Theorem 5.5 (POCP Privacy)**.** *Let $\mathcal{A}$ be an OCP algorithm that satisfies the sensitivity assumption* (5.1) *with sensitivity parameter $\lambda_{\mathcal{A}}$. Then, the* POCP *algorithm (Algorithm 5.1) with the noise parameter $\beta = \lambda_{\mathcal{A}} T^{0.5+c} \sqrt{\frac{2}{\epsilon}\left(\log\frac{T}{\delta} + \frac{\sqrt{\epsilon}}{T^{0.5+c}}\right)}$ is $(3\epsilon, 2\delta)$-differentially private for $c = \frac{\left(\log(\frac{1}{2\epsilon}\log\frac{2}{\delta}\right)}{2\log T}$.*

5.5.2 Utility (Regret) Analysis for POCP

In this section, we provide a generic regret bound analysis for our POCP algorithm (see Algorithm 5.1). The regret bound of POCP depends on the regret $\mathcal{R}_{\mathcal{A}}(T)$ of the non-private OCP algorithm $\mathcal{A}$. For typical OCP algorithms like IGD, GIGA , $\mathcal{R}_{\mathcal{A}}(T) = O(\log T)$, assuming each cost function f_t is strongly convex.

Theorem 5.6 (POCP Regret)**.** *Let $\mathcal{A}$ be an OCP algorithm that satisfies sensitivity assumption* (5.1)*, and let $\lambda_{\mathcal{A}}$ be the corresponding sensitivity parameter. Also, let $\mathcal{R}_{\mathcal{A}}(T)$ be the regret of $\mathcal{A}$ over T-time steps, and $L > 0$ be the maximum Lipschitz constant of any function f_t. Then, the expected regret of our* POCP *algorithm (Algorithm 5.1) satisfies:*

$$\mathbb{E}\left[\sum_{t=1}^{T} f_t(\hat{\theta}_t)\right] - \min_{\theta\in\mathcal{C}} \sum_{t=1}^{T} f_t(\theta) \leq 2\sqrt{p}L(\lambda_{\mathcal{A}} + \|\mathcal{C}\|_2)\sqrt{T}\frac{\log^2\frac{T}{\delta}}{\epsilon} + \mathcal{R}_{\mathcal{A}}(T),$$

where $\theta \in \mathbb{R}^p$ and $\|\mathcal{C}\|_2$ is the diameter of the convex set $\mathcal{C}$.

The above theorem shows that in expectation, POCP($\mathcal{A}$) algorithm has an additional regret of $\tilde{O}(\sqrt{pT})$ compared to the regret of $\mathcal{A}$. We present a detailed proof of this theorem in Section 5.5.3.3. The main idea is to bound the total effect of noise on the regret via bounding the effect of noise on the output of individual iterations.

Using Chebyshev's inequality, we can obtain the following high probability bound on the regret.

Corollary 5.7. *Let $L > 0$ be the maximum Lipschitz constant of any function f_t in the sequence F, $\mathcal{R}_{\mathcal{A}}(T)$, the regret of the non-private OCP algorithm $\mathcal{A}$ over T-time steps and $\lambda_{\mathcal{A}}$, the sensitivity parameter of $\mathcal{A}$ (see* (5.1)*). Then with probability at least $1-\gamma$, the regret of our Private OCP algorithm (Algorithm 5.1) satisfies:*

$$\sum_{t=1}^{T} f_t(\hat{\theta}_t) - \min_{\theta\in\mathcal{C}} \sum_{t=1}^{T} f_t(\theta) \leq 2\sqrt{p}L(\lambda_{\mathcal{A}} + \|\mathcal{C}\|_2)\sqrt{T}\frac{\log^2\frac{T}{\delta}}{\epsilon\sqrt{\gamma}} + \mathcal{R}_{\mathcal{A}}(T),$$

where d is the dimensionality of the output space, $\|\mathcal{C}\|_2$ is the diameter of $\mathcal{C}$.

5.5.3 Proofs of Lemma 5.4, Theorems 5.5 and 5.6

5.5.3.1 Proof of Lemma 5.4

Proof. First, note that $\hat{\theta}_{t+1}$ is obtained by projecting $\tilde{\theta}_{t+1}$ on $\mathcal{C}$. Thus if $\tilde{\theta}_{t+1}$ is (ϵ', δ')-differentially private, then $\hat{\theta}_{t+1}$ is also (ϵ', δ')-differentially private . Therefore, we prove the lemma for $\tilde{\theta}_{t+1}$.

Let F and F' be two input function sequences that differ in exactly one entry. Suppose θ_{t+1} and θ'_{t+1} are the uncorrupted outputs of the OCP algorithm $\mathcal{A}$ (before adding noise) on input sequences F and F', respectively. Similarly, let $\tilde{\theta}_{t+1} = \theta_{t+1} + b_{t+1}$ and $\tilde{\theta}'_{t+1} = \theta'_{t+1} + b_{t+1}$ be the perturbed t-th step outputs of the algorithm $\mathcal{A}$ on sequences F and F' (see Algorithm 5.1, Step 7). Now by the definition of differential privacy (see Definition 5.2), $\tilde{\theta}_{t+1}$ is $(\epsilon_1, \frac{\delta}{T})$-differential private, if for any measurable set $\Omega \subseteq \mathbb{R}^d$:

$$\Pr[\tilde{\theta}_{t+1} \in \Omega] \leq e^{\epsilon_1} \Pr[\tilde{\theta}'_{t+1} \in \Omega] + \delta/T.$$

Recall that $b_{t+1} \sim \mathcal{N}(0, \frac{\beta^2}{t^2}\mathbb{I}^p)$. We have $(\tilde{\theta}_{t+1} - \theta_{t+1})^T \Delta\theta_{t+1} = {b_{t+1}}^T \Delta\theta_{t+1}$
$\sim \mathcal{N}(0, \frac{\beta^2}{t^2}\|\Delta\theta_{t+1}\|_2^2)$, where $\Delta\theta_{t+1} = \theta_{t+1} - \theta'_{t+1}$.

Also, using the low sensitivity property (5.1) of the OCP algorithm $\mathcal{A}$, $\|\Delta\theta_{t+1}\| \leq \frac{\lambda_{\mathcal{A}}}{t}$. Thus,

$$\begin{aligned}
\Pr\left[\left|(\tilde{\theta}_{t+1} - \theta_{t+1})^T \Delta\theta_{t+1}\right| \geq \frac{\beta\lambda_{\mathcal{A}}}{t^2} z\right] &= \Pr\left[\left|{b_{t+1}}^T \Delta\theta_{t+1}\right| \geq \frac{\beta\lambda_{\mathcal{A}}}{t^2} z\right] \\
&\leq \Pr\left[\left|{b_{t+1}}^T \Delta\theta_{t+1}\right| \geq \frac{\beta}{t}\|\theta_{t+1} - \theta'_{t+1}\|_2 z\right], \\
&\leq e^{-\frac{z^2}{2}},
\end{aligned}$$

where $z > 0$, and the second inequality follows from Mill's inequality. Setting R.H.S. $\leq \frac{\delta}{T}$, we have $z \geq \sqrt{2\log\frac{T}{\delta}}$.

Let $\mathcal{G} \subseteq \mathbb{R}^p$ be a **good** set defined by:

$$b \in \mathcal{G} \text{ iff } \left|b^T \Delta\theta_{t+1}\right| \leq \frac{\beta\lambda_{\mathcal{A}}}{t^2}\sqrt{2\log\frac{T}{\delta}}. \tag{5.3}$$

Note that,

$$\Pr[b_{t+1} \notin \mathcal{G}] = \Pr\left[\left|{b_{t+1}}^T \Delta\theta_{t+1}\right| \geq \frac{\beta\lambda_{\mathcal{A}}}{t^2}\sqrt{2\log\frac{T}{\delta}}\right] \leq \frac{\delta}{T}. \tag{5.4}$$

We now bound $\Pr[\tilde{\theta}_{t+1} \in \Omega]$:

$$\Pr[\tilde{\theta}_{t+1} \in \Omega] \leq \Pr[\tilde{\theta}_{t+1} \in \Omega \wedge b_{t+1} \in \mathcal{G}] + \Pr[b_{t+1} \notin \mathcal{G}] \leq \Pr[\tilde{\theta}_{t+1} \in \Omega \wedge b_{t+1} \in \mathcal{G}] + \frac{\delta}{T}. \tag{5.5}$$

For the purpose of brevity, we define the following notation (which we will be using in the later parts of the proof): for a given set $S \subseteq \mathbb{R}^p$ and a vector $\theta \in \mathbb{R}^p$, the set $\{y : y + \theta \in S\}$ is denoted as $S - \theta$.

Let us define $\Psi = \{\theta : \left|(\theta - \theta_{t+1})^T \Delta\theta_{t+1}\right| \leq \frac{\beta\lambda_{\mathcal{A}}}{t^2}\sqrt{2\log\frac{T}{\delta}}\}$. As $b_{t+1} \sim \mathcal{N}(0, \frac{\beta^2}{t^2}\mathbb{I}^p)$,

$$\begin{aligned}\Pr[\tilde{\theta}_{t+1} \in \Omega \wedge b_{t+1} \in \mathcal{G}] &= \int\limits_{b \in \Omega - \theta_{t+1} \cap \Psi - \theta_{t+1}} \exp\left(-\frac{\|b\|_2^2}{2\beta^2/t^2}\right) db \\ &= \int\limits_{\theta \in \Omega \cap \Psi} \exp\left(-\frac{\|\theta - \theta_{t+1}\|_2^2}{2\beta^2/t^2}\right) d\theta \end{aligned} \tag{5.6}$$

Now, for $\theta \in \Omega \cap \Psi$:

$$\begin{aligned}\frac{\exp\left(-\frac{t^2\|\theta-\theta_{t+1}\|_2^2}{2\beta^2}\right)}{\exp\left(-\frac{t^2\|\theta-\theta'_{t+1}\|_2^2}{2\beta^2}\right)} &= \exp\left(\frac{t^2}{2\beta^2}\Delta\theta_{t+1}^T(2\theta - \theta_{t+1} - \theta'_{t+1})\right), \\ &= \exp\left(\frac{t^2}{2\beta^2}\left(2\Delta\theta_{t+1}^T(\theta - \theta_{t+1}) - \|\Delta\theta_{t+1}\|_2^2\right)\right), \\ &\leq \exp\left(\frac{t^2}{2\beta^2}\left(2|\Delta\theta_{t+1}^T(\theta - \theta_{t+1})| + \|\Delta\theta_{t+1}\|_2^2\right)\right), \\ &\leq \exp\left(\frac{\lambda_{\mathcal{A}}}{\beta}\sqrt{2\log\frac{T}{\delta}} + \frac{\lambda_{\mathcal{A}}^2}{2\beta^2}\right), \\ &\leq e^{\epsilon_1}, \end{aligned} \tag{5.7}$$

where $\epsilon_1 = \frac{\sqrt{\epsilon}}{T^{0.5+c}}$ and β is as given in the lemma statement. The second last inequality follows from the definition of $\mathcal{G}$ and the sensitivity assumption (5.1).

Hence, using (5.5), (5.6), and (5.7), we get:

$$\Pr[\tilde{\theta}_{t+1} \in \Omega] \leq \int\limits_{\theta \in \Omega \cap \Psi} e^{\epsilon_1} \exp\left(-\frac{t^2\|\theta - \theta'_{t+1}\|_2^2}{2\beta^2}\right) d\theta + \frac{\delta}{T} \leq e^{\epsilon_1}\Pr[\tilde{\theta}'_{t+1} \in \Omega] + \frac{\delta}{T}. \tag{5.8}$$

This completes the proof. □

We use the following result from [Dwork et al., 2010b] in our proof of Theorem 5.5.

Lemma 5.8 ([Dwork et al., 2010b])**.** *Suppose that random variables Y and Z satisfy $\max_x \frac{\Pr(Y=x)}{\Pr(Z=x)} \leq \epsilon$ and $\max_x \frac{\Pr(Z=x)}{\Pr(Y=x)} \leq \epsilon$. Then,*

$$\mathcal{D}(Y||Z) = \mathbb{E}_Z\left[\log\frac{\Pr(Z=x)}{\Pr(Y=x)}\right] \leq \epsilon^2,$$

where $\mathcal{D}(Y||Z)$ is the KL-divergence between probability distribution of Y and Z.

5.5.3.2 Proof of Theorem 5.5

Proof. Following the notation in the proof of Lemma 5.4, let $\mathcal{G}_{t+1}$ be the t-th step "good set" defined as:

$$b \in \mathcal{G}_{t+1} \text{ iff } \left|b^T \Delta\theta_{t+1}\right| \leq \frac{\beta\lambda_{\mathcal{A}}}{t^2}\sqrt{2\log\frac{T}{\delta}}, \tag{5.9}$$

where $\Delta\theta_{t+1} = \theta_{t+1} - \theta'_{t+1}$ for $1 \leq t \leq T$.
Now, using (5.4), for each time step t,

$$\Pr[b_{t+1} \notin \mathcal{G}_{t+1}] \leq \frac{\delta}{T}. \tag{5.10}$$

By union bound, the probability that every output vector $b_{t+1} \in \mathcal{G}_{t+1}$ for $1 \leq t \leq T$, is at least $1 - T \cdot \frac{\delta}{T} = 1 - \delta$. That is,

$$\Pr[\exists t \text{ s.t. } b_{t+1} \notin \mathcal{G}_{t+1}] \leq \delta. \tag{5.11}$$

For a random variable θ and any point $\mathrm{a} \in \mathbb{R}^p$, let $\mathrm{pdf}[\theta = \mathrm{a}]$ denote the probability density function of the random variable x evaluated at the point a.
Now, define the following sequence of functions with ξ being some event in the event space,

$$\mathrm{Z}_{t+1}(\mathrm{a}_{t+1}; \xi) = \log\left(\frac{\mathrm{pdf}[\tilde{\theta}_{t+1} = \mathrm{a}_{t+1} \mid \xi]}{\mathrm{pdf}[\tilde{\theta}'_{t+1} = \mathrm{a}_{t+1} \mid \xi]}\right),$$

where $\mathrm{a}_{t+1} \in \mathbb{R}^p$. Recall that $\tilde{\theta}_{t+1} = \theta_{t+1} + b_{t+1}$ and $\tilde{\theta}'_{t+1} = \theta'_{t+1} + b_{t+1}$. Hence, the pdfs in the above equation are associated with the random choice of the noise vectors b_{t+1} which is drawn from a multivariate Gaussian.

Using Lemma 5.4, we have that at each time step t, the output $\tilde{\theta}_{t+1}$ of Algorithm 5.1 is $(\frac{\sqrt{\epsilon}}{T^{0.5+c}}, \frac{\delta}{T})$-differentially private. That is, for $1 \leq t \leq T$,

$$-\frac{\sqrt{\epsilon}}{T^{0.5+c}} \leq \mathrm{Z}_{t+1}(\mathrm{a}_{t+1}; b_{t+1} \in \mathcal{G}_{t+1}) = \log\left(\frac{\mathrm{pdf}[\tilde{\theta}_{t+1} = \mathrm{a}_{t+1} \mid b_{t+1} \in \mathcal{G}_{t+1}]}{\mathrm{pdf}[\tilde{\theta}'_{t+1} = \mathrm{a}_{t+1} \mid b_{t+1} \in \mathcal{G}_{t+1}]}\right) \leq \frac{\sqrt{\epsilon}}{T^{0.5+c}}.$$

Using Lemma 5.8 along with the observation above, we obtain:

$$\mathbb{E}_{b_{t+1}}\left[\mathrm{Z}_{t+1}(\tilde{\theta}_{t+1}; b_{t+1} \in \mathcal{G}_{t+1})\right] \leq \frac{2\epsilon}{T^{1+2c}}.$$

Let $L(\tilde{\theta}_2, \cdots, \tilde{\theta}_{T+1}; b_2 \in \mathcal{G}_2 \cdots, b_{T+1} \in \mathcal{G}_{T+1}) = \sum_{t=1}^{T} \mathrm{Z}_{t+1}(\tilde{\theta}_{t+1}; b_2 \in \mathcal{G}_2 \cdots, b_{T+1} \in \mathcal{G}_{T+1})$. Therefore, we have

$$\begin{aligned}
&\mathbb{E}[L(\tilde{\theta}_2, \cdots, \tilde{\theta}_{T+1}; b_2 \in \mathcal{G}_2 \cdots, b_{T+1} \in \mathcal{G}_{T+1})] \\
&= \sum_{t=1}^{T} \mathbb{E}\left[\mathrm{Z}_{t+1}(\tilde{\theta}_{t+1}; b_{t+1} \in \mathcal{G}_{t+1})\right] \\
&\leq \frac{2T\epsilon}{T^{1+2c}} \leq \frac{2\epsilon}{T^{2c}} \leq 2\epsilon.
\end{aligned}$$

Note also that for every $1 \leq t \leq T$ and $\mathrm{a}_{t+1} \in \mathbb{R}^p$, $|\mathrm{Z}_{t+1}(\mathrm{a}_{t+1}; b_{t+1} \in \mathcal{G}_{t+1})| \leq \frac{\sqrt{\epsilon}}{T^{0.5+c}}$ (from Lemma 5.4). Thus, using Azuma-Hoeffding inequality,

$$\begin{aligned}\Pr[L(\tilde{\theta}_2, \cdots, \tilde{\theta}_{T+1}; b_2 \in \mathcal{G}_2 \cdots, b_{T+1} \in \mathcal{G}_{T+1}) \geq 2\epsilon + \epsilon] &\leq 2\exp\left(\frac{-2\epsilon^2}{T \times \frac{\epsilon}{T^{1+2c}}}\right) \\ &\leq 2\exp\left(-2\epsilon T^{2c}\right). \qquad (5.12)\end{aligned}$$

Now, setting RHS $\leq \delta$, we get: $\delta \geq 2\exp\left(-2\epsilon T^{2c}\right)$. Hence, we select $c = \frac{(\log(\frac{1}{2\epsilon}\log\frac{2}{\delta})}{2\log T}$.

Using (5.12) along with the selected value of c, we have, with probability at least $1-\delta$ over the draws of a_{t+1} from $\tilde{\theta}_{t+1}$,

$$\sum_{t=1}^{T} \log\left(\frac{\mathrm{pdf}[\tilde{\theta}_{t+1} = \mathrm{a}_{t+1} \mid b_{t+1} \in \mathcal{G}_{t+1}]}{\mathrm{pdf}[\tilde{\theta}'_{t+1} = \mathrm{a}_{t+1} \mid b_{t+1} \in \mathcal{G}_{t+1}]}\right) \leq 3\epsilon.$$

That is, with probability at least $1-\delta$, over the draw of $\forall \mathrm{a}_2, \ldots, \mathrm{a}_{T+1} \in \mathbb{R}^p$,

$$\Pi_{t=1}^{T}\mathrm{pdf}(\tilde{\theta}_{t+1} = \mathrm{a}_{t+1} \mid b_{t+1} \in \mathcal{G}_{t+1}) \leq e^{3\epsilon}\ \Pi_{t=1}^{T}\mathrm{pdf}(\tilde{\theta}'_{t+1} = \mathrm{a}_{t+1} \mid b_{t+1} \in \mathcal{G}_{t+1}).$$

Hence, given that $b_{t+1} \in \mathcal{G}_{t+1}$ for $1 \leq t \leq T$, with at least $1-\delta$ probability each $\tilde{\theta}_{t+1}$ $(1 \leq t \leq T)$ is 3ϵ-differentially private.

Now, using (5.11), $\Pr[\exists t \text{ s.t. } b_{t+1} \notin \mathcal{G}_{t+1}] \leq \delta$. Hence, with probability at least $1-2\delta$ over the choice of $b_2, \cdots, b_{T+1}$, each $\tilde{\theta}_{t+1}$ is 3ϵ-differentially private. Therefore, $(3\epsilon, 2\delta)$-differential privacy now follows using a standard argument similar to (5.5). □

5.5.3.3 Proof of Theorem 5.6

Proof. Let $\hat{\theta}_1, \cdots, \hat{\theta}_T$ be the output of the POCP algorithm. By the Lipschitz continuity of the cost functions f_t we have,

$$\begin{aligned}\sum_{t=1}^{T} f_t(\hat{\theta}_t) - \min_{x\in\mathcal{C}} \sum_{t=1}^{T} f_t(\theta) &\leq \sum_{t=1}^{T} f_t(\theta_t) - \min_{\theta\in\mathcal{C}} \sum_{t=1}^{T} f_t(\theta) + L\sum_{t=1}^{T} \|\hat{\theta}_t - \theta_t\|_2, \\ &\leq R_{\mathcal{A}}(T) + L\sum_{t=1}^{T} \|\hat{\theta}_t - \theta_t\|_2. \qquad (5.13)\end{aligned}$$

Since at any time $t \geq 1$, $\hat{\theta}_t$ is the projection of $\tilde{\theta}_t$ on the convex set $\mathcal{C}$, we have

$$\|\theta_{t+1} - \hat{\theta}_{t+1}\|_2 \leq \|\theta_{t+1} - \tilde{\theta}_{t+1}\|_2 = \|b_{t+1}\|_2, \quad \forall 1 \leq t \leq T-1,$$

where b_{t+1} is the noise vector added in the t-th iteration of the POCP algorithm. Therefore,

$$L\sum_{t=1}^{T} \|\theta_t - \hat{\theta}_t\|_2 \leq L\left(\|\mathcal{C}\|_2 + \sum_{t=1}^{T-1} \|b_{t+1}\|_2\right). \qquad (5.14)$$

Now, $b_{t+1} \sim \mathcal{N}(0^p, \frac{\beta^2}{t^2}\mathbb{I}^p)$ where

$$\beta = \lambda_{\mathcal{A}} T^{0.5+c} \sqrt{\frac{2}{\epsilon}\left(\log\frac{T}{\delta} + \frac{\sqrt{\epsilon}}{T^{0.5+c}}\right)}.$$

Therefore, $\|b_{t+1}\|_2$ follows Chi-distribution with parameters $\mu = \frac{\sqrt{2}\frac{\beta}{t}\Gamma((p+1)/2)}{\Gamma(p/2)}$ and $\sigma^2 = \frac{\beta^2}{t^2}(p - \mu^2)$.

Using $c = \frac{\log(\frac{1}{2\epsilon}\log\frac{2}{\delta})}{2\log T}$,

$$\begin{aligned}
\mathbb{E}[\sum_{t=1}^{T-1}\|b_{t+1}\|_2] &\leq \frac{\sqrt{2}\beta\Gamma((p+1)/2)}{\Gamma(p/2)}\int_1^{T-1}\frac{1}{t}dt, \\
&\leq \frac{\Gamma((p+1)/2)}{\Gamma(p/2)}\lambda_{\mathcal{A}}\sqrt{T}\log T\sqrt{\frac{2}{\epsilon^2}\log\frac{2}{\delta}\left(\log\frac{T}{\delta} + \frac{\epsilon}{\sqrt{\frac{T}{2}\log\frac{2}{\delta}}}\right)}, \\
&\leq 2\sqrt{p}\lambda_{\mathcal{A}}\sqrt{T}\frac{\log^2\frac{T}{\delta}}{\epsilon}. \qquad (5.15)
\end{aligned}$$

The theorem now follows by combining (5.13), (5.14), (5.15). □

5.6 Implicit Gradient Descent Algorithm

In this section and in Section 5.7, we provide transformation of two standard online learning algorithms into corresponding privacy preserving algorithms with provable regret. In both these examples, we show low-sensitivity of the corresponding learning algorithms and use our analysis of POCP to obtain privacy and utility bounds. We can obtain similar low-sensitivity bounds for several other OCP algorithms such as Follow The Leader (FTL) and Follow the Regularized Leader (FTRL) [Hazan et al., 2007], and hence use those methods with our POCP framework as well. Our low-sensitivity proofs should be of independent interest as well, as they point to a connection between stability (sensitivity) and low-regret (online learnability)—an active topic of research in the learning community [Ross and Bagnell, 2011; Poggio et al., 2011].

In this section we consider the Implicit Gradient Descent (IGD) algorithm by [Kulis and Bartlett, 2010] and present a differentially private version using our generic framework (see Algorithm 5.1). At each step t, IGD selects the output θ_{t+1} using:

$$\text{IGD}: \qquad \theta_{t+1} \leftarrow \operatorname*{argmin}_{\theta\in\mathcal{C}} \frac{1}{2}\|\theta - \theta_t\|_2^2 + \eta_t f_t(\theta), \qquad (5.16)$$

where $\eta_t = \frac{1}{\Psi t}$, $\Psi > 0$ is the minimum strong convexity parameter of any f_t, $t \leq T$. Now if each $f_t(x)$ is a Lipschitz continuous strongly convex function, then a simple modification to the proof by [Kulis and Bartlett, 2010] shows $O(\log T)$ regret for IGD, i.e. $\mathcal{R}_{\text{IGD}}(T) = O(\log T)$.

Now, we instantiate our generic POCP framework using the IGD algorithm. See

Algorithm 5.2 Private Implicit Gradient Descent (PIGD)

Input: Cost function sequence $F = \langle f_1, \cdots, f_T \rangle$ and the convex set $\mathcal{C}$
Parameter: privacy parameters (ϵ, δ), maximum Lipschitz constant L and minimum strong convexity parameter Ψ of any function in F
Choose θ_1 and $\hat{\theta}_1$ randomly from $\mathcal{C}$
for $t = 1$ to $T - 1$ **do**
  Cost: $L_t(\hat{\theta}_t) = f_t(\hat{\theta}_t)$
  Learning rate: $\eta_t = \frac{1}{\Psi t}$
  IGD Update: $\theta_{t+1} \leftarrow \operatorname{argmin}_{\theta \in \mathcal{C}} \left(\frac{1}{2} \|\theta - \theta_t\|_2^2 + \eta_t f_t(\theta) \right)$
  Noise Addition: $\tilde{\theta}_{t+1} \leftarrow \theta_{t+1} + b_{t+1}$, $b_{t+1} \sim \mathcal{N}(0^p, \frac{\beta^2}{t^2} \mathbb{I}^p)$, where $\beta = \frac{2LT^{0.5+c}}{\Psi} \sqrt{\frac{2}{\epsilon} \left(\log \frac{T}{\delta} + \frac{\sqrt{\epsilon}}{T^{0.5+c}} \right)}$ and $c = \frac{\log \frac{1}{2\epsilon} \log(2/\delta)}{2 \log T}$
  Output $\hat{\theta}_{t+1} = \operatorname{argmin}_{\theta \in \mathcal{C}} \left(\|\theta - \tilde{\theta}_{t+1}\|_2^2 \right)$
end for

Algorithm 5.2 for a pseudo-code of our Private IGD (PIGD) algorithm. Similar to POCP, our PIGD algorithm also adds an appropriately calibrated noise at each step to obtain differentially private outputs $\hat{\theta}_{t+1}$.

Now, to use generic privacy analysis of our POCP framework, we need to show that IGD satisfies sensitivity bound of (5.1). To this end, in the following lemma we bound sensitivity of IGD at each step. At a high level, our proof uses optimality of each output θ_{t+1} along with strong convexity of each f_t.

Lemma 5.9 (IGD Sensitivity)**.** *L_2-sensitivity (see Definition 5.3) of the IGD algorithm is $\frac{2L}{\Psi t}$ for the t-th iterate, where L is the maximum Lipschitz constant of any function $f_\tau, 1 \leq \tau \leq t$.*

Proof. We prove the lemma using mathematical induction.
Base Case ($t = 1$): As θ_1 is chosen randomly, it's value doesn't depend on the underlying data set.
Induction Step ($t = \tau + 1$): Consider the following function that is optimized at the $(\tau + 1)$-step of IGD:

$$\tilde{f}_\tau(\theta) = \frac{1}{2} \|\theta - \theta_\tau\|_2^2 + \eta_\tau f_\tau(\theta).$$

As f_τ is Ψ strongly convex, the strong convexity coefficient of the above given function is $\frac{\tau+1}{\tau}$.

Now using strong convexity of $\tilde{f}_\tau$ and the fact that at optima $\theta_{\tau+1}$, $\langle \bigtriangledown \tilde{f}_\tau(\theta_{\tau+1}), \theta - \theta_{\tau+1} \rangle \geq 0, \forall \theta \in \mathcal{C}$, we get:

$$\tilde{f}_\tau(\theta'_{\tau+1}) \geq \tilde{f}_\tau(\theta_{\tau+1}) + \frac{\tau + 1}{2\tau} \|\theta_{\tau+1} - \theta'_{\tau+1}\|_2^2. \tag{5.17}$$

Now, we consider two cases:

- $F - F' = \{f_\tau\}$: Define $\tilde{f}'_\tau(\theta) = \frac{1}{2} \|\theta - \theta_\tau\|^2 + \eta_\tau f'_\tau(\theta)$ and let $\theta'_{\tau+1} = \operatorname{argmin}_{\theta \in \mathcal{C}} \tilde{f}'_\tau(\theta)$.

Then, similar to (5.17), we get:

$$\tilde{f}'_\tau(\theta_{\tau+1}) \geq \tilde{f}'_\tau(\theta'_{\tau+1}) + \frac{\tau+1}{2\tau}\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2^2. \tag{5.18}$$

Adding (5.17) and (5.18), we get:

$$\begin{aligned}\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2^2 &\leq \frac{1}{\Psi(\tau+1)}|f_\tau(\theta'_{\tau+1}) + f'_\tau(\theta_{\tau+1}) - f_\tau(\theta_{\tau+1}) - f'_\tau(\theta'_{\tau+1})| \\ &\leq \frac{2L}{\Psi(\tau+1)}\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2.\end{aligned}$$

Lemma now follows using simplification.

- $F - F' = \{f_i\},\ i < \tau$: Define $\tilde{f}'_\tau(\theta) = \frac{1}{2}\|\theta - \theta'_\tau\|^2 + \eta_\tau f_\tau(\theta)$ and let $\theta'_{\tau+1} = \operatorname{argmin}_{\theta\in\mathcal{C}} \tilde{f}'_\tau(\theta)$. Then, similar to (5.17), we get:

$$\tilde{f}'_\tau(\theta_{\tau+1}) \geq \tilde{f}'_\tau(\theta'_{\tau+1}) + \frac{\tau+1}{2\tau}\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2^2. \tag{5.19}$$

Adding (5.17) and (5.19), we get:

$$\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2^2 \leq \frac{\tau}{\tau+1}|(\theta_{\tau+1} - \theta'_{\tau+1}) \cdot (\theta_\tau - \theta'_\tau)| \leq \frac{\tau}{\tau+1}\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2\|\theta_\tau - \theta'_\tau\|_2.$$

The lemma now follows using the induction hypothesis.

□

Using the above lemma and Theorem 5.5, privacy guarantee for PIGD follows directly.

Theorem 5.10 (PIGD Privacy)**.** PIGD *(see Algorithm 5.2) is* $(3\epsilon, 2\delta)$*-differentially private.*

Next, the utility (regret) analysis of our PIGD algorithm follows directly using Theorem 5.6 along with the regret bound of the IGD algorithm, $\mathcal{R}_{\text{IGD}}(T) = O((\frac{L^2}{\Psi} + \|\mathcal{C}\|_2)\log T)$.

Theorem 5.11 (PIGD Regret)**.** *Let L be the maximum Lipschitz constant and let Ψ be the minimum strong convexity parameter of any function f_t in the function sequence F. Then the expected regret of the private IGD algorithm over T steps is $\tilde{O}(\sqrt{pT})$. Specifically,*

$$\mathbb{E}[\sum_{t=1}^{T} f_t(\hat{\theta}_t)] - \min_{\theta\in\mathcal{C}} \sum_{t=1}^{T} f_t(\theta) = O\left(\frac{(L^2/\Psi + \|\mathcal{C}\|_2)\sqrt{p}\log^2\frac{T}{\delta}}{\epsilon}\sqrt{T}\right),$$

where p is the dimensionality of the output space.

Algorithm 5.3 Private GIGA (PGIGA)

1: **Input:** Cost function sequence $F = \langle f_1, \cdots, f_T \rangle$ and the convex set $\mathcal{C}$
2: **Parameter:** Privacy parameters (ϵ, δ), Lipschitz continuity (L) and strong convexity (Ψ) bound on the function sequence F, $t_q = 2L_G^2/\Psi^2$
3: Choose $\theta_1, \ldots, \theta_{t_q-1}$ and $\hat{\theta}_1, \ldots, \hat{\theta}_{t_q-1}$ randomly from $\mathcal{C}$, incurring a cost of $\sum_{t=1}^{t_q-1} f_t(\hat{\theta}_t)$
4: **for** $t = t_q$ to $T-1$ **do**
5: **Cost:** $L_t(\hat{\theta}_t) = f_t(\hat{\theta}_t)$
6: **Step Size:** $\eta_t = \frac{2}{\Psi t}$
7: **GIGA Update:** $\theta_{t+1} \leftarrow \text{argmin}_{x \in \mathcal{C}} \left(\|\theta - (\theta_t - \eta_t \bigtriangledown f_t(\theta_t))\|_2^2 \right)$
8: **Noise Addition:** $\tilde{\theta}_{t+1} \leftarrow \theta_{t+1} + b_{t+1}$, $b_{t+1} \sim \mathcal{N}(0^p, \frac{\beta^2}{t^2}\mathbb{I}^p)$, where $\beta = \frac{2GT^{0.5+c}}{\Psi} \sqrt{\frac{2}{\epsilon}\left(\log \frac{T}{\delta} + \frac{\sqrt{\epsilon}}{T^{0.5+c}}\right)}$ where $c = \frac{\log \frac{1}{2\epsilon} \log(2/\delta)}{2 \log T}$
9: Output $\hat{\theta}_{t+1} = \text{argmin}_{x \in \mathcal{C}} \left(\|x - \tilde{\theta}_{t+1}\|_2^2 \right)$
10: **end for**

5.7 Private Generalized Infinitesimal Gradient Ascent Algorithm

In this section, we apply our general differential privacy framework to the Generalized Infinitesimal Gradient Ascent (GIGA) algorithm [Zinkevich, 2003], which is one of the most popular algorithms for OCP. GIGA is a simple extension of the classical projected gradient method to the OCP problem. Specifically, the iterates θ_{t+1} are obtained by a projection onto the convex set $\mathcal{C}$, of the output of the gradient descent step $\theta_t - \eta_t \bigtriangledown f_t(\theta_t)$ where $\eta_t = 1/\Psi t$, and Ψ is the minimum strong convexity parameter of any function f_t in F.

For the rest of this section, we assume that each of the function f_t in the input function sequence F are *differentiable*, Lipschitz continuous gradient and strongly convex. Note that this is a stricter requirement than our private IGD algorithm where we require only the Lipschitz continuity of f_t.

Proceeding similar to IGD, we obtain a privacy preserving version of the GIGA algorithm using our generic POCP framework (See Algorithm 5.1). Algorithm 5.3 details the steps involved in our Private GIGA (PGIGA) algorithm. Note that PGIGA has an additional step (Step 3) compared to POCP (Algorithm 5.1). This step is required to prove the sensitivity bound in Lemma 5.12 given below.

Furthermore, we provide the privacy and regret guarantees for our PGIGA algorithm using Theorem 5.5 and Theorem 5.6. To this end, we first show that GIGA satisfies the sensitivity assumption mentioned in (5.1).

Lemma 5.12 (GIGA Sensitivity). *Let $\Psi > 0$ be the minimum strong convexity parameter of any function f_t in the function sequence F. Also, let L_G be the maximum Lipschitz continuity parameter of the gradient of any function $f_t \in F$ and let $G = \max_\tau \| \bigtriangledown f_t(x)\|_2, \forall x \in \mathcal{C}$. Then, L_2-sensitivity (see Definition 5.3) of the GIGA algorithm is $\frac{2G}{\Psi t}$ for the t-th iterate, where $1 \leq t \leq T$.*

Proof. Let θ_{t+1} and $\tilde{\theta}'_{t+1}$ be the t-th iterates when GIGA is applied to F and F', respectively. Using this notation, to prove the L_2 sensitivity of GIGA, we need to show that:

$$\|\theta_{t+1} - \theta'_{t+1}\| \leq \frac{2G}{\Psi t}$$

We prove the above inequality using mathematical induction.
Base Case ($1 \leq t \leq t_q = 2L_G^2/\Psi^2 + 1$): As $\theta_1, \ldots, \theta_{t_q}$ are selected randomly, their value doesn't depend on the underlying data set. Hence, $\theta_t = \theta'_t, \forall 1 \leq t \leq t_q$.
Induction Step $t = \tau > 2L_G^2/\Psi^2 + 1$: We consider two cases:

- $F - F' = \{f_\tau\}$: Since the difference between F and F' is only the τ-th function, hence $\theta_\tau = \theta'_\tau$. As $\mathcal{C}$ is a convex set, projection onto $\mathcal{C}$ always decreases distance, hence:

$$\begin{aligned}\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2 &\leq \|(\theta_\tau - \eta_\tau \bigtriangledown f_\tau(\theta_\tau)) - (\theta_\tau - \eta_\tau \bigtriangledown f'_\tau(\theta_\tau))\|_2, \\ &= \eta_\tau \| \bigtriangledown f_\tau(\theta_\tau) - \bigtriangledown f'_\tau(\theta_\tau)\|_2, \\ &\leq \frac{2G}{\Psi\tau}.\end{aligned}$$

  Hence, lemma holds in this case.

- $F - F' = \{f_i\},\ i < \tau$: Again using convexity of $\mathcal{C}$, we get:

$$\begin{aligned}\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2^2 &\leq \|(\theta_\tau - \eta_\tau \bigtriangledown f_\tau(\theta_\tau)) - (\theta'_\tau - \eta_\tau \bigtriangledown f_\tau(\theta'_\tau))\|_2^2, \\ &= \|\theta_\tau - \theta'_\tau\|_2^2 + \eta_\tau^2 \| \bigtriangledown f_\tau(\theta_\tau) - \bigtriangledown f_\tau(\theta'_\tau)\|_2^2 \\ &\qquad - 2\eta_\tau(\theta_\tau - \theta'_\tau)^T(\bigtriangledown f_\tau(\theta_\tau) - \bigtriangledown f_\tau(\theta'_\tau)), \\ &\leq (1 + \eta_\tau^2 L_G^2)\|\theta_\tau - \theta'_\tau\|_2^2 - 2\eta_\tau(\theta_\tau - \theta'_\tau)^T(\bigtriangledown f_\tau(\theta_\tau) - \bigtriangledown f_\tau(\theta'_\tau)),\end{aligned} \tag{5.20}$$

  where the last equation follows using Lipschitz continuity of $\bigtriangledown f_t$. Now, using strong convexity:

$$(\theta_\tau - \theta'_\tau)^T(\bigtriangledown f_\tau(\theta_\tau) - \bigtriangledown f_\tau(\theta'_\tau)) \geq \Psi\|\theta_\tau - \theta'_\tau\|_2^2.$$

  Combining the above observation and the induction hypothesis with (5.20):

$$\|\theta_{\tau+1} - \theta'_{\tau+1}\|_2^2 \leq \left(1 + L_G^2\eta_\tau^2 - 2\Psi\eta_\tau\right) \cdot \frac{4G^2}{(\tau-1)^2}. \tag{5.21}$$

  Lemma now follows by setting $\eta_\tau = \frac{2}{\Psi\tau}$ and $\tau > \frac{2L_G^2}{\Psi^2}$.

□

Using the lemma above with the privacy analysis of POCP (Theorem 5.5), the privacy guarantee for PGIGAfollows immediately.

Theorem 5.13 (PGIGA Privacy)**.** PGIGA *(see Algorithm 5.3) is* $(3\epsilon, 2\delta)$*-differentially private.*

Next, using the regret bound analysis for GIGA from [Hazan et al., 2007] (Theorem 1) along with Theorem 5.6, we get the following utility (regret bound) analysis for our PGIGA algorithm. Here again, ignoring constants, the regret simplifies to $\tilde{O}(\sqrt{pT})$.

Theorem 5.14 (PGIGA Regret)**.** *Let $\Psi > 0$ be the minimum strong convexity parameter of any function f_t in the function sequence F. Also, let L_G be the maximum Lipschitz continuity parameter of the gradient of any function $f_t \in F$ and let $G = \max_\tau \| \bigtriangledown f_t(x)\|_2, \forall x \in \mathcal{C}$. Then, the expected regret of* PGIGA *satisfies*

$$\mathbb{E}[\mathcal{R}_{\mathsf{PGIGA}}(T)] \leq \frac{4\sqrt{p}(G/\Psi + \|\mathcal{C}\|_2)G\log^2\frac{T}{\delta}}{\epsilon}\sqrt{T} + \frac{2G^2}{\Psi}(1+\log T) + \frac{2L_G^2 G\|\mathcal{C}\|_2}{\Psi^2}$$

where $\|\mathcal{C}\|_2$ is the diameter of the convex set $\mathcal{C}$ and d is the dimensionality of the output space.

Proof. Observe that for the first $t_q = \frac{2L_G^2}{\Psi^2}$ iterations PGIGA outputs random samples from $\mathcal{C}$. The additional regret incurred during this time is bounded by a constant (w.r.t. T) that appears as the last term in the regret bound given above. For iterations $t \geq t_q$, the proof follows directly by using Theorem 5.6 and regret bound of GIGA. Note that we use a slightly modified step-size $\eta_t = 2/\Psi t$, instead of the standard $\eta_t = 1/\Psi t$. This difference in the step size increases the regret of GIGA as given by [Hazan et al., 2007] by a factor of 2. □

5.8 Application to Offline Learning

In Section 5.5, we proposed a generic framework for differentially private OCP algorithms with sub-linear regret bounds. Recently, [Kakade and Tewari, 2008] showed that OCP algorithms with sub-linear regret bounds can be used to solve several offline learning problems as well. In this section, we exploit this connection to provide a generic differentially private framework for a large class of offline learning problems as well.

In related works, [Chaudhuri et al., 2011; Rubinstein et al., 2009] also proposed methods to obtain differentially private algorithms for offline learning problems. However, as discussed later in the section, our method is more practical and obtains better error bounds for the same level of privacy. It also covers a wider range of problems than [Chaudhuri et al., 2011].

First, we describe the standard offline learning model that we use. Consider a domain $\mathcal{T}$ and an arbitrary distribution $\mathcal{P}$ over the domain $\mathcal{T}$ from which the training data is generated. Let $\mathcal{D} = \langle d_1, \cdots, d_T \rangle$ be the training data set, where each d_i is drawn i.i.d. from the distribution $\mathcal{P}$. Typically, s_i is a tuple of a training point and its label. Also, consider a loss function $\ell : \mathcal{C} \times \mathcal{T} \to \mathbb{R}$, where $\mathcal{C} \subseteq \mathbb{R}^p$ is a (potentially unbounded) convex set. Let $\ell(\cdot;\cdot)$ be a L-Lipschitz (in the first parameter) convex function. Intuitively, the loss function quantifies the goodness of a learned model $\theta \in \mathcal{C}$ w.r.t. the training data. Now, the goal is to solve the following *risk minimization* problem:

$$\min_{\theta \in \mathcal{C}} \mathbb{E}_{d \sim \mathcal{P}}[\ell(\theta; d)]. \tag{5.22}$$

Let θ^* be the optimal solution to (5.22), i.e., $\theta^* = \arg\min_{\theta\in\mathcal{C}} \mathbb{E}_{d\sim\mathcal{P}}[\ell(\theta; d)]$. Recently, [Kakade and Tewari, 2008] provided a stochastic offline learning algorithm to obtain an additive approximation to (5.22) via OCP. The algorithm of [Kakade and Tewari, 2008] is as follows: execute any reasonable OCP algorithm $\mathcal{A}$ (like IGD or GIGA) on the function sequence F, where $f_t = \ell(\theta; d_t) + \frac{\Psi}{2}\|\theta\|^2$. Note that each d_t is sampled i.i.d. from $\mathcal{P}$. Also, if the convex set $\mathcal{C}$ required in OCP is an unbounded set, then it can be set to be an L_2 ball of radius $\|\theta^*\|_2$, i.e, $\mathcal{C} = \{\theta : \theta \in \mathbb{R}^p, \|\theta\|_2 \leq \|\theta^*\|_2\}$. In practice, $\|\theta^*\|$ can be estimated using cross validation, which is analogous to tuning the regularization parameter in standard learning problems like support vector machine (SVM).

Let $\theta_1, \cdots, \theta_T$ be the sequence of outputs produced by $\mathcal{A}$. Then, the output of the stochastic offline learning algorithm is given by, $\tilde{\theta} = \frac{1}{T}\sum_{t=1}^{T}\theta_t$. [Kakade and Tewari, 2008] show that $\tilde{\theta}$ is a reasonable approximation to θ^* with provable approximation error (see Theorem 5.18).

To produce differentially private output, we add noise of an appropriate variance to the output $\tilde{x}$ and project it back to $\mathcal{C}$. That is,

$$\mathsf{POffL} : \hat{\theta} = \operatorname*{argmin}_{\theta\in\mathcal{C}} \|\theta - \tilde{\theta} - b\|_2^2, \quad b \sim \mathcal{N}(0, \beta^2\mathbb{I}^p), \quad \beta = \frac{2\sqrt{2}(L + \Psi\|\theta^*\|_2)\log T}{T\epsilon_p}\sqrt{\log\frac{1}{\delta} + \epsilon_p}.$$

We refer to this algorithm as Private Offline Learning (POffL) and provide a detailed pseudo-code in Algorithm 5.4. Next, we show that POffL (Algorithm 5.4) is differentially private.

Algorithm 5.4 Private Offline Learning (POffL)

1: **Input:** Input data set $D = \langle d_1, \cdots, d_T\rangle$ and the convex set $\mathcal{C}$
2: **Parameter:** Privacy parameters (ϵ_p, δ), generalization error parameter ϵ_g, Lipschitz bound L on the loss function ℓ, bound on $\|\theta^*\|_2$
3: If $\mathcal{C} = \mathbb{R}^p$ then set $\mathcal{C} = \{\theta : \theta \in \mathbb{R}^p, \|\theta\|_2 \leq \|\theta^*\|_2\}$.
4: Choose θ_1 randomly from $\mathcal{C}$
5: Set $\Psi \leftarrow \frac{\epsilon_g}{\|\theta^*\|_2^2}$
6: Initialize $\boldsymbol{s} = \theta_1$
7: **for** $t = 1$ to $T - 1$ **do**
8: **Learning rate:** $\eta_t = \frac{1}{\Psi t}$
9: **IGD Update:** $\theta_{t+1} \leftarrow \operatorname{argmin}_{\theta\in\mathcal{C}}\left(\frac{1}{2}\|\theta - \theta_t\|_2^2 + \eta_t(\ell(\theta; d_t) + \frac{\Psi}{2}\|\theta\|_2^2)\right)$
10: **Store sum:** $\boldsymbol{s} \leftarrow \boldsymbol{s} + \theta_{t+1}$
11: **end for**
12: **Average:** $\tilde{\theta} \leftarrow \frac{\boldsymbol{s}}{T}$
13: **Noise Addition:** $\bar{\theta} \leftarrow \tilde{\theta} + b$, where $b \sim \mathcal{N}(0^p, \beta^2\mathbb{I}^p)$ and $\beta = \frac{2\sqrt{2}(L+\Psi\|\theta^*\|_2)\log T}{T\epsilon_p}\sqrt{\log\frac{1}{\delta} + \epsilon_p}$
14: Output $\hat{\theta} = \operatorname{argmin}_{\theta\in\mathcal{C}}\left(\|\theta - \bar{\theta}\|_2^2\right)$

Theorem 5.15 (POffL Privacy)**.** *Private Offline Learning (POffL) algorithm (see Algorithm 5.4) is (ϵ_p, δ)-differentially private.*

Proof. Recall that to prove differential privacy, one needs to show that changing one

training point from the data set D doesn't lead to significant change in the algorithm's output $\hat{\theta}$ which is a perturbation of $\tilde{\theta} = \frac{1}{T}\sum_{t=1}^{T}\theta_t$. Hence, we need to show that the L_2-sensitivity (see Definition 5.3) of $\tilde{\theta}$ is low.

Now let $\theta'_1, \cdots, \theta'_T$ be the sequence of outputs produced by the IGD algorithm used in Algorithm 5.4 when executed on a data set D' which differs in exactly one entry from D. To estimate the sensitivity of $\tilde{\theta}$, we need to bound $\|\frac{1}{T}\sum_{t=1}^{T}(\theta_t - \theta'_t)\|_2$. Now, using triangle inequality and Lemma 5.9, we get:

$$\|\frac{1}{T}\sum_{t=1}^{T}(\theta_t - \theta'_t)\|_2 \leq \frac{1}{T}\sum_{t=1}^{T}\|\theta_t - \theta'_t\|_2 \leq \frac{1}{T}\sum_{t=2}^{T}\frac{2L'}{t-1} \leq \frac{2L'\log T}{T}, \tag{5.23}$$

where L' is the maximum Lipschitz continuity coefficient of $\ell(\theta, d_t) + \frac{\Psi}{2}\|\theta\|_2^2, \forall t$ over the set $\mathcal{C}$. Using the fact that $\|\mathcal{C}\|_2 = \|\theta^*\|_2$, we obtain $L' = L + \Psi\|\theta^*\|_2$.

The theorem now follows using L_2-sensitivity of $\tilde{x}$ (see (5.23)) and an argument similar to that of the proof for Lemma 5.4. □

Next, we provide a utility guarantee for POffL, i.e., a bound on the approximation error for the Risk Minimization problem (5.22). For the ease of presentation, we defer the proof till Section 5.8.2. The main tool in the proof is a bound on the approximation error for the stochastic offline learning by [Kakade and Tewari, 2008].

Theorem 5.16 (POffL Utility (Approximation Error in Eq. 5.22))**.** *Let L be the Lipschitz bound on the loss function ℓ and T be the total number of points in the training data set $D = \{d_1, \cdots, d_T\}$. Let (ϵ_p, δ) be the differential privacy parameters, p the dimensionality, $C > 0$ a global constant. Then, with probability at least $1 - \gamma$,*

$$\mathbb{E}_{d\sim\mathcal{P}}[\ell(\hat{\theta}; d)] - \min_{\theta\in\mathcal{C}}\mathbb{E}_{d\sim\mathcal{P}}[\ell(\theta; d)] \leq \epsilon_g,$$

when the number of points sampled (T) satisfies,

$$T \geq C\max\left(\frac{\sqrt{p}L(L+\epsilon_g/\|\theta^*\|_2)\sqrt{\log\frac{1}{\gamma}\log\frac{1}{\delta}}}{\epsilon_g\epsilon_p}, \frac{(L+\epsilon_g/\|\theta^*\|_2)^2\|\theta^*\|_2^2\log T\log\frac{\log T}{\gamma}}{\epsilon_g^2}\right).$$

Empirical risk minimization via POffL: One can interpret the above utility guarantee (Theorem 5.16) in terms of empirical risk minimization too. Consider the data set $\mathcal{D} = \{d_1, \cdots, d_T\}$ and consider a data set $\mathcal{D}_{emp}$ of size T generated by sampling points from $\mathcal{D}$ uniformly at random with replacement. Now, with probability at least $1 - e^{-T}$, any entry in $\mathcal{D}$ does not appear in $\mathcal{D}_{emp}$ more than $\log T$ number of times. So using data set $\mathcal{D}_{emp}$ and setting the new ϵ_p as $\epsilon_p/\log T$ and δ as $\delta - e^{-T}$ in Algorithm 5.4, we can get (ϵ_p, δ)-differential privacy guarantee.

Now notice that $\mathcal{P}$ in Theorem 5.16 is empirical distribution defined on the data set $\mathcal{D}$. Therefore, $\mathbb{E}_{d\sim\mathcal{P}}[\ell(\theta; d)] = \frac{1}{T}\sum_{i=1}^{T}\ell(\theta; d_i)$. Therefore, we get the following utility guarantee in terms of empirical risk minimization.

Corollary 5.17 (POffL Utility (empirical risk minimization bound))**.** *Let L be the Lipschitz bound on the loss function ℓ and T be the total number of points in the training*

data set $D = \{d_1, \cdots, d_T\}$. *Let* (ϵ_p, δ) *be the differential privacy parameters,* p *the dimensionality,* $C > 0$ *a global constant. Then, with probability at least* $1 - \gamma$,

$$\frac{1}{T}\sum_{i=1}^{T} \ell(\hat{\theta}; d_i) - \min_{\theta \in \mathcal{C}} \frac{1}{T}\sum_{i=1}^{T} \ell(\theta; d_i) \leq \epsilon_g,$$

when the number of points T *satisfies,*
$T \geq C \max\left(\frac{\sqrt{p} L \log T (L + \epsilon_g/\|\theta^*\|_2)\sqrt{\log\frac{1}{\gamma}\log\frac{1}{\delta}}}{\epsilon_g \epsilon_p}, \frac{(L+\epsilon_g/\|\theta^*\|_2)^2 \|\theta^*\|_2^2 \log T \log\frac{\log T}{\gamma}}{\epsilon_g^2}\right)$. *Here* $\theta^* = \arg\min_{\theta \in \mathcal{C}} \frac{1}{T}\sum_{i=1}^{T} \ell(\theta; d_i)$.

5.8.1 Comparison to Output and Objective perturbation algorithms from Chapter 3

We now compare our POffL algorithm for private (offline) risk minimization with the existing methods (output and objective perturbation) [Chaudhuri et al., 2011; Rubinstein et al., 2009]:

- **Better error bound**: Our Theorem 5.16 improves the worst case sample complexity bounds of [Chaudhuri et al., 2011; Rubinstein et al., 2009] by a factor of $\sqrt{p}$ in terms of dimensionality (but at the cost of $\sqrt{\log(1/\delta)}$). We believe the difference is primarily due to our use of Gaussian noise instead of Gamma noise added by the existing methods. It is important to mention that the current sample complexity bound via OCP is still worse than the version of the objective perturbation algorithm in Chapter 3.

- **More practical**: Both output perturbation and objective perturbation (in Chapter 3) need to compute the *exact* optimal solution to the optimization problem that they consider and it not clear if their privacy guarantees hold if one can obtain only an approximate solution to their respective optimization problems. In contrast, our method uses an explicit iterative method for solving (5.22) and provides privacy and utility guarantees even if the algorithm stops early.

  It is important to note that this comparison is meaningful under the assumption that one can efficiently perform an Euclidean projection onto the convex set $\mathcal{C}$.

Remark: Note that the output perturbation of [Chaudhuri et al., 2011] does not allow the loss function ℓ to be non-differentiable and the convex set $\mathcal{C}$ to be bounded. In comparison both [Rubinstein et al., 2009] and our POffL method supports non-differentiable loss functions and bounded convex sets. Also, note that our $\sqrt{p}$ sample complexity bound does not contradict the corresponding $\Omega(p)$ lower bound proved by [Chaudhuri and Hsu, 2011]; reason being, we use (ϵ, δ)-differential privacy notion which is a less strict notion of privacy than the ϵ-differential privacy notion used by [Chaudhuri and Hsu, 2011].

Note on the definition of differential privacy for online learning: In Definition 5.2, the two neighboring function streams F and F' differ in one entry in the following

way. Any one entry in F is replaced by an arbitrary entry from the domain. Thus, both F and F' have the same length. In contrast, Definition 2.1 defines two neighboring data sets $\mathcal{D}$ and $\mathcal{D}'$ to be such that either $\mathcal{D}$ has one entry more (less) compared to $\mathcal{D}'$. Thus the lengths of $\mathcal{D}$ and $\mathcal{D}'$ are off by one. For the exposition of results in this chapter, it is easier to work with Definition 5.2.

5.8.2 Proof of Theorem 5.16

Before proving the utility guarantee, we first rewrite the approximation error incurred by $\tilde{\theta} = \frac{1}{T}\sum_{t=1}^{T}\theta_t$, as derived by [Kakade and Tewari, 2008].

Theorem 5.18 (Approximation Error in Risk Minimization (Eq. 5.22) [Kakade and Tewari, 2008])**.** *Let $\mathcal{R}_{\mathcal{A}}(T)$ be the regret for the algorithm $\mathcal{A}$ and $B = \max_{d\in\mathcal{T},\theta\in\mathcal{C}} \ell(\theta; d)$. Then with probability at least $1-\gamma$,*

$$\mathbb{E}_{d\sim\mathcal{P}}[\ell(\tilde{\theta}; d)] - \mathbb{E}_{d\sim\mathcal{P}}[\ell(\theta^*; z)] \leq \frac{\Psi}{2}\|\theta^*\|^2 + \frac{\mathcal{R}_{\mathcal{A}}(T)}{T} + \frac{4}{T}\sqrt{\frac{L'^2\mathcal{R}_{\mathcal{A}}(T)\log(\frac{4\log T}{\gamma})}{\Psi}} + \frac{\max\{\frac{16L'^2}{\Psi}, 6B\}\log(\frac{4\log T}{\gamma})}{T}$$

where $L' = L + \Psi\|\theta^*\|_2$, L is the Lipschitz continuity bound on the loss function ℓ and Ψ is the strong convexity parameter of the function sequence F.

With this result in place, we now proceed to the proof for Theorem 5.16.

Proof. To prove the result, we upper bound $\mathbb{E}_{d\sim\mathcal{P}}[\ell(\hat{\theta}; d)] - \mathbb{E}_{d\sim\mathcal{P}}[\ell(\theta^*; d)]$ as:

$$\begin{aligned}
\mathbb{E}_{d\sim\mathcal{P}}[\ell(\hat{\theta}; d)] - \mathbb{E}_{d\sim\mathcal{P}}[\ell(\theta^*; z)] &= \mathbb{E}_{d\sim\mathcal{P}}[\ell(\hat{\theta}; d)] - \mathbb{E}_{d\sim\mathcal{P}}[\ell(\tilde{\theta}; d)] \\
&\quad + \mathbb{E}_{d\sim\mathcal{P}}[\ell(\tilde{\theta}; d)] - \mathbb{E}_{d\sim\mathcal{P}}[\ell(\theta^*; d)], \\
&\leq L\|\hat{\theta} - \tilde{\theta}\|_2 + \mathbb{E}_{d\sim\mathcal{P}}[\ell(\tilde{\theta}; d) - \ell(\theta^*; d)], \\
&= L\|b\|_2 + \mathbb{E}_{d\sim\mathcal{P}}[\ell(\tilde{\theta}; z) - \ell(\theta^*; d)], \qquad (5.24)
\end{aligned}$$

where the second inequality follows using Lipschitz continuity of ℓ and the last equality follows by the noise addition step (Step 13) of Algorithm 5.4.

From the tail bound on the norm of a Gaussian random vector, it follows that with probability at least $1 - \frac{\gamma}{2}$,

$$\|b\|_2 \leq 3\sqrt{p}\beta\sqrt{\log\frac{1}{\gamma}} \leq 12\sqrt{p}L'\frac{\log T}{T\epsilon_p}\sqrt{\log\frac{1}{\gamma}\log\frac{1}{\delta}}, \qquad (5.25)$$

where $L' = L + \epsilon_g/\|\theta^*\|_2$, L is the Lipschitz continuity parameter of ℓ. Note that in Line 5 of Algorithm 5.4 we set the strong convexity parameter $\Psi = \frac{\epsilon_g}{\|\theta^*\|_2^2}$.

Now, regret bound of IGD is given by:

$$R_{\text{IGD}}(T) = O(\|\theta^*\|_2 + \frac{L'^2}{\Psi}\log T), \qquad (5.26)$$

Thus, by combining (5.24), (5.25), (5.26), and Theorem 5.18, with probability at least $1-\gamma$,

$$\mathbb{E}_{d\sim\mathcal{P}}[\ell(\hat{\theta};d)] - \min_{\theta\in\mathcal{C}} \mathbb{E}_{d\sim\mathcal{P}}[\ell(\theta;z)] \leq \frac{\epsilon_g}{2} + C\frac{\sqrt{p}L(L+\frac{\epsilon_g}{\|\theta^*\|_2})\log T\sqrt{\log\frac{1}{\gamma}\log\frac{1}{\delta}}}{\epsilon_p T} + C\frac{(L+\frac{\epsilon_g}{\|\theta^*\|_2})^2\|\theta^*\|_2^2\log T\log\frac{\log T}{\gamma}}{\epsilon_g T},$$

where $C>0$ is a global constant.

The result now follows by bounding the RHS above by ϵ_g. □

Chapter 6

Differentially Private Model Selection and Sparse Linear Regression

6.1 Introduction

Model selection is a basic problem in machine learning and statistics. Given a data set and an collection of "models", where each model is normally a family of probability distributions, the goal is to determine the model that best "fits" the data in some sense. The choice of model could reflect a measure of complexity, such as the number of components in a mixture model, or a choice about which aspects of the data appear to be most relevant, such as the set of features used for a regression model.

In sparse linear regression problems, for example, each entry in the data set consists of a p-dimensional real *feature vector* x and real-valued *response* (or *label*) y. The overall goal is to find a parameter vector $\theta \in \mathbb{R}^p$ such that $\langle x_i, \theta \rangle \approx y_i$ for all n data points (x_i, y_i). When p is much larger than n, the problem is underdetermined and so solutions to this problem won't necessarily generalize well. A common approach is to look for a vector θ with at most s nonzero entries (where $s \ll n$) that labels the data set well. Each set of at most s positions defines a model and, for a specific model, the problem simplifies to textbook linear regression. The model selection problem is to decide which of the roughly $\binom{p}{s}$ subsets to consider. Once the subset is decided, the regression problem can be reduced to a s-dimensional problem and the results from Chapter 3 can be directly applied. In this chapter, we restrict our focus on choosing only the correct subset.

In this chapter we investigate the possibility of carrying out sophisticated model selection algorithms without leaking significant information about individuals entries in the data set. This is critical when the information in the data set is sensitive, for example if it consists of financial records or health data. Our algorithms satisfy *differential privacy* [Dwork et al., 2006b; Dwork, 2006], which essentially ensures that adding or removing an individual's data from a data set will have little effect on the inferences made about them based on an algorithm's output [Dwork, 2006; Ganta et al., 2008].

Formally, there is no reason to separate model selection from the fitting of a specific distribution of the data once the model is selected—either way, one is trying to select a best fit from among a class of probability distributions. However, the separation into two phases survives for (at least) two reasons: First, individual models are often parameterized by a finite-dimensional real vector, and so fitting a particular model to the data is a continuous optimization task. In contrast, the set of models is typically discrete, and the corresponding optimization problems tend to have a very different feel. Second, the model selection step is typically much more computationally expensive.

The question, then, is how well differentially private algorithms can do at model selection, and how to design algorithms that are computationally efficient. The challenge is to select from a large number of possible models using as few resources (samples and running time) as possible. Furthermore, we want to answer the question on fitting a specific distribution on the data (in the context of *sparse regression*) while preserving differential privacy.

Our Contributions. We consider the setting in which there is a "well-defined" answer, in the following sense: Suppose that there is nonprivate model selection procedure f, which is the reference to which we compare our performance. Our algorithms output the correct value $f(\mathcal{D})$ whenever f is *stable* on the input data set $\mathcal{D}$. We work with two notions, *perturbation* stability and *subsampling* stability.

We give two classes of results: generic ones, that apply to any function; and specific algorithms for the problem of sparse linear regression. The algorithms we describe are efficient and in some cases match the optimal asymptotic sample complexity for nonprivate algorithms.

Our algorithms for sparse linear regression require analyzing the stability properties of the popular LASSO estimator. We give sufficient conditions for the LASSO estimator to be robust to small changes in the data set, and show that these conditions hold with high probability under essentially the same stochastic assumptions that are used in the literature to analyze convergence of the LASSO. This analysis may be of independent interest. The next two sections describe our contributions in more detail.

Organization of the chapter: In Section 6.2 we study two different algorithms using which we can transform the stability guarantees about a particular model selection algorithm into differential privacy guarantees. In Section 6.3 we study the consistency and stability properties of the LASSO algorithm for feature (support) selection. In Section 6.4 we provide two different algorithms for differentially private support selection in sparse linear regression. In Section 6.5 we provide a generic framework for private sparse linear regression which outputs a parameter vector that has vanishing excess empirical risk (due to privacy) with respect to the underlying parameter vector (θ^*) for sparse linear regression. We instantiate this framework with two support selection algorithms, one based on the stability of subsampling (from Section 6.4.1) and the other based on exponential sampling (a new algorithm for support selection using the *exponential mechanism* of [McSherry and Talwar, 2007]).

6.1.1 Generic Algorithms For Stable Functions

We give two simple, generic transformations that, given any function f and parameters $\epsilon, \delta > 0$, return a (ϵ, δ)-differentially private algorithm (see Definition 2.1) that is correct whenever f is sufficiently stable on a particular input $\mathcal{D}$. The two algorithms correspond to different notions of stability. In both cases, the correctness guarantees do not have any dependence on the size of the range of f, only on the privacy parameters ϵ and δ. In the context of model selection, this implies that there is no dependency on the number of models under consideration.

- *Perturbation Stability*: We say that f is *stable* on $\mathcal{D}$ if f takes the value $f(\mathcal{D})$ on all of the neighbors of $\mathcal{D}$ (and *unstable* otherwise). We give an algorithm $\mathcal{A}_{dist}$ that, on input $\mathcal{D}$, outputs $f(\mathcal{D})$ with high probability if $\mathcal{D}$ is at distance at least $\frac{2\log(1/\delta)}{\epsilon}$ from the nearest *un*stable data set. Unfortunately, the algorithm $\mathcal{A}_{dist}$ is not efficient, in general.

- *Subsampling stability*: We say f is q-subsampling stable on $\mathcal{D}$ if $f(\hat{D}) = f(\mathcal{D})$ with probability at least $3/4$ when $\hat{D}$ is a random subsample from $\mathcal{D}$ which includes each entry independently with probability q. We give an algorithm $\mathcal{A}_{samp}$ that, on input $\mathcal{D}$, outputs $f(\mathcal{D})$ with high probability whenever f is q-subsampling stable for $q = \frac{\epsilon}{32\log(1/\delta)}$. The running time of $\mathcal{A}_{samp}$ dominated by running f about $1/q^2$ times; hence it is efficient whenever f is.

  This result has an clean statistical interpretation: Given a collection of models, let the sample complexity of model selection be the minimum number of samples (over nonprivate algorithms) from a distribution in one of the models needed to select the correct model with probability at least $2/3$. Then the sample complexity needed for differentially private model selection increases by a problem-independent factor of $O(\log(1/\delta)/\epsilon)$.

Technique: Proxies for the distance to instability The idea behind the first algorithm comes from the work of [Dwork and Lei, 2009] on private parametric estimation. If we were somehow given a *promise* that f is stable on $\mathcal{D}$, we could release $f(\mathcal{D})$ without violating differential privacy. The issue is that stability itself can change between neighboring data sets, and so stating that f is stable on $\mathcal{D}$ may violate differential privacy. The solution implicit in [Dwork and Lei, 2009] (specifically, in their algorithms for estimating interquartile distance and the median) is to instead look at the *distance* to the nearest unstable instance. This distance changes by at most 1 between neighboring data sets, and so one can release a noisy version of the distance privately, and release $f(\mathcal{D})$ when that noisy estimate is sufficiently high. Developing this simple idea leads to the algorithm $\mathcal{A}_{dist}$.

The difficulty with this approach is that it requires computing the distance to the nearest unstable instance explicitly. We observe that if one can compute a *lower bound* $\hat{d}(\mathcal{D})$ on the distance to the nearest unstable instance, and if $\hat{d}$ does not change much between neighboring data sets, then one can release a noisy version of $\hat{d}$ differentially privately, and release $f(\mathcal{D})$ when the noisy estimate is sufficiently high. *The challenge, then, is to efficiently compute useful proxies for the distance to the nearest unstable input.*

We obtain our algorithm for subsampling-stable functions by giving an efficient distance bound for a bootstrapping-based model selector $\hat{f}(\mathcal{D})$ that outputs the most commonly occurring value of f in a set of about $1/\epsilon^2$ random subsamples taken from the input $\mathcal{D}$. The approach is inspired by the "sample and aggregate" framework of [Nissim et al., 2007]. However, our analysis allows working with much larger subsamples than those in previous work [Nissim et al., 2007; Smith, 2011; Kifer et al., 2012]. In our context, the analysis from previous work would lead to a polynomial blowup in sample complexity (roughly, squaring the number of samples needed nonprivately), whereas our result increases the sample complexity by a small factor.

6.1.2 Feature Selection for Sparse Linear Regression and Robustness of the LASSO

Our second class of results concerns feature selection for sparse linear regression. Recall the task is to select a set of s out of p features to be used for linear regression. This problem provides an interesting challenge for model selection problems since the number of distinct models to be considered is enormous ($\binom{p}{s}$, where we want to make s as large as possible without losing statistical validity).

Given a data set of n entries $(x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}$, let $X \in \mathbb{R}^{n \times p}$ be the matrix with rows x_i and $y \in \mathbb{R}^p$ be the column vector with entries y_i. Suppose that the data set satisfies a linear system

$$y = X\theta^* + w \tag{6.1}$$

where θ^* is a parameter vector (in $\mathbb{R}^p$) to be estimated, and $w \in \mathbb{R}^{n \times 1}$ is an error vector whose entries are assumed to be small. We say a vector is s-sparse if has at most s nonzero entries. The problem we consider is: assuming that θ^* is s-sparse, *under what conditions can we recover the support of θ^* while satisfying differential privacy?*

The nonprivate version of this problem has been studied extensively in the literature on high-dimensional statistics and compressed sensing. Several works [Zhao and Yu, 2007; Wainwright, 2006; Negahban et al., 2010] have shown that $n = O(s \log p)$ samples suffice to recover the support of θ^*, assuming the data are drawn i.i.d. from one of a fairly large class of probability distributions. Moreover, this bound is known to be asymptotically tight [Wainwright, 2006; Raskutti et al., 2011].

Differentially private algorithms for sparse regression were first considered in our recent work ([Kifer et al., 2012]). They gave feature selection procedures that require $n \gg s \log(p) \cdot (\min\{s, \log p\})$ samples to recover the support of θ^*. Matching the optimal sample complexity (under reasonable assumptions) was left as an open problem.

We give two efficient algorithms that approach the optimal sample complexity $s \log p$.

- Our results on subsampling stability imply immediately that one can get efficient differentially private algorithms with sample complexity $O(\frac{\log(1/\delta)}{\epsilon} s \log p)$ under the same stochastic assumptions used in nonprivate upper bounds. This is already a significant improvement over the previous work [Kifer et al., 2012]. However, it retains the multiplicative dependence on $\log(1/\delta)/\epsilon$.

- We also give explicit estimators for the distance to instability of a popular technique for sparse regression known as the Lasso (as well as a more robust vari-

ant). This allows us to get efficient differentially private algorithms with *optimal* sample complexity $O(s \log p)$ — removing the dependence on ϵ and δ —when p is very large. More specifically, we derive algorithms with sample complexity $n = O(\max\{s \log p, k^2 s^4 / \log(p), k s^{3/2}\})$, where $k = \log(1/\delta)/\epsilon$. This is $O(s \log p)$ when $s < \frac{\log^{2/3} p}{k^{2/3}}$ and $k < \log^{2/3} p$. Note that it is interesting to come up with good model selectors even when s is constant, since p may be very large.

Techniques: Stability and Robustness of the LASSO The (efficient, nonprivate) upper bounds on feature selection for sparse linear regression derive mostly from analyses of a popular approach known as the Lasso. The idea is to find an estimate $\hat{\theta}$ of θ^* which is sparse and which minimizes some norm of the estimated error $\hat{w} = y - X\hat{\theta}$. This is done by penalizing the usual mean squared error loss with some multiple of the L_1 norm of θ:

$$\hat{\theta}(\mathcal{D}) = \arg\min_{\theta \in \mathcal{C}} \frac{1}{2n} \|y - X\theta\|_2^2 + \frac{\Lambda}{n} \|\theta\|_1 \tag{6.2}$$

The consistency properties of the Lasso are well-studied: a variety of assumptions on the data, when $n = \omega(s \log p)$, the estimate $\hat{\theta}$ is known to converge to θ^* in the L_2 norm [Wainwright, 2006; Negahban et al., 2010]. Moreover, if the entries of θ^* are bounded away from zero, $\hat{\theta}$ will have the same support as θ^* [Wainwright, 2006].

We extend these results to show that, *under essentially the same assumptions*, the support of $\hat{\theta}$ does not change when a small number of data points are changed. Other work on LASSO robustness captures different properties. (See Section 6.1.3 below.) Our analysis requires significantly refining the "primal-dual" construction technique of [Wainwright, 2006]. The idea is to show that an optimal solution to (6.2) for data set $\mathcal{D}'$ which is "near" $\mathcal{D}$ can be transformed into an optimal solution for $\mathcal{D}$. This involves analyzing how the KKT conditions on the subgradient of the nondifferentiable loss function in (6.2) change as the data varies.

Significantly, we also use the primal-dual analysis to give an *efficient* and smooth estimator for the distance from a given data set $\mathcal{D}$ to the nearest unstable data set. The estimator essentially uses the subgradient of the regularized loss (6.2) to measure how big a change would be needed to one of the zero entries of $\hat{\theta}$ to "jump" away from zero. This is delicate because changing the data set changes both the minimizer and the geometry of the loss function. The efficient distance estimator gives us the differentially private feature selector with optimal sample complexity.

Assumptions As mentioned, our analyses of stability make various assumptions about the data. First, it is important to note that the assumptions are made only for the utility analysis. *The privacy guarantees are unconditional.* Second, we distinguish between "fixed data" assumptions, which give deterministic conditions the data set for a given algorithm to perform well, and "stochastic" assumptions, which give conditions on a distribution from which the data are drawn i.i.d. We analyze the Lasso's robustness under essentially the same assumptions (fixed-data and stochastic) used in previous work to analyze consistency. The difference is that we require certain constants to be larger, leading to a constant factor increase in sample complexity.

6.1.3 Prior Work on Learning and Stability

The relationship between learning, statistics and stability has been studied in the learning theory literature (e.g., [Rogers and Wagner, 1978]) and in robusts statistics (e.g., [Huber, 1981]) for over thirty years. Many variants of stability have been studied, and the literature is too vast to survey here. We highlight only the most relevant works.

The main difference with our work is that the learning literature focuses on algorithms whose output lies in a metric space; stability measures how much the output changes under various models of perturbation, and the focus is on settings where some change in unavoidable even for very "nice" data sets [Bousquet and Elisseeff, 2002; Shalev-Shwartz et al., 2010; Ben-David et al., 2006; Rakhlin and Caponnetto, 2007]. Several papers on privacy have sought to exploit such stability properties for privacy purposes [Dwork and Lei, 2009; Rubinstein et al., 2009; Chaudhuri and Hsu, 2012]. In contrast, we look at settings where some discrete structure may remain unchanged under perturbations. This is effectively a stronger assumption, leading to tighter sample complexity bounds.

Both notions of peturbation that we conisder have been studied previously, namely robustness to changes in the input data set $\mathcal{D}$ [Bousquet and Elisseeff, 2002; Shalev-Shwartz et al., 2010; Xu et al., 2010; Dwork and Lei, 2009] and stability to subsampling or resampling from the training data set $\mathcal{D}$ [Shao, 1996; Bach, 2008; Meinshausen and Bhlmann, 2006; Meilă, 2006].

Robustness to small changes in the input data was studied both to provide resilience to outliers and noise (as in robust statistics) as well as to get good generalization error [Bousquet and Elisseeff, 2002; Shalev-Shwartz et al., 2010]. One consequence of these works is that if a learning algorithm f satisfies our notion of stability, then it generalizes well. Perhaps the most relevant work in this line is by [Xu et al., 2010], who study the L_2 robustness of Lasso-like estimators to small perturbations, and show that *uniform* stability (in which the set of selected features changes by only small steps between *any pairs* of neighbors) is impossible for algorithms with sparse output. Finally, [Lee et al., 2011] look at Huberized versions of the LASSO with the goal of providing robustness, but no not provide formal consistency or convergence guarantees.

Stability under subsampling and resampling were also studied extensively in the prior work [Shao, 1996; Bach, 2008; Meinshausen and Bhlmann, 2006; Meilă, 2006]. In particular, they were used for model selection and clustering [Meilă, 2006]. Again, their notion of stability is weaker than ours.

6.2 Stability and Privacy

Consider a function $f : \mathcal{T}^* \to \mathcal{R}$ from data sets to a range $\mathcal{R}$. We assume that the range $\mathcal{R}$ is finite, for simplicity.

Definition 6.1. *A function $f : \mathcal{T}^* \to \mathcal{R}$ is* k-stable on input $\mathcal{D}$ *if adding or removing any k elements from $\mathcal{D}$ does not change the value of f, that is, $f(D) = f(D')$ for all $\mathcal{D}'$ such that $|\mathcal{D} \triangle \mathcal{D}'| \leq k$. We say f is* stable *on $\mathcal{D}$ if it is (at least) 1-stable on $\mathcal{D}$, and* unstable *otherwise.*

The distance to instability *of a data set $\mathcal{D} \in \mathcal{T}^*$ with respect to a function f is the number of elements that must be added to or removed from $\mathcal{D}$ to reach an data set that*

is not stable. Note that $\mathcal{D}$ is k-stable iff its distance to instability is at least k.

A First Attempt For any function f, there is a differentially private algorithm $\mathcal{A}_{dist}$ that outputs $f(\mathcal{D})$ whenever $\mathcal{D}$ is sufficiently stable. It follows the lines of more general approaches from previous work [Dwork and Lei, 2009; Karwa et al., 2011] that calibrate noise to differentially private estimates of local sensitivity. The algorithm is not efficient, in general, but it is very simple: On input $\mathcal{D}$ and parameters $\epsilon, \delta > 0$, $\mathcal{A}_{dist}$ computes the distance d from $\mathcal{D}$ to the nearest unstable instance, and add $\text{Lap}(1/\epsilon)$ noise to get an estimate $\tilde{d}$ of d. Finally, if $\tilde{d} > \frac{\log(1/\delta)}{\epsilon}$, then it releases $f(\mathcal{D})$, otherwise it outputs a special symbol $\perp$.

Prosition 6.2. *For every function f:*

1. *$\mathcal{A}_{dist}$ is (ϵ, δ)-differentially private.*
2. *For all $\beta > 0$: if f is $\frac{\log(1/\delta)+\log(1/\beta)}{\epsilon}$-stable on $\mathcal{D}$, then $A(\mathcal{D}) = f(\mathcal{D})$ with probability at least $1 - \beta$.*

For brevity, we defer the proof of this proposition to Section 6.2.2.1.This result based on distance is the best possible, in the following sense: if there are two data sets $\mathcal{D}_1$ and $\mathcal{D}_2$ for which $\mathcal{A}$ outputs different values $f(\mathcal{D}_1)$ and $f(\mathcal{D}_2)$, respectively, with at least constant probability, then the distance from $\mathcal{D}_1$ to $\mathcal{D}_2$ must be $\Omega(\log(1/\delta)/\epsilon)$.

However, there are two problems with this straightforward approach. First, the algorithm is not efficient, in general, since it may require searching all data sets within distance up to d from $\mathcal{D}$ (this may not even be implementable at all if U is infinite). Second, the model selection algorithm given to us may not be stable on the instances of interest.

More Robust Functions, and Efficient Proxies for Distance We remedy these problems by (a) modifying the functions to obtain a more stable function $\hat{f}$ that equals f on "nice" inputs, and (b) designing efficient, private estimators for the distance to instability with respect to $\hat{f}$.

We combine these two goals into a single definition: we are looking for a pair of functions $\hat{f}, \hat{d}$ that act as proxies for f and the stability of f, respectively. We measure the usefulness of the pair by a set $\mathcal{N}$ of "nice" inputs on which this pair allows us to release the actual value f.

Definition 6.3. *Given $f : \mathcal{T}^* \to \mathcal{R}$, a pair of functions $\hat{f} : \mathcal{T}^* \to \mathcal{R}$, $\hat{d} : \mathcal{T}^* \to \mathbb{R}$ are* proxies *for f and its stability which are accurate on a set $\mathcal{N}$ (which depends on parameters ϵ, δ) if the following hold:*

1. *For all $\mathcal{D}$: $\hat{d}(\mathcal{D}) \leq$ (dist. of $\mathcal{D}$ to instability of $\hat{f}$).*
2. *$GS_{\hat{d}} \leq 1$*
3. *For all $\mathcal{D} \in \mathcal{N}$: $f(\mathcal{D}) = \hat{f}(\mathcal{D})$ and $\hat{d}(\mathcal{D}) \geq 2\log(1/\delta)/\epsilon$.*

One can use such a proxy by adding Laplace noise to $\hat{d}$ and releasing $\hat{f}(\mathcal{D})$ whenever the noisy version of $\hat{d}$ is sufficiently large. The resulting mechanism will be (ϵ, δ)-differentially private and, on all inputs $\mathcal{D} \in \mathcal{N}$, will release $f(\mathcal{D})$ with probability at least $1 - \delta$.

For every function f, one can get a valid proxy by letting $\hat{f} = f$ and letting $\hat{d}(\mathcal{D})$ be the distance to instability of $\mathcal{D}$ w.r.t. f. The set $\mathcal{N}$ of good instances for this pair is exactly the set of inputs $\mathcal{D}$ on which f is $\frac{2\log(1/\delta)}{\epsilon}$-stable. As mentioned above, the main problem is computational efficiency.

Given a function f, the goal is to find proxies $(\hat{f}, \hat{d})$ that are efficient (ideally, as efficient as evaluating f alone) and have as large as possible a set $\mathcal{N}$ of good inputs.

6.2.1 From Sampling Stability to Stability

We give a generic construction that takes any function f and produces a pair functions $(\hat{f}, \hat{d})$ that are efficient—they take essentially the same time to evaluate as f—and are accurate for data sets on which the original f is *subsampling stable*.

Definition 6.4 (Subsampling stability)**.** *Given a data set $\mathcal{D} \in \mathcal{T}^*$, let $\hat{\mathcal{D}}$ be a random subset of $\mathcal{D}$ in which each element appears independently with probability q. We say f* is q-subsampling stable on input $\mathcal{D} \in \mathcal{T}^*$ *if $f(\hat{\mathcal{D}}) = f(\mathcal{D})$ with probability at least $3/4$ over the choice of $\hat{\mathcal{D}}$.*

The algorithm $\mathcal{A}_{samp}$ (Algorithm 6.1) uses bootstrapping to create a modified function $\hat{f}$ that equals $f(\mathcal{D})$ and is far from unstable on a given $\mathcal{D}$ whenever f is subsampling stable on $\mathcal{D}$. The output of $\hat{f}(\mathcal{D})$ is the mode (most frequently occurring value) in the list $F = (f(\hat{\mathcal{D}}_1), ..., f(\hat{\mathcal{D}}_m))$ where the $\hat{\mathcal{D}}_i$'s are random subsamples of size about $\epsilon n / \log(1/\delta)$. The distance estimator $\hat{d}$ is, up to a scaling factor, the difference between the frequency of the mode and the next most frequent value in F. Following the generic template in the previous section, the algorithm $\mathcal{A}_{samp}$ finally adds Laplace noise to $\hat{d}$ and outputs $\hat{f}(\mathcal{D})$ if the noise distance estimate is sufficiently high.

We summarize the properties of $\mathcal{A}_{samp}$ below.

Theorem 6.5.

1. *Algorithm $\mathcal{A}_{samp}$ is (ϵ, δ)-differentially private.*
2. *If f is q-subsampling stable on input $\mathcal{D}$ where $q = \frac{\epsilon}{32\log(1/\delta)}$, then algorithm $\mathcal{A}_{samp}(\mathcal{D})$ outputs $f(\mathcal{D})$ with probability at least $1 - 3\delta$.*
3. *If f can be computed in time $T(n)$ on inputs of length n, then $\mathcal{A}_{samp}$ runs in expected time*
   $O(\frac{\log n}{q^2})(T(qn) + n)$.

Note that the utility statement here is an input-by-input guarantee; f need not be subsampling stable on all inputs. *Importantly, there is no dependence on the size of the range $\mathcal{R}$.* In the context of model selection, this means that one can efficiently satisfy differential privacy with a modest blow-up in sample complexity (about $\log(1/\delta)/\epsilon$) whenever there is a particular model that gets selected with reasonable probability.

Algorithm 6.1 $\mathcal{A}_{samp}$: Bootstrapping for Subsampling-Stable f

Require: dataset: $\mathcal{D}$, function $f : \mathcal{T}^* \to \mathcal{R}$, privacy parameters $\epsilon, \delta > 0$.

$q \leftarrow \frac{\epsilon}{32\log(1/\delta)}$, $m \leftarrow \frac{\log(n/\delta)}{q^2}$.
repeat
  Subsample m data sets $\hat{\mathcal{D}}_1, ..., \hat{\mathcal{D}}_m$ from $\mathcal{D}$, where $\hat{\mathcal{D}}_i$ includes each position of $\mathcal{D}$ independently w.p. q.
until each position of $\mathcal{D}$ appears in at most $2mq$ sets $\hat{\mathcal{D}}_i$
Compute $F = \langle f(\hat{\mathcal{D}}_1), \cdots, f(\hat{\mathcal{D}}_m)\rangle$.
For each $r \in \mathcal{R}$, let $count(r) = \#\{i : f(\hat{\mathcal{D}}_i) = r\}$.
$\hat{d} \leftarrow (count_{(1)} - count_{(2)})/(4mq) - 1$ where $count_{(1)}, count_{(2)}$ are the two highest counts from the previous step.
$\tilde{d} \leftarrow \hat{d} + \mathrm{Lap}(\frac{1}{\epsilon})$.
if $\tilde{d} > \log(1/\delta)/\epsilon$ **then**
  Output $\hat{f}(\mathcal{D}) = mode(F)$.
else
  Output $\perp$.
end if

Previous work in data privacy has used the idea of bootstrapping or subsampling to convert from various forms of subsampling stability to some sort of stability [Nissim et al., 2007; Dwork and Lei, 2009; Smith, 2011; Kifer et al., 2012]. The main advantage of the version we present here is that size of the subsamples is quite large: our algorithm requires a blowup in sample complexity of about $\log(1/\delta)/\epsilon$, independent of the size of the output range $\mathcal{R}$, as opposed to previous algorithms that had blowups polynomial in n and some measure of "dimension" of the output.

The following lemma provides the key to analyzing our approach. The main observation is that the stability of the *mode* is a function of the difference between the frequency of the mode and the next most frequent element. The lemma roughly says that if f is subsampling stable on $\mathcal{D}$, then $\mathcal{D}$ is far from unstable w.r.t. $\hat{f}$ (not necessarily w.r..t f), and moreover one can estimate the distance to instability of $\mathcal{D}$ *efficiently* and privately. Proof of this lemma is deferred to Section 6.2.2.2 for brevity.

Lemma 6.6. *Fix $q \in (0, 1)$. Given $f : \mathcal{T}^* \to \mathcal{R}$, let $\hat{f} : \mathcal{T}^* \to \mathcal{R}$ be defined as $\hat{f}(\mathcal{D}) = mode(f(\hat{\mathcal{D}}_1), ..., f(\hat{\mathcal{D}}_m))$ where each $\hat{\mathcal{D}}_i$ includes elements of $\mathcal{D}$ independently w.p. q and $m = \log(n/\delta)/q^2$. Let $\hat{d}(\mathcal{D}) = (count_{(1)} - count_{(2)})/(4mq) - 1$. Fix a data set $\mathcal{D}$. Let E be the event that no position of $\mathcal{D}$ is included in more than $2mq$ of the subsets $\hat{\mathcal{D}}_i$.*

1. *E occurs with probability at least $1 - \delta$.*
2. *Conditioned on E, the pair $(\hat{f}, \hat{d})$ are a good proxy for f and its stability (that is, $\hat{d}$ lower bounds the stability of $\hat{f}$ on $\mathcal{D}$, and $\hat{d}$ has global sensitivity 1).*
3. *If f is q-subsampling stable on $\mathcal{D}$, then with probability at least $1-\delta$ over the choice of subsamples, we have $\hat{f}(\mathcal{D}) = f(\mathcal{D})$, and $\hat{d}(\mathcal{D}) \geq 1/16q$.*

The events (2) and (3) occur simultaneously with probability at least $1 - 2\delta$.

Theorem 6.5 follows from the lemma by noting that for small enough q, the function d, which acts as an efficient proxy for stability, will be large enough that even after adding Laplace noise one can tell that $\hat{f}$ is stable on instance $\mathcal{D}$, and release f.

6.2.2 Missing Proof in Section 6.2 (Stability and Privacy)

6.2.2.1 Proof of Proposition 6.2

Proof of part (1). Note that Algorithm $\mathcal{A}_{dist}$ can have only two possible outputs: $\perp$ or $f(\mathcal{D})$. We show that for each of the outputs, the differential privacy condition holds. Firstly, since the true distance d can change by at most one if one entry is removed (added) from (to) the data set $\mathcal{D}$, therefore, by the following theorem (*Laplace mechanism*) from [Dwork et al., 2006b], the variable $\tilde{d}$ (in Algorithm $\mathcal{A}_{dist}$) satisfies $(\epsilon, 0)$-differential privacy.

Theorem 6.7 (Laplace Mechanism [Dwork et al., 2006b])**.** *Let* $f : \mathcal{T}^* \to \mathbb{R}$ *be a function (with* $\mathcal{T}^*$ *being the domain of data sets). If for any pair of data sets* $\mathcal{D}$ *and* $\mathcal{D}'$ *with symmetric difference at most one,* $|f(\mathcal{D}) - f(\mathcal{D}')| \leq 1$*, then the output* $\mathcal{A}(\mathcal{D}) = f(\mathcal{D}) + Lap\left(\frac{1}{\epsilon}\right)$ *is* $(\epsilon, 0)$*-differentially private.*

Since we have shown $\tilde{d}$ is $(\epsilon, 0)$-differentially private, it follows that for any pair of data sets $\mathcal{D}$ and $\mathcal{D}'$ differing in one entry, differential privacy condition holds for the output $\perp$, i.e.,

$$\Pr[\mathcal{A}_{\text{dist}}(\mathcal{D}) = \perp] \leq e^{\epsilon} \Pr[\mathcal{A}_{\text{dist}}(\mathcal{D}') = \perp]$$

Notice that by the tail property of Laplace distribution, it follows that if $\tilde{d} > \frac{\log(1/\delta)}{\epsilon}$, then with probability at least $1 - \delta$ the actual distance d is greater than zero. Define the event E equal to be true, if the noise $\text{Lap}(1/\epsilon)$ is greater than $\frac{1}{\epsilon}\log(1/\delta)$. Then, we have,

$$\begin{aligned}
\Pr[\mathcal{A}_{\text{dist}}(\mathcal{D}) = f(\mathcal{D})] &\leq \Pr[\mathcal{A}_{\text{dist}}(\mathcal{D}) = f(\mathcal{D}) \wedge \bar{E}] + \Pr[E] \\
&\leq \Pr[\mathcal{A}_{\text{dist}}(\mathcal{D}') = f(\mathcal{D}) \wedge \bar{E}] + \delta \\
&\leq \Pr[\mathcal{A}_{\text{dist}}(\mathcal{D}') = f(\mathcal{D})] + \delta
\end{aligned}$$

Thus, we can conclude that Algorithm $\mathcal{A}_{\text{dist}}$ is (ϵ, δ)-differentially private.

□

Proof of Part (2). By the tail property of Laplace distribution, if the true distance d is at least $\frac{1}{\epsilon}(\log(1/\delta) + \log(1/\beta))$, then with probability at least $1 - \beta$, the noisy distance $\tilde{d}$ is greater than $\frac{1}{\epsilon}\log(1/\delta)$. Hence with probability at least $1 - \beta$, $f(\mathcal{D})$ is output. □

6.2.2.2 Proof of Lemma 6.6

Proof. Proof of part (1) of the lemma follows by a direct application of Chernoff-Hoeffding's bound. To prove part (2), notice that conditioned on the event E adding or removing one entry in the original data set changes any of the counts $count_{(r)}$ by at most

$2mq$. Therefore, $count_{(1)} - count_{(2)}$ changes by at most $4mq$. This in turn means that $\hat{d}(\mathcal{D})$ changes by at most one for any $\mathcal{D}$ and hence have global sensitivity of one. This also implies that $\hat{d}$ lower bounds the stability of $\hat{f}$ on $\mathcal{D}$. To prove part (3), notice that when $\hat{d}(D) \geq 1/16q$, it implies that $count_{(1)} - count_{(2)} \geq m/4$. Thus, if we bound the probability of the highest bin having count less than $5/8m$ by $1 - \delta$, then we are done. Recall that in expectation the highest bin has count at least $3/4m$. Now the remaining proof follows directly via the application of Chernoff-Hoeffding's bound. □

6.3 Consistency and Stability of Sparse Regression using LASSO

Recall the linear system in (6.1). A common approach for estimating the underlying parameter vector θ^* is via least-squared regression, where the loss function in the regression problem is $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{2n}\|y - X\theta\|_2^2$, where the data set $\mathcal{D} = (y, X)$ and has size n. When the dimensionality of the problem (p) is larger than the data set size (n), a common approach is to add an L_1 penalty term to the loss function to encourage selection of sparse minimizers. This formulation is commonly called LASSO *(Least Absolute Shrinkage and Selection Operator)* [Tibshirani, 1996]. The formal optimization problem corresponding to the current formulation is given in (6.3). Here $\mathcal{C} \subseteq \mathbb{R}^p$ is some fixed convex set and Λ is some regularization parameter.

$$\hat{\theta}(\mathcal{D}) = \arg\min_{\theta \in \mathcal{C}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Lambda}{n}\|\theta\|_1 \tag{6.3}$$

[Wainwright, 2006; Negahban et al., 2010] showed that under certain "niceness" conditions on the data set $\mathcal{D} = (y, X)$ and the underlying parameter vector θ^*, the support of $\hat{\theta}(\mathcal{D})$ equals the support of θ^* and $\|\hat{\theta}(\mathcal{D}) - \theta^*\|_2$ goes down to zero as n goes to infinity. This property is often referred to as *consistency*. In this work we revisit the consistency assumptions (in [Wainwright, 2006]) for LASSO and relate the two different sets of assumptions sufficient for consistency, namely, *fixed data* and *stochastic* assumptions. Additionally, we weaken the *fixed data* assumptions that are sufficient for consistency.

An important property of any algorithm is the *stability* of its output with respect any changes in its input data. In this work we study the stability properties of the support of the minimizer of a LASSO program. We follow a very strong notion of stability where the changes in the data set can be addition or removal of any constant (k) number of entries. At a high level, we show that *almost* under the same set of "niceness" conditions for consistency one can also guarantee stability.

In this work we study the consistency and stability properties of LASSO (and one of its variants) in two different settings: i) *fixed data setting* where the data set $\mathcal{D}$ is deterministic, and ii) *stochastic* where the dataset $\mathcal{D}$ is drawn from some underlying distribution. The general flavor of our results in this section is that we first prove the consistency and stability properties in the fixed data setting and then show one particular stochastic setting which satisfies the fixed data assumptions with high probability. The fixed data assumptions are given in Assumption 6.8 below.

Assumption 6.8 (Typical system)**.** *Data set* $(X_{n\times p}, y_{n\times 1})$ *and parameter vector* $\theta^* \in \mathbb{R}^p$

are (s, Ψ, σ, Φ)-TYPICAL *if there exists a* $w \in \mathbb{R}^p$ *such that* $y = X\theta^* + w$ *and*

1. ***Column normalization:*** $\forall j, \|c_j\|_2 \leq \sqrt{n}$, *where* c_j *is the* j*-th column of* X.
2. ***Bounded parameter vector:*** $\|\theta^*\|_0 \leq s$ *and all nonzero entries of* θ^* *have absolute value in* $(\Phi, 1 - \Phi)$.
3. ***Incoherence:*** *Let* Γ *be the support of* θ^*. $\|({X_{\Gamma^c}}^T X_\Gamma)({X_\Gamma}^T X_\Gamma)^{-1} sign(\theta^*)\|_\infty < \frac{1}{4}$. *Here* $\Gamma^c = [p] - \Gamma$ *is the complement of* Γ; X_Γ *is the matrix formed by the columns of* X *whose indices are in* Γ; *and* $sign(\theta^*) \in \{-1, 1\}^{|\Gamma|}$ *is the vector of signs of the nonzero entries in* θ^*.
4. ***Restricted Strong Convexity:*** *The minimum eigenvalue of* ${X_\Gamma}^T X_\Gamma$ *is at least* Ψn.
5. ***Bounded Noise:*** $\|X_{\Gamma^c}^T V w\|_\infty \leq 2\sigma\sqrt{n \log p}$, *where* $V = \mathbb{I}_{n \times n} - X_\Gamma({X_\Gamma}^T X_\Gamma)^{-1}{X_\Gamma}^T$ *is the projector on to the complement of the column space of* X_Γ.

6.3.1 Consistency of LASSO Estimator

Under a strengthened version of the *fixed data conditions* above (Assumption 6.8), [Wainwright, 2006] showed that one can correctly recover the exact support of the parameter vector θ^* and moreover the estimated parameter vector $\hat{\theta}(\mathcal{D})$ is close to θ^* in the L_2 metric. Theorem 6.9 restates the result of [Wainwright, 2006] in the context of this work. We note that the result of [Wainwright, 2006] holds even under this weaker assumption (Assumption 6.8).

Theorem 6.9 (Modified Theorem 1 of [Wainwright, 2006])**.** *Let* $\Lambda = 4\sigma\sqrt{n \log p}$. *If there exists a* θ^* *such that* (X, y, θ^*) *is* (s, Ψ, σ, Φ)-TYPICAL *with* $\Phi = \frac{16\sigma}{\Psi}\sqrt{\frac{s \log p}{n}}$, *then* $\|\hat{\theta}(\mathcal{D}) - \theta^*\|_2 \leq \frac{8\sigma}{\Psi}\sqrt{\frac{s \log p}{n}}$. *Moreover, the support of* $\hat{\theta}(\mathcal{D})$ *and* θ^* *are same.*

Along with the fixed data setting, [Wainwright, 2006] considered the *stochastic setting* where the rows of the design matrix X are drawn from $\mathcal{N}(0, \frac{1}{4}\mathbb{I}_p)$ and the noise vector w is drawn independently from a mean zero sub-Gaussian distribution with variance σ^2. They showed that with high probability, under such a setting and choosing $\Lambda = 4\sigma\sqrt{n \log p}$, one has $\|\hat{\theta}(\mathcal{D}) - \theta^*\|_2 \leq \frac{8\sigma}{\Psi}\sqrt{\frac{s \log p}{n}}$ and the support of $\hat{\theta}(\mathcal{D})$ and θ^* are same. The analysis of [Wainwright, 2006] in the stochastic setting relies on a different set of arguments compared to the arguments for the fixed data setting in Theorem 6.9. Moreover, it is not clear apriori if *any* stochastic setting satisfies the fixed data setting conditions with high probability. In the following theorem, we connect the stochastic and the fixed data setting, i.e., we show that under the stochastic setting considered above, with high probability, the data set $\mathcal{D} = (y, X)$ satisfies the fixed data conditions. It should be mentioned here that [Wainwright, 2006] considered a more general distribution $\mathcal{N}(0, \Sigma)$ (with a specific class of covariance matrices). But for the purposes of brevity, we stick to the simpler $\mathcal{N}(0, \frac{1}{4}\mathbb{I}_p)$ distribution.

Theorem 6.10 (Stochastic Consistency)**.** *Let* $\Lambda = 4\sigma\sqrt{n \log p}$ *and* $n = \omega(s \log p)$*. If each row of the design matrix* X *be drawn i.i.d. from* $\mathcal{N}(0, \frac{1}{4}\mathbb{I}_p)$ *and each entry of the noise vector* w *be drawn i.i.d. from a mean zero sub-Gaussian distribution with variance* σ^2*, then there exists a constant* Ψ *such that with probability at least* $3/4$*, the data set* $\mathcal{D} = (y, X)$ *obtained via* (6.1) *and under permissible choices of* θ^* *in Assumption 6.8,* (y, X, θ^*) *satisfies* (s, Ψ, σ, Φ)-TYPICAL *with* $\Phi = \frac{16\sigma}{\Psi}\sqrt{\frac{s \log p}{n}}$.

Proof of Theorem 6.10 (Stochastic Consistency). In the following we show that each of the Conditions 1, 3, 4, and 5 in Assumption 6.8 are satisfied with probability at least 15/16. By union bound over the failure probabilities of these events, this will straightaway imply Theorem 6.10.

- **Column normalization condition:** Since we assumed $n = \Omega(s \log p)$, by tail bound over the norm of random Gaussian vectors, with probability at least 15/16, the *column normalization condition* is satisfied.

- **Restricted strong convexity (RSC):** By Proposition 1 from [Raskutti et al., 2011], it directly follows that there exists a constant Ψ such that with probability at least 15/16 the minimum eigenvalue of $X_\Gamma^T X_\Gamma$ is at least Ψn.

- **Incoherence:** Let us represent the vector $(X^T{}_\Gamma X_\Gamma)\text{sign}(\theta^*)$ to be u. Recall that by definition $\|\text{sign}(\theta^*)\|_\infty \leq 1$. Hence, by the RSC property above, $\|u\|_2 \leq \frac{\sqrt{s}}{\Psi n}$, which implies that $\|u\|_\infty \leq \frac{\sqrt{s}}{\Psi n}$.

  Let a_i be the i-th column of the matrix X_{Γ^c} and b_i be the i-th column of the matrix X_Γ Now for any row $j \in [p - s]$,

  $$|\left(X^T{}_{\Gamma^c} X_\Gamma u\right)_j| = \left|\sum_{i\in[s]} u_i \langle a_j, b_i\rangle\right| = \left|\langle a_j, \sum_{i\in[s]} u_i b_i\rangle\right| \tag{6.4}$$

  Notice that $\sum_{i\in[s]} u_i b_i = X_\Gamma u$. Therefore, $\|\sum_{i\in[s]} u_i b_i\|_2 \leq |\text{largest singular value of } X_\Gamma| \cdot \|u\|_2$. It is well known from random matrix theory that with probability at least $1 - e^{-n}$, the largest singular value of X_Γ is at most $\sqrt{n}$. Therefore, it follows that $\|\sum_{i\in[s]} u_i b_i\|_2 \leq \frac{1}{\Psi}\sqrt{\frac{s}{n}}$. Since $a_j \sim \mathcal{N}(0, \frac{1}{4}\mathbb{I}_p)$, $\left|\langle a_j, \sum_{i\in[s]} u_i b_i\rangle\right|$ in (6.4) is sub-Gaussian with standard deviation at most $\frac{1}{\Psi}\sqrt{\frac{s}{n}}$. Therefore by the tail property of sub-Gaussian random variables, with probability at most $\frac{1}{p}$, $\left|\langle a_j, \sum_{i\in[s]} u_i b_i\rangle\right| \leq \frac{1}{\Psi}\sqrt{\frac{s \log p}{n}}$. Taking union bound over all the possible columns in X_{Γ^c}, as long as $n = \omega(s \log p)$, we obtain the required *incoherence* condition with probability at least 15/16.

- **Bound** $\|X_{\Gamma^c}^T V w\|_\infty \leq 2\sigma\sqrt{n \log p}$**:** From the column normalization condition, we know that with probability at least 15/16 each column of X_{Γ^c} has L_2-norm of at most $\sqrt{n}$. Let $\tilde{a}_i$ be the random variable for the $i \in [p - s]$-th entry of the vector $X_{\Gamma^c}^T V w$. Notice that (over the randomness of w) $\tilde{a}_i$ is sub-Gaussian with standard deviation at most $\sigma\sqrt{n}$. Therefore, using the tail property of sub-Gaussian random

variables and taking an union bound over all the columns of X_{Γ^c}, with probability at least 15/16, we get the required bound $\|X_{\Gamma^c}^T V w\|_\infty \leq 2\sigma\sqrt{n \log p}$.

□

6.3.2 Normalization

Recall the linear system defined in (6.1). In the rest of this chapter, we assume following normalization bounds on the data set $\mathcal{D} = (y, X)$ and the underlying parameter vector θ^*. We assume that θ^* is from the convex set $\mathcal{C} = \{\theta : \|\theta\|_\infty \leq 1\}$. Often we restrict the convex set $\mathcal{C}$ to a support Γ (represented by $\mathcal{C}_\Gamma$). The set $\mathcal{C}_\Gamma$ is set of all vectors in $\mathcal{C}$ whose coordinates are zero outside Γ. Notice that since $\mathcal{C}$ is convex, $\mathcal{C}_\Gamma$ is also convex.

We assume that each entry of the design matrix X has absolute value at most one, i.e., $\|X\|_{\max} \leq 1$ and additionally we assume that the response vector y has L_∞-norm at most s, i.e., $\|y\|_\infty \leq s$. Notice that bounding the L_∞-norm of y is without loss of generality, since when the design matrix X and the parameter vector θ^* are bounded as above, bounding y will only decrease the noise. In case the data set $\mathcal{D} = (y, X)$ does not satisfy the above bound we *normalize* the data set to enforce such a bound. By *normalizing* we mean scaling down each data entry individually, so that they satisfy the above bound. For clarity of exposition, in the rest of this chapter we define the universe of data sets $\mathcal{T}^*$ to be sets of entries from this domain and we will assume this normalization to be implicit in all the algorithms we state (unless mentioned otherwise).

6.3.3 Stability of LASSO Estimator in the Fixed Data Setting

In Section 6.3.1 we saw that under certain "niceness" conditions (Assumption 6.8) on the data set $\mathcal{D} = (y, X)$, with suitable choice of regularization parameter Λ, one can ensure that the support of $\hat{\theta}(\mathcal{D})$ equals the support of θ. Moreover, $\|\hat{\theta}(\mathcal{D}) - \theta^*\|_2$ goes down to zero as $n \to \infty$ as long as $n = \omega(s \log p)$. In this section we ask the following question: *"Under what (further) assumptions on the data set $\mathcal{D}$ and the parameter vector θ^*, the support of the minimizer $\hat{\theta}(\mathcal{D})$ does not change even if a constant k number of entries from the domain U are either added or removed from $\mathcal{D}$?"*

We answer this question in two different settings. In the first setting we analyze the stability properties of the original LASSO program in (6.3) where we show that under assumptions very similar to the one for consistency, the support of the minimizer $\hat{\theta}(\mathcal{D})$ is also stable. In the second setting we huberize the LASSO program (i.e., transform the program to make sure that the gradient of the objective function is always bounded.) This enables us to get better stability guarantees without compromising on the correctness of support selection.

6.3.3.1 Stability of unmodified LASSO

We show that under Assumption 6.8, the support of the minimizer $\hat{\theta}(\mathcal{D})$ in (6.3) does not change even if k data entries are removed or added to $\mathcal{D}$ as long as $n = \omega(s \log p, \frac{s^4 k^2}{\log p}, ks^{3/2})$. We call this property *k-stability* (Definition 6.1). Moreover, the support of $\hat{\theta}(\mathcal{D})$ equals the support of underlying parameter vector θ^* (see (6.1)) and $\|\hat{\theta}(\mathcal{D}) - \theta^*\|_2$ goes

down to zero as $n \to \infty$. It is important to note that Assumption 6.8 in particular is satisfied by a random Gaussian design matrix X and a sub-gaussian noise vector w. We will discuss the stochastic setting for stability in Section 6.3.6.

The main stability theorem for LASSO is given in Theorem 6.11. For the purpose of clarity, we defer the complete proof of the stability theorem to Section 6.3.5.1. The correctness follows directly from Theorem 6.9. It is important to note that our stability theorem bypasses the impossibility result of [Xu et al., 2008]. In their work, [Xu et al., 2008] showed that under worst case assumptions, minimizer of the LASSO program (i.e., $\hat{\theta}(\mathcal{D})$ in (6.3)) does not have a stable support, i.e., the support changes with changing one entry in $\mathcal{D}$. Since, we work with stronger assumptions, the impossibility result does not apply to us.

Theorem 6.11 (Stability of unmodified LASSO)**.** *Fix $k \geq 1$. Suppose $s \leq \sqrt{\frac{\sigma n^{1/2} \log^{1/2} p}{2k(1/\Psi+1)}}$ and $\Lambda = 4\sigma\sqrt{n \log p}$. If there exists a θ^* such that (X, y, θ^*) is (s, Ψ, σ, Φ)-*TYPICAL *with $\Phi = \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s \log p}{n}}, \frac{8ks^{3/2}}{\Psi n}\right\}$ (for the data set $\mathcal{D} = (y, X)$ from $\mathcal{T}^*$), then $\hat{\theta}(\mathcal{D})$ has k-stable support.*

Proof sketch For any data set $\mathcal{D}'$ differing in at most k entries from $\mathcal{D}$, we construct a vector v which has the same support as $\hat{\theta}(\mathcal{D})$ and then argue that $v = \hat{\theta}(\mathcal{D}')$, i.e., v is indeed the true minimizer of the LASSO program on $\mathcal{D}'$. The novelty in the proof goes in constructing the vector v.

Let $\hat{\Gamma}$ be the support of $\hat{\theta}(\mathcal{D})$. We obtain the vector v by minimizing the objective function $\hat{\mathcal{L}}(\theta; \mathcal{D}') + \Lambda\|\theta\|_1$ restricted to the convex set $\mathcal{C}_{\hat{\Gamma}}$. Recall that all the vectors in $\mathcal{C}_{\hat{\Gamma}}$ have support in $\hat{\Gamma}$. Using the consistency result from Theorem 6.9 and a claim that shows that the L_2 distance between $\hat{\theta}(\mathcal{D})$ and v is small, we conclude that the support of v equals $\hat{\Gamma}$. By showing that under the assumptions of the theorem, the objective function at v has a zero sub-gradient, we conclude that $v = \hat{\theta}(\mathcal{D}')$.

We should mention here that a similar line of argument was used in the proof of Theorem 6.9 by [Wainwright, 2006] to argue consistency of LASSO estimators. Here we use it to argue stability of the support.

6.3.3.2 Stability of huberized LASSO

In this section, we modify the LASSO program of (6.3) in the following (6.5) to have better stability properties when $s = \Omega(\log n)$. The main idea is to huberize the loss function in order to control the gradient of the loss. Before providing the exact details of the huberization, we provide a toy example below to make the presentation clear.

Consider a simple quadratic function $f(x) = \frac{1}{2}x^2$ and a maximum gradient constraint of $\alpha \in \mathbb{R}$. One way to modify the function such that it satisfies the gradient constraint is by replacing $f(x)$ with the following.

$$\hat{f}(x) = \begin{cases} \alpha x - \frac{\alpha^2}{2} & \text{if } x > \alpha \\ \alpha x - \frac{\alpha^2}{2} & \text{if } x < -\alpha \\ \frac{1}{2}x^2 & \text{otherwise} \end{cases}$$

The two main properties of $\hat{f}$ are: i) it is continuously differentiable and ii) its gradient is always bounded by α. We will perform a similar transformation to the loss function for linear regression to control its gradient.

Recall that the loss function for linear regression is given by $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{2n}\sum_{i=1}^{n}(y_i - \langle x_i, \theta\rangle)^2$, where y_i is the i-th entry of the vector y and x_i is the i-th row of the design matrix X. We denote the function $(y_i - \langle x_i, \theta\rangle)^2$ by $\ell(\theta; y_i, x_i)$. Consider the following huberization of the loss function ℓ. For any given $y \in \mathbb{R}$ and $x \in \mathbb{R}^p$, $\hat{\ell}(\theta; y, x)$ is defined as follows. (Here s denotes the number of non-zero entries in the underlying parameter vector θ^* in the linear system defined in (6.1).)

$$\hat{\ell}(\theta; y, x) = \begin{cases} 5\sqrt{s\log n}(y - \langle x, \theta\rangle) - 12.5s\log n & \text{if } (y - \langle x, \theta\rangle) > 5\sqrt{s\log n} \\ -5\sqrt{s\log n}(y - \langle x, \theta\rangle) - 12.5s\log n & \text{if } (y - \langle x, \theta\rangle) < -5\sqrt{s\log n} \\ \frac{1}{2}(y - \langle x, \theta\rangle)^2 & \text{otherwise} \end{cases}$$

$$\tilde{\theta}(\mathcal{D}) = \arg\min_{\theta \in \mathcal{C}} \frac{1}{n}\sum_{i=1}^{n} \hat{\ell}(\theta; y_i, x_i) + \frac{\Lambda}{n}\|\theta\|_1 \tag{6.5}$$

In this section we show the correctness (Theorem 6.12) and stability property (Theorem 6.13) of $\tilde{\theta}(\mathcal{D})$ under Assumption TYPICAL (Assumption 6.8).

Theorem 6.12 (Correctness of huberized LASSO)**.** *Let $\Lambda = 4\sigma\sqrt{n\log p}$, let $\mathcal{D} = (y, X)$ be a data set from $\mathcal{T}^*$ and $n = \omega(s\log p)$. If there exists a θ^* such that for each row x_i in the design matrix X, $|\langle x_i, \theta^*\rangle| \le 2\sqrt{s\log n}$, (y, X, θ^*) is (s, Ψ, σ, Φ)-TYPICAL with $\Phi = \frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}$, then the support of $\tilde{\theta}(\mathcal{D})$ matches the support of θ^* and moreover $\|\tilde{\theta}(\mathcal{D}) - \theta^*\|_\infty \le \frac{8\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}$.*

We defer the proof of this theorem till Section 6.3.5.2. In the proof of Theorem 6.12 we show that under the assumptions of the theorem, the region where the unconstrained minimizer of the huberized LASSO estimator lies, the huberized loss function and the unmodified loss functions are the same. In Theorem 6.13 we show that as long as the data set size $n = \omega(s\log p, \frac{s^3k^2\log n}{\log p}, ks\sqrt{\log n})$, the support of $\tilde{\theta}(\mathcal{D})$ does not change even if a constant number (k) of data entries from U are removed or added in $\mathcal{D}$. The proof structure of Theorem 6.13 is same as the proof structure of Theorem 6.11 for the unmodified LASSO. For the purpose of brevity we defer the proof of Theorem 6.13 till Section 6.3.5.2.

Theorem 6.13 (Stability of huberized LASSO)**.** *Fix $k > 1$. Under assumptions of Theorem 6.12 and $n = \omega(s\log p, \frac{s^3k^2\log n}{\log p})$, (y, X, θ^*) is (s, Ψ, σ, Φ)-TYPICAL with $\Phi = \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}, \frac{20ks\sqrt{\log n}}{\Psi n}\right\}$, then $\tilde{\theta}(\mathcal{D})$ has a k-stable support.*

6.3.4 Efficient Test for k-stability

In Section 6.3.3 we saw that under Assumption 6.8 and under proper asymptotic setting of the size of the data set (n) with respect to the parameters $s, \log p$ and k, both the unmodified LASSO in (6.3) and the huberized LASSO in (6.5) have k-stable support

Function	Instantiation (Parameters: s, Λ, Ψ)	Threshold (t_i)	Slack (Δ_i)
$g_1(\mathcal{D})$	negative of the $(s+1)^{\text{st}}$ largest absolute value of $n \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}); \mathcal{D})$	$-\frac{\Lambda}{2}$	$\frac{12s^2}{\Psi}$
$g_2(\mathcal{D})$	minimum eigenvalue of $X_{\hat{\Gamma}}{}^T X_{\hat{\Gamma}}$	$2\Psi n$	s
$g_3(\mathcal{D})$	minimum absolute value of the non-zero entries in $\hat{\theta}(\mathcal{D}) \times n$	$\frac{8s^{3/2}}{\Psi}$	$\frac{4s^{3/2}}{\Psi}$
$g_4(\mathcal{D})$	negative of the max. absolute value of the non-zero entries in $\hat{\theta}(\mathcal{D}) \times n$	$\frac{8s^{3/2}}{\Psi} - n$	$\frac{4s^{3/2}}{\Psi}$

Table 6.1. Instantiation of the four test functions

for their minimizers $\hat{\theta}(\mathcal{D})$ and $\tilde{\theta}(\mathcal{D})$ respectively. An interesting question that arises is *"can we efficiently test the stability of the support of the minimizer, given a LASSO instance?"* In this section we design efficiently testable proxy conditions which allow us to test for k-stability of the support of a LASSO minimizer. For the ease of exposition, we present the results in the context of unmodified LASSO instance only. The result for the huberized LASSO follows analogously.

The main idea in designing the proxy conditions is to define a set of four test functions $g_1, \cdots, g_4$ (with each $g_i : \mathcal{T}^* \to \mathbb{R}$) that have the following properties: i) For a given data set $\mathcal{D}$ from $\mathcal{T}^*$ and given set of thresholds $t_1, \cdots, t_4$, if each $g_i(\mathcal{D}) > t_i$, then adding or removing any one entry in $\mathcal{D}$ does not change the support of the minimizer $\hat{\theta}(\mathcal{D})$. In other words, the minimizer $\hat{\theta}(\mathcal{D})$ is 1-stable. ii) Let $\Delta_1, \cdots, \Delta_4$ be a set of *slack* values. If each $g_i(\mathcal{D}) > t_i + (k-1)\Delta_i$, then the support of the minimizer $\hat{\theta}(\mathcal{D})$ is k-stable. In Table 6.1 we define the test functions (in the notation of LASSO from (6.3)) and the corresponding thresholds (t_i) and the slacks (s_i). There s refers to the sparsity parameter and $(s+1)^{\text{st}}$ largest absolute value of $n \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}); \mathcal{D})$ refers the $(s+1)$-st maximum absolute value of the coordinates from the vector $n \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}); \mathcal{D}) = -X^T(y - \langle X, \hat{\theta}(\mathcal{D})\rangle)$.

Design intuition The main intuitions that govern the design on the proxy conditions in Table 6.1 are as follows. i) One needs to make sure that gradients of the loss function along the directions not in the support of the minimizer are sufficiently smaller than Λ/n, so that changing k data entries do not increase gradient beyond Λ/n, otherwise that particular coordinate will become non-zero. ii) Along the directions in the support of the minimizer, one needs to make sure that the objective function has sufficient strong convexity, so that changing k data entries do not move the minimizer along that direction too far. iii) On data sets where the minimizer has stable support, the *local sensitivity* [Nissim et al., 2007] of the proxy conditions at $\mathcal{D}$ should be small. By local sensitivity we mean the amount by which the value of a proxy condition changes when one entry is added or removed from the data set $\mathcal{D}$.

Theorem 6.14 shows that the g_i's (with their corresponding thresholds t_i and slacks Δ_i) are efficiently testable proxy conditions for the k-stability of the support of the minimizer $\hat{\theta}(\mathcal{D})$. For the purposes of brevity, we defer the proof of this theorem till Section 6.3.5.3. Next in Theorem 6.16 we show that if the data set $\mathcal{D} = (y, X)$ satisfies a slight strengthening of Assumption 6.8 (see Assumption 6.15), then for all $i \in \{1, \cdots, 4\}$, $g_i(\mathcal{D}) > t_i + (k-1)\Delta_i$. This ensures that the proxy conditions are almost as good as the *fixed data conditions* in Assumption 6.8. In Section 6.3.6 we analyze a stochastic setting where Assumption 6.15 is satisfied with high probability.

Theorem 6.14 (k-stability (proxy version))**.** *Let $\mathcal{D}$ be a data set from $\mathcal{T}^*$. If $g_i(\mathcal{D}) > t_i + (k-1)\Delta_i$ for all $i \in \{1, \cdots, 4\}$ and $\Lambda > \frac{16ks^2}{\Psi}$, then $\hat{\theta}(\mathcal{D})$ has k-stable support.*

Assumption 6.15 (Super-typical system)**.** *Data set $(X_{n\times p}, y_{n\times 1})$ and parameter vector $\theta^* \in \mathbb{R}^p$ are $(s, \Psi, \sigma, \Phi, k)$-*Strongly-TYPICAL *if there exists a $w \in \mathbb{R}^p$ such that $y = X\theta^* + w$ and*

1. *(y, X, θ^*) is (s, Ψ, σ, Φ)-*TYPICAL *.*
2. ***Restricted Strong Convexity:** The minimum eigenvalue of ${X_\Gamma}^T X_\Gamma$ is at least $\hat{\Psi}n$, where $\hat{\Psi}n = 2\Psi n + (k-1)s$.*
3. ***Bounded Noise:** For any set Γ of size s, $\|X_{\Gamma^c}^T V w\|_\infty \leq 2\sigma\sqrt{n\log p} - 12(k-1)s^2/\Psi$, where $V = \mathbb{I}_{n\times n} - X_\Gamma({X_\Gamma}^T X_\Gamma)^{-1}{X_\Gamma}^T$ is the projector on to the complement of the column space of X_Γ.*

Theorem 6.16. *Let $\mathcal{D} = (y, X)$ be a data set from $\mathcal{T}^*$ and let $\Lambda = 4\sigma\sqrt{n\log p}$. If there exists a θ^* such that (y, X, θ^*) is $(s, \Psi, \sigma, \Phi, k)$-*Strongly-TYPICAL *with $\Phi = \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}},\ \frac{16ks^{3/2}}{\Psi n}\right\}$, then $g_i > t_i + (k-1)\Delta_i$ for all $i \in \{1, \cdots, 4\}$.*

The proof of this theorem follows using an intuition very similar to that used in the proof of Theorem 6.11. For the sake of clarity, we defer the proof till Section 6.3.5.3.

6.3.5 Missing Proofs: Stability of LASSO Estimator in the Fixed Data Setting

6.3.5.1 Proof of Theorem 6.11 (Stability of Unmodified LASSO)

Proof of Theorem 6.11 follows directly from the following two lemmas and a claim (Lemmas 6.17 and 6.18 and Claim 6.19). The main idea is to show that under Assumption (s, Ψ, σ, Φ)-TYPICAL with $\Phi = \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}, \frac{8ks^{3/2}}{\Psi n}\right\}$, changing k entries in $\mathcal{D}$ does not change the support of $\hat{\theta}(\mathcal{D})$.

Lemma 6.17. *Under the assumptions of Theorem 6.11 if $\hat{\Gamma}$ is the support of $\hat{\theta}(\mathcal{D})$ and $\hat{\theta}(\mathcal{D})_{\hat{\Gamma}} = \arg\min_{\theta\in\mathcal{C}_{\hat{\Gamma}}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Lambda}{n}\|\theta\|_1$, then $\hat{\theta}(\mathcal{D})_{\hat{\Gamma}}$ equals $\hat{\theta}(\mathcal{D})$.*

For the ease of notation, we denote $\hat{\theta}(\mathcal{D})_{\hat{\Gamma}}$ by z.

Lemma 6.18. *Let $\mathcal{D}' = (y', X')$ be a data set formed by inserting (removing) k entries in the data set $\mathcal{D}$ from the domain U and let $z' = \arg\min_{\theta\in\mathcal{C}_{\hat{\Gamma}}} \frac{1}{2|\mathcal{D}'|}\|y' - X'\theta\|_2^2 + \frac{\Lambda}{|\mathcal{D}'|}\|\theta\|_1$. Under assumptions of Lemma 6.17, $z' = \hat{\theta}(\mathcal{D}')$, where $\hat{\theta}(\mathcal{D}') = \arg\min_{\theta\in\mathcal{C}} \frac{1}{2|\mathcal{D}'|}\|y' - X'\theta\|_2^2 + \frac{\Lambda}{|\mathcal{D}'|}\|\theta\|_1$.*

To prove the above lemma, we use a proof technique which was developed by [Wainwright, 2006] under the name of *primal-dual construction* and was used to argue consistency in non-private sparse linear regression.

Claim 6.19. *Under assumptions of Lemma 6.18, $\hat{\theta}(\mathcal{D})$ and $\hat{\theta}(\mathcal{D}')$ have the same support.*

In the following we provide the proofs of the above two lemmas and the claim.

Proof of Lemma 6.17. In order to prove this lemma, we first prove that the minimizer $\hat{\theta}(\mathcal{D})$ is unique. We use Theorem 6.9 (which is a modified version of Theorem 1 from [Wainwright, 2006]) to prove the above claim.

Since from Theorem 6.9 we have $\|\hat{\theta}(\mathcal{D}) - \theta^*\|_\infty \leq \Phi$, it follows that $\hat{\theta}(\mathcal{D})$ lies in the interior of the set $\mathcal{C}$. This in turn implies that the objective function $\frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Lambda}{n}\|\theta\|_1$ has a sub-gradient of zero at $\hat{\theta}(\mathcal{D})$. Additionally, notice that by assumption, the objective function restricted to the support of $\hat{\theta}(\mathcal{D})$ is strongly convex, since the support of $\hat{\theta}(\mathcal{D})$ and θ^* are same. These two observations along with the fact that the gradient of the objective function just outside $\hat{\theta}(\mathcal{D})$ is at least Λ (on the subspace orthogonal to the support of $\hat{\theta}(\mathcal{D})$) imply that the gradient of the objective function just outside $\hat{\theta}(\mathcal{D})$ is strictly greater than zero. Hence, $\hat{\theta}(\mathcal{D})$ is the unique minimizer.

By the restricted strong convexity property of the objective function, $\frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Lambda}{n}\|\theta\|_1$ has an unique minimizer $\hat{\theta}(\mathcal{D})_{\hat{\Gamma}}$ in $\mathcal{C}_{\hat{\Gamma}}$. Now, if $\hat{\theta}(\mathcal{D})_{\hat{\Gamma}}$ does not equal $\hat{\theta}(\mathcal{D})$, then it contradicts that $\hat{\theta}(\mathcal{D}) = \arg\min_{\theta \in \mathcal{C}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Lambda}{n}\|\theta\|_1$. □

Proof of Lemma 6.18. For the ease of notation, we fix the following: i) $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{n}\sum_{i=1}^n \ell(\theta; d_i)$, where $d_i = (y_i, x_i)$, y_i is the i-th entry of y and x_i is the i-th row of X, ii) we denote $\hat{\theta}(\mathcal{D})_{\hat{\Gamma}}$ by z. Also, since by Theorem 6.9, $\hat{\Gamma}$ equals the support of θ^* (i.e., Γ^*), we fix $\hat{\Gamma} = \Gamma^*$.

Let $z' = \arg\min_{\theta \in \mathcal{C}_{\Gamma^*}} \hat{\mathcal{L}}(\theta; \mathcal{D}') + \frac{\Lambda}{n+k}\|\theta\|_1$. W.l.o.g. assume that $\mathcal{D}'$ has k entries more than $\mathcal{D}$ and call these entries $\alpha_1, \cdots, \alpha_k$. (The analysis for the case when $\mathcal{D}'$ has k entries less than $\mathcal{D}$ follows analogously.) In the following claim we show that z' does not differ too much from z in the L_2-metric.

Claim 6.20. $\|z - z'\|_2 \leq \frac{4ks^{3/2}}{\Psi n}$.

Proof. By restricted strong convexity of $\hat{\mathcal{L}}$ at z in a ball (in the subspace formed by the support set Γ^*) of radius $\frac{2k\zeta}{\Psi n}$ around it, we have the following.

$$
\begin{aligned}
n\hat{\mathcal{L}}(z'; \mathcal{D}) + \Lambda\|z'\|_1 &\geq n\hat{\mathcal{L}}(z; \mathcal{D}) + \Lambda\|z\|_1 + \frac{\Psi n}{2}\|z' - z\|_2^2 \\
\Rightarrow \left((n+k)\hat{\mathcal{L}}(z'; \mathcal{D}') - \sum_{i=1}^k \ell(z'; \alpha_i)\right) + \lambda\|z'\|_1 &\geq \left((n+k)\hat{\mathcal{L}}(z; \mathcal{D}') - \sum_{i=1}^k \ell(z; \alpha_i)\right) \\
&\quad + \Lambda\|z\|_1 + \frac{\Psi n}{2}\|z' - z\|_2^2 \\
\Rightarrow \frac{\Psi n}{2}\|z - z'\|_2^2 &\leq \sum_{i=1}^k |\ell(z; \alpha_i) - \ell(z'; \alpha_i)|
\end{aligned}
$$

The last inequality follows from the fact that $\hat{\mathcal{L}}(z';\mathcal{D}') \leq \hat{\mathcal{L}}(z;\mathcal{D}')$. Now, by mean value theorem for any data entry d, $|\ell(z;d) - \ell(z';d)| \leq \| \bigtriangledown \ell(z'';d)\|_2 \|z - z'\|_2$, where z'' is some vector in $\mathcal{C}_{\Gamma^*}$. By assumption, $\| \bigtriangledown \ell(z'';d)\|_2 \leq 2s^{3/2}$.

Hence, it follows that $\|z - z'\|_2 \leq \frac{4ks^{3/2}}{\Psi n}$. □

Now using Claim 6.21 below, we conclude that z' is indeed the unique minimizer in $\mathcal{C}$ which minimizes $\hat{\mathcal{L}}(\theta;\mathcal{D}') + \frac{\Lambda}{n+k}\|\theta\|_1$.

Claim 6.21. *If $\Lambda = 4\sigma\sqrt{\log p}$, then z' is the unique minimizer of* $\arg\min_{\theta\in\mathcal{C}} \hat{\mathcal{L}}(\theta;\mathcal{D}') + \frac{\Lambda}{n+k}\|\theta\|_1$.

Proof. By assumption, $\|\theta^*\|_\infty \leq 1 - \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}, \frac{8ks^{3/2}}{\Psi n}\right\}$. Also from Theorem 6.9, we know that $\|\theta^* - \hat{\theta}(\mathcal{D})\|_\infty \leq \frac{8\sigma}{\Psi}\sqrt{\frac{\log p}{n}}$. Using the bound obtained in Claim 6.20, we conclude that z' lie in the interior of the set $\mathcal{C}$. Hence, along any direction $i \in \Gamma^*$ there exist a sub-gradient of the objective function at z' whose slope is zero. In the following we analyze the sub-gradients of the objective functions along directions $i \in [p] - \Gamma^*$.

For any direction $i \in [p] - \Gamma^*$ we have,

$$(n+k) \bigtriangledown \hat{\mathcal{L}}(z';\mathcal{D}')_i = n \bigtriangledown \hat{\mathcal{L}}(z;\mathcal{D})_i + n(\bigtriangledown\hat{\mathcal{L}}(z';\mathcal{D})_i - \bigtriangledown\hat{\mathcal{L}}(z;\mathcal{D})_i) + \sum_{j=1}^{k} \bigtriangledown\ell(z';\alpha_j)_i$$

$$\Rightarrow |(n+k) \bigtriangledown \hat{\mathcal{L}}(z';\mathcal{D}')_i| \leq \underbrace{|n \bigtriangledown \hat{\mathcal{L}}(z;\mathcal{D})_i|}_{A} + \underbrace{|n(\bigtriangledown\hat{\mathcal{L}}(z';\mathcal{D})_i - \bigtriangledown\hat{\mathcal{L}}(z;\mathcal{D}))_i|}_{B} + \underbrace{\left|\sum_{j=1}^{k} \bigtriangledown\ell(z';\alpha_j)_i\right|}_{C} \tag{6.6}$$

We will bound each of the terms (A, B and C) on the right individually in order to show that $A + B + C < \Lambda$. This will imply that z' is the minimizer of the objective function $\hat{\mathcal{L}}(\theta;\mathcal{D}') + \frac{\Lambda}{n+k}\|\theta\|_1$ when restricted to the convex set $\mathcal{C}$. The uniqueness follows from the restricted strong convexity of the objective function in the directions in Γ^*.

Bound term $A \leq \frac{\Lambda}{2}$ in (6.6): Notice that term A is equal to $|X^T(y - Xz)|_i$. We have argued in the proof of Lemma 6.17, that z lies in the interior of the convex set $\mathcal{C}$. Now since z is the minimizer of $\frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Lambda}{2n}\|\theta\|_1$, therefore

$$\frac{1}{n}\begin{bmatrix} {X_{\Gamma^*}}^T X_{\Gamma^*} & {X_{\Gamma^*}}^T X_{\Gamma^{*c}} \\ {X_{\Gamma^{*c}}}^T X_{\Gamma^*} & {X_{\Gamma^{*c}}}^T X_{\Gamma^*} \end{bmatrix} \begin{bmatrix} z_{|\Gamma^*} - \theta^*_{|\Gamma^*} \\ 0 \end{bmatrix} + \frac{1}{n}\begin{bmatrix} {X_{\Gamma^*}}^T \\ X_{\Gamma^{*c}}^T \end{bmatrix} w + \frac{\Lambda}{n}\begin{bmatrix} v_{|\Gamma^*} \\ v_{|\Gamma^{*c}} \end{bmatrix} = 0 \tag{6.7}$$

Here $\Gamma^{*c} = [p] - \Gamma^*$ and for any vector $\theta \in \mathbb{R}^p$, $\theta_{|\Gamma^*}$ is the vector formed by the coordinates of θ which are in Γ^*. Additionally, the vector v is a sub-gradient of $\|\cdot\|_1$ at z. From (6.7) we have the following.

$$({X_{\Gamma^*}}^T X_{\Gamma^*})(z_{|\Gamma^*} - {\theta^*}_{\Gamma^*}) + {X_{\Gamma^*}}^T w + \Lambda v_{|\Gamma^*} = 0 \tag{6.8}$$

$$\Leftrightarrow (z_{|\Gamma^*} - {\theta^*}_{|\Gamma^*}) = -({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} {X_{\Gamma^*}}^T w - \Lambda({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} v_{|\Gamma^*} \tag{6.9}$$

In the above expression $v_{|\Gamma^*} \in \{-1,1\}^{|\Gamma^*|}$, since for all $i \in \Gamma^*$, we have $|z_i| > 0$, where z_i is the i-th coordinate of z. Now note that $v_{|\Gamma^{*c}} \in [-1,1]^{p-|\Gamma^*|}$. Therefore, if we bound each of the coordinates of $v_{|\Gamma^{*c}}$ to be in $[-\frac{1}{2},\frac{1}{2}]$, we can conclude that for $i \in \Gamma^{*c}$, $|X^T(y - Xz)_i| \leq \frac{\Lambda}{2}$.

Combining (6.7) and (6.9), we have the following.

$$
\begin{aligned}
(X_{\Gamma^{*c}}^T X_{\Gamma^*})(z_{\Gamma^*} - \theta^*_{\Gamma^*}) + {X_{\Gamma^{*c}}}^T w + \Lambda v_{\Gamma^{*c}} &= 0 \\
\Leftrightarrow v_{\Gamma^{*c}} &= \frac{1}{\Lambda}\left(({X_{\Gamma^{*c}}}^T X_{\Gamma^*})({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} {X_{\Gamma^*}}^T w - X_{\Gamma^{*c}}^T w \right. \\
&\quad \left. - \Lambda ({X_{\Gamma^{*c}}}^T X_{\Gamma^*})({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} v_{\Gamma^*} \right) \\
&= -({X_{\Gamma^{*c}}}^T X_{\Gamma^*})({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} v_{\Gamma^*} \\
&\quad - \frac{{X_{\Gamma^{*c}}}^T}{\Lambda}\left(\mathbb{I}_{n\times n} - X_{\Gamma^*}({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} X_{\Gamma^*}^T \right) w \\
\Leftrightarrow \|v_{\Gamma^{*c}}\|_\infty &\leq \|({X_{\Gamma^{*c}}}^T X_{\Gamma^*})({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} v_{\Gamma^*}\|_\infty \\
&\quad + \frac{1}{\Lambda}\|{X_{\Gamma^{*c}}}^T \left(\mathbb{I}_{n\times n} - X_{\Gamma^*}({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} X_{\Gamma^*}^T \right) w\|_\infty \\
&= \|({X_{\Gamma^{*c}}}^T X_{\Gamma^*})({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} v_{\Gamma^*}\|_\infty + \\
&\quad + \frac{1}{\Lambda}\|{X_{\Gamma^{*c}}}^T V w\|_\infty \qquad (6.10)
\end{aligned}
$$

In the above expression $V = \left(\mathbb{I}_{n\times n} - X_{\Gamma^*}({X_{\Gamma^*}}^T X_{\Gamma^*})^{-1} X_{\Gamma^*}^T \right)$ is a projection matrix. Applying the bounds from Bullets 3 and 5 from Assumption TYPICAL (Assumption 6.8), we have $\|v_{\Gamma^c}\|_\infty < \frac{1}{2}$. From this it directly follows that for all $i \in \Gamma^{*c}$, $|X^T(y - Xz)_i| < \frac{\Lambda}{2}$.

Bound on term $B \leq \frac{4ks^2}{\Psi}$ in (6.6): The term B is upper bounded by $\|X^T X(z' - z)\|_\infty$. Since by assumption on the domain of data entries U every column of X has L_2-norm of at most $\sqrt{n}$, it follows that every entry of the matrix $X^T X$ is at most n. Also note that $(z - z')$ has only s-non-zero entries. Therefore, $\|X^T X(z' - z)\|_\infty \leq n\sqrt{s}\|z - z'\|_2$. From Claim 6.20 we already know that $\|z - z'\|_2 \leq \frac{4ks^{3/2}}{\Psi n}$. With this we get the relevant bound on B.

Bound on term $C \leq 2ks^{3/2}$ in (6.6): By the definition of $\ell(z;\alpha_j)$ (where $\alpha_j = (y,x)$ is as defined in (6.6)), we have $\bigtriangledown \ell(z;\alpha_j) = -x(y - \langle x, z\rangle)$. Using the assumed bounds on y and $\|x\|_2$, we bound $|\bigtriangledown \ell(z;\alpha_j)_i|$ by $2s^{3/2}$. Now, it directly follows that the term C is bounded by $2ks^{3/2}$.

Now to complete the proof of Claim 6.21, we show that $A + B + C < \Lambda$. From the bounds on A, B and C above, we have $A + B + C \leq \frac{\Lambda}{2} + \frac{4ks^2}{\Psi} + 2ks^{3/2}$. Recall, that $\Lambda = 4\sigma\sqrt{n \log p}$. By assumption on s, it now follows that $A + B + C < \Lambda$. □

This concludes the proof of Lemma 6.18. □

To complete the proof of Theorem 6.11 (utility guarantee), all that is left is to prove Claim 6.19.

Proof of Claim 6.19. We need to show that the supports of $\hat{\theta}(\mathcal{D})$ and $\hat{\theta}(\mathcal{D}')$ are the same. From Lemma 6.18 it directly follows that $\text{supp}(\hat{\theta}(\mathcal{D}')) \subseteq \text{supp}(\hat{\theta}(\mathcal{D}))$. To prove equality, we provide the following argument.

From Theorem 6.9 we know that $\|\hat{\theta}(\mathcal{D}) - \theta^*\|_\infty \leq \frac{8\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}$. Additionally, by assumption the absolute value of the minimum non-zero entry of θ^* is at least $\Phi = \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}, \frac{8ks^{3/2}}{\Psi n}\right\}$. This means that the absolute value of the minimum non-zero entry of $\hat{\theta}(\mathcal{D})$ is at least $\frac{4ks^{3/2}}{\Psi n}$. Recall that in Claim 6.20 we showed $\|\hat{\theta}(\mathcal{D}) - \hat{\theta}(\mathcal{D}')\|_\infty \leq \frac{4ks^{3/2}}{\Psi n}$. From this we can conclude that every coordinate where $\hat{\theta}(\mathcal{D})$ is non-zero, $\hat{\theta}(\mathcal{D}')$ is also non-zero.

Hence, $\text{supp}(\hat{\theta}(\mathcal{D}')) = \text{supp}(\hat{\theta}(\mathcal{D}))$. This concludes the proof. □

6.3.5.2 Proofs of Theorems 6.12 (Correctness Theorem) and 6.13 (Stability Theorem) for Huberized LASSO

Proof of Theorem 6.12 (Correctness Theorem). We first show that the support of $\tilde{\theta}(\mathcal{D})$ in (6.5) will be the same as the output of LASSO in (6.3), i.e., the support of $\hat{\theta}(\mathcal{D})$ in (6.3) is same as the support of $\tilde{\theta}(\mathcal{D})$. Moreover, we show that the minimizer $\tilde{\theta}(\mathcal{D})$ equals $\hat{\theta}(\mathcal{D})$.

Claim 6.22. *$\tilde{\theta}(\mathcal{D})$ equals $\hat{\theta}(\mathcal{D})$.*

Proof. In order to prove this claim, we invoke Theorem 1 from [Wainwright, 2006] (see Theorem 6.9). Notice for all the rows x_i of X, by assumption $|\langle x_i, \theta^*\rangle| \leq 2\sqrt{s\log n}$. By triangle inequality we have

$$\begin{aligned}|\langle x_i, \hat{\theta}(\mathcal{D})\rangle| &\leq |\langle x_i, \theta^*\rangle| + |\langle x_i, \hat{\theta}(\mathcal{D}) - \theta^*\rangle| \\ &\leq 2\sqrt{s\log n} + \sqrt{\frac{s^2\log p}{n}}\end{aligned}$$

The last inequality follows from the bound $\|\hat{\theta}(\mathcal{D}) - \theta^*\|_2$ (see Theorem 6.9). Since, we assumed $n = \omega(s\log p)$, it follows that for all the rows x_i (with $i \in [n]$), $|\langle x_i, \hat{\theta}(\mathcal{D})\rangle| \leq 3\sqrt{s\log n}$. Therefore the following are true for all $i \in [n]$: $-x_i(y_i - \langle x_i, \hat{\theta}(\mathcal{D})\rangle) = \bigtriangledown\hat{\ell}(\hat{\theta}(\mathcal{D}); y_i, x_i)$. This property straight away implies that $\hat{\theta}(\mathcal{D})$ is the minimizer of the objective function in (6.5). To show that $\tilde{\theta}(\mathcal{D}) = \hat{\theta}(\mathcal{D})$, now all we need to show is that $\hat{\theta}(\mathcal{D})$ is the *unique* minimizer of the objective function in (6.5). This is true because at $\hat{\theta}(\mathcal{D})$ in a ball of radius $r \to 0$, the function $\hat{\ell}(\theta; y_i, x_i)$ equals the function $\frac{1}{2}(y_i - \langle x_i, \theta\rangle)^2$ for all $i \in [n]$. Hence, from the proof Lemma 6.17 since $\hat{\theta}(\mathcal{D})$ is the unique minimizer of (6.3), it follows that $\tilde{\theta}(\mathcal{D}) = \hat{\theta}(\mathcal{D})$. □

To conclude the proof of Theorem 6.12, we invoke Theorem 1 from [Wainwright, 2006]. For completeness purposes we provide it in Theorem 6.9. □

Proof of Theorem 6.13 (Stability Theorem). Since, in huberized LASSO we intend to get a better dependence on the data set size n, we weaken the constraint on the maximum and minimum allowable values of θ^*. We assume that any non-zero entry of θ^* have

absolute value between $(\Phi, 1-\Phi)$, where $\Phi = \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}, \frac{20ks\sqrt{\log n}}{\Psi n}\right\}$. Similar to the stability proof for LASSO (Theorem 6.11), we prove the stability guarantee via Lemma 6.23 and 6.24, and Claim 6.25.

Lemma 6.23. *Under assumptions of Theorem 6.12, if $\hat{\Gamma}$ is the support of $\tilde{\theta}(\mathcal{D})$ and $\tilde{\theta}(\mathcal{D})_{\hat{\Gamma}} = \arg\min_{\theta\in\mathcal{C}_{\hat{\Gamma}}} \frac{1}{n}\sum_{i=1}^{n}\hat{\ell}(\theta;(y_i,X_i)) + \frac{\Lambda}{n}\|\theta\|_1$, then $\tilde{\theta}(\mathcal{D})_{\hat{\Gamma}}$ equals $\tilde{\theta}(\mathcal{D})$.*

For the ease of notation, we denote $\tilde{\theta}(\mathcal{D})_{\hat{\Gamma}}$ by z.

Lemma 6.24. *Let $\mathcal{D}' = (y', X')$ be a data set formed by inserting (removing) k entries in $\mathcal{D}$ (which are from the domain U) and let $z' = \arg\min_{\theta\in\mathcal{C}_{\hat{\Gamma}}} \frac{1}{|\mathcal{D}'|}\sum_{i=1}^{|\mathcal{D}'|}\hat{\ell}(\theta;(y'_i,X'_i)) + \frac{\Lambda}{|\mathcal{D}'|}\|\theta\|_1$. Under assumptions of Lemma 6.23, we have $z' = \tilde{\theta}(\mathcal{D}')$, where $\tilde{\theta}(\mathcal{D}') = \arg\min_{\theta\in\mathcal{C}} \frac{1}{|\mathcal{D}'|}\sum_{i=1}^{|\mathcal{D}'|}\hat{\ell}(\theta;(y'_i,X'_i)) + \frac{\Lambda}{|\mathcal{D}'|}\|\theta\|_1$.*

To prove the above lemma, we use a proof technique which was developed by [Wainwright, 2006] under the name of *primal-dual construction* and was used to argue consistency in non-private sparse linear regression.

Claim 6.25. *Under assumptions of Lemma 6.24, $\tilde{\theta}(\mathcal{D})$ and $\tilde{\theta}(\mathcal{D}')$ have the same support.*

In the following we provide the proofs of the above two lemmas and the claim. The proof of Lemma 6.23 is exactly the same for Lemma 6.17 in Section 6.3.5.1 and hence omitted here.

Proof of Lemma 6.24. For the ease of notation, we fix the following: i) $\hat{\mathcal{L}}(\theta;\mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n}\hat{\ell}(\theta;d_i)$, where $d_i = (y_i, x_i)$, y_i is the i-th entry of y and x_i is the i-th row of X, ii) we denote $\tilde{\theta}(\mathcal{D})_{\hat{\Gamma}}$ by z. Also, since by Theorem 6.12, $\hat{\Gamma}$ equals the support of θ^* (i.e., Γ^*), we fix $\hat{\Gamma} = \Gamma^*$.

Let $z' = \arg\min_{\theta\in\mathcal{C}_{\Gamma^*}} \hat{\mathcal{L}}(\theta;\mathcal{D}') + \frac{\Lambda}{n+k}\|\theta\|_1$. W.l.o.g. assume that $\mathcal{D}'$ has k entries more than $\mathcal{D}$ and call these entries $\alpha_1, \cdots, \alpha_k$. (The analysis for the case when $\mathcal{D}'$ has k entries less than $\mathcal{D}$ follows analogously.) In the following claim we show that z' does not differ too much from z in the L_2-metric.

Claim 6.26. $\|z - z'\|_2 \leq \frac{10ks\sqrt{\log n}}{\Psi n}$.

Proof. By restricted strong convexity of $\hat{\mathcal{L}}$ at z in a ball (in the subspace formed by the support set Γ^*) of radius $\frac{2k\zeta}{\Psi n}$ around it, we have the following.

$$
\begin{aligned}
n\hat{\mathcal{L}}(z';\mathcal{D}) + \Lambda\|z'\|_1 &\geq n\hat{\mathcal{L}}(z;\mathcal{D}) + \Lambda\|z\|_1 + \frac{\Psi n}{2}\|z'-z\|_2^2 \\
\Rightarrow \left((n+k)\hat{\mathcal{L}}(z';\mathcal{D}') - \sum_{i=1}^{k}\ell(z';\alpha_i)\right) + \lambda\|z'\|_1 &\geq \left((n+k)\hat{\mathcal{L}}(z;\mathcal{D}') - \sum_{i=1}^{k}\ell(z;\alpha_i)\right) \\
&\quad + \Lambda\|z\|_1 + \frac{\Psi n}{2}\|z'-z\|_2^2
\end{aligned}
$$

$$\Rightarrow \frac{\Psi n}{2}\|z - z'\|_2^2 \;\leq\; \sum_{i=1}^{k} |\ell(z;\alpha_i) - \ell(z';\alpha_i)|$$

The last inequality follows from the fact that $\hat{\mathcal{L}}(z';\mathcal{D}') \leq \hat{\mathcal{L}}(z;\mathcal{D}')$. Now, by mean value theorem for any data entry d, $|\ell(z;d) - \ell(z';d)| \leq \| \bigtriangledown \ell(z'';d)\|_2 \|z - z'\|_2$, where z'' is some vector in $\mathcal{C}_{\Gamma^*}$. Therefore, $\| \bigtriangledown \ell(z'';d)\|_2 \leq 2s\sqrt{\log n}$.

Hence, it follows that $\|z - z'\|_2 \leq \frac{10ks\sqrt{\log n}}{\Psi n}$. □

Now using Claim 6.27 below, we conclude that z' is indeed the unique minimizer in $\mathcal{C}$ which minimizes $\hat{\mathcal{L}}(\theta;\mathcal{D}') + \frac{\Lambda}{n+k}\|\theta\|_1$.

Claim 6.27. *z' is the unique minimizer of* $\arg\min_{\theta\in\mathcal{C}} \hat{\mathcal{L}}(\theta;\mathcal{D}') + \frac{\Lambda}{n+k}\|\theta\|_1$.

Proof. By assumption, $\|\theta^*\|_\infty \leq 1 - \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}, \frac{20ks\sqrt{\log n}}{\Psi n}\right\}$. Also from Theorem 6.9, we know that $\|\theta^* - \tilde{\theta}(\mathcal{D})\|_\infty \leq \frac{8\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}$. Using the bound obtained in Claim 6.26, we conclude that z' lie in the interior of the set $\mathcal{C}$. Hence, along any direction $i \in \Gamma^*$ there exist a sub-gradient of the objective function at z' whose slope is zero. In the following we analyze the sub-gradients of the objective functions along directions $i \in [p] - \Gamma^*$.

For any direction $i \in [p] - \Gamma^*$ we have,

$$(n+k) \bigtriangledown \hat{\mathcal{L}}(z';\mathcal{D}')_i = n \bigtriangledown \hat{\mathcal{L}}(z;\mathcal{D})_i + n(\bigtriangledown\hat{\mathcal{L}}(z';\mathcal{D})_i - \bigtriangledown\hat{\mathcal{L}}(z;\mathcal{D})_i) + \sum_{j=1}^{k} \bigtriangledown\ell(z';\alpha_j)_i$$

$$\Rightarrow |(n+k) \bigtriangledown \hat{\mathcal{L}}(z';\mathcal{D}')_i| \leq \underbrace{|n \bigtriangledown \hat{\mathcal{L}}(z;\mathcal{D})_i|}_{A} + \underbrace{|n(\bigtriangledown\hat{\mathcal{L}}(z';\mathcal{D})_i - \bigtriangledown\hat{\mathcal{L}}(z;\mathcal{D}))_i|}_{B} + \underbrace{\left|\sum_{j=1}^{k} \bigtriangledown\ell(z';\alpha_j)_i\right|}_{C} \tag{6.11}$$

We will bound each of the terms (A, B and C) on the right individually in order to show that $A + B + C < \Lambda$. This will imply that z' is the minimizer of the objective function $\hat{\mathcal{L}}(\theta;\mathcal{D}') + \frac{\Lambda}{n+k}\|\theta\|_1$ when restricted to the convex set $\mathcal{C}$. The uniqueness follows from the restricted strong convexity of the objective function in the directions in Γ^*.

Bound term $A \leq \frac{\Lambda}{2}$ in (6.11): Notice that term A is equal to $|X^T(y - Xz)|_i$. We have argued in the proof of Lemma 6.17, that z lies in the interior of the convex set $\mathcal{C}$. Now since z is the minimizer of $\frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Lambda}{2n}\|\theta\|_1$, therefore

$$\frac{1}{n}\begin{bmatrix} {X_{\Gamma^*}}^T X_{\Gamma^*} & {X_{\Gamma^*}}^T X_{\Gamma^{*c}} \\ {X_{\Gamma^{*c}}}^T X_{\Gamma^*} & {X_{\Gamma^{*c}}}^T X_{\Gamma^*} \end{bmatrix} \begin{bmatrix} z_{|\Gamma^*} - \theta^*_{|\Gamma^*} \\ 0 \end{bmatrix} + \frac{1}{n}\begin{bmatrix} {X_{\Gamma^*}}^T \\ X_{\Gamma^{*c}}^T \end{bmatrix} w + \frac{\Lambda}{n}\begin{bmatrix} v_{|\Gamma^*} \\ v_{|\Gamma^{*c}} \end{bmatrix} = 0 \tag{6.12}$$

Here $\Gamma^{*c} = [p] - \Gamma^*$ and for any vector $\theta \in \mathbb{R}^p$, $\theta_{|\Gamma^*}$ is the vector formed by the coordinates of θ which are in Γ^*. Additionally, the vector v is a sub-gradient of $\|\cdot\|_1$ at z. From

(6.12) we have the following.

$$(X_{\Gamma^*}{}^T X_{\Gamma^*})(z_{|\Gamma^*} - \theta^*{}_{\Gamma^*}) + X_{\Gamma^*}{}^T w + \Lambda v_{|\Gamma^*} = 0 \tag{6.13}$$

$$\Leftrightarrow (z_{|\Gamma^*} - \theta^*{}_{|\Gamma^*}) = -(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} X_{\Gamma^*}{}^T w - \Lambda (X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} v_{|\Gamma^*} \tag{6.14}$$

In the above expression $v_{|\Gamma^*} \in \{-1, 1\}^{|\Gamma^*|}$, since for all $i \in \Gamma^*$, we have $|z_i| > 0$, where z_i is the i-th coordinate of z. Now note that $v_{|\Gamma^{*c}} \in [-1, 1]^{p-|\Gamma^*|}$. Therefore, if we bound each of the coordinates of $v_{|\Gamma^{*c}}$ to be in $[-\frac{1}{2}, \frac{1}{2}]$, we can conclude that for $i \in \Gamma^{*c}$, $|X^T(y - Xz)_i| \leq \frac{\Lambda}{2}$.

Combining Equations 6.12 and 6.14, we have the following.

$$\begin{aligned}
(X^T_{\Gamma^{*c}} X_{\Gamma^*})(z_{\Gamma^*} - \theta^*_{\Gamma^*}) + X_{\Gamma^{*c}}{}^T w + \Lambda v_{\Gamma^{*c}} &= 0 \\
\Leftrightarrow v_{\Gamma^{*c}} &= \frac{1}{\Lambda}\left((X_{\Gamma^{*c}}{}^T X_{\Gamma^*})(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} X_{\Gamma^*}{}^T w - X^T_{\Gamma^{*c}} w\right. \\
&\quad \left. - \Lambda (X_{\Gamma^{*c}}{}^T X_{\Gamma^*})(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} v_{\Gamma^*}\right) \\
&= -(X_{\Gamma^{*c}}{}^T X_{\Gamma^*})(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} v_{\Gamma^*} \\
&\quad - \frac{X_{\Gamma^{*c}}{}^T}{\Lambda}\left(\mathbb{I}_{n\times n} - X_{\Gamma^*}(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} X^T_{\Gamma^*}\right) w \\
\Leftrightarrow \|v_{\Gamma^{*c}}\|_\infty &\leq \|(X_{\Gamma^{*c}}{}^T X_{\Gamma^*})(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} v_{\Gamma^*}\|_\infty \\
&\quad + \frac{1}{\Lambda}\|X_{\Gamma^{*c}}{}^T \left(\mathbb{I}_{n\times n} - X_{\Gamma^*}(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} X^T_{\Gamma^*}\right) w\|_\infty \\
&= \|(X_{\Gamma^{*c}}{}^T X_{\Gamma^*})(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} v_{\Gamma^*}\|_\infty + \\
&\quad + \frac{1}{\Lambda}\|X_{\Gamma^{*c}}{}^T V w\|_\infty
\end{aligned}$$

In the above expression $V = \left(\mathbb{I}_{n\times n} - X_{\Gamma^*}(X_{\Gamma^*}{}^T X_{\Gamma^*})^{-1} X^T_{\Gamma^*}\right)$ is a projection matrix. Applying the bounds from Bullets 3 and 5 from Assumption TYPICAL (Assumption 6.8), we have $\|v_{\Gamma^c}\|_\infty < \frac{1}{2}$. From this it directly follows that for all $i \in \Gamma^{*c}$, $|X^T(y - Xz)_i| < \frac{\Lambda}{2}$.

Bound on term $B \leq \frac{10ks^{3/2}\sqrt{\log n}}{\Psi}$ in (6.11): The term B is upper bounded by $\|X^T X(z' - z)\|_\infty$. First notice that since by Assumption (s, Ψ, σ, Φ)-TYPICAL every column of X has L_2-norm of at most $\sqrt{n}$. Hence, it follows that every entry of the matrix $X^T X$ is at most n. Also note that $(z - z')$ has only s-non-zero entries. Therefore, $\|X^T X(z' - z)\|_\infty \leq n\sqrt{s}\|z - z'\|_2$. From Claim 6.26 we already know that $\|z - z'\|_2 \leq \frac{10ks\sqrt{\log n}}{\Psi n}$. With this we get the relevant bound on B.

Bound on term $C \leq 10ks\sqrt{\log n}$ in (6.11): By the definition of $\ell(z; \alpha_j)$ (where $\alpha_j = (y, x)$ is as defined in (6.11)), we have $\bigtriangledown \ell(z; \alpha_j) = -x(y - \langle x, z\rangle)$. From the assumed bounds on y and $\|x\|_2$ in Section 6.3.2, we bound $|\bigtriangledown \ell(z; \alpha_j)_i|$ by $10s^2\sqrt{\log n}$. Now, it directly follows that the term C is bounded by $10ks\sqrt{\log n}$.

Now to complete the proof of Claim 6.27, we show that $A + B + C < \Lambda$. From the bounds on A, B and C above, we have $A + B + C \leq \frac{\Lambda}{2} + \frac{10ks^{3/2}\sqrt{\log n}}{\Psi} + 10ks\sqrt{\log n}$. Recall, that $\Lambda = 4\sigma\sqrt{n \log p}$. By assumption on s, it now follows that $A + B + C < \Lambda$. □

This concludes the proof of Lemma 6.24. □

To complete the proof of Theorem 6.13 (utility guarantee), all is left is to provide the proof for Claim 6.25.

Proof of Claim 6.25. We need to show that the supports of $\tilde{\theta}(\mathcal{D})$ and $\tilde{\theta}(\mathcal{D}')$ are the same. From Lemma 6.18 it directly follows that $\text{supp}(\tilde{\theta}(\mathcal{D}')) \subseteq \text{supp}(\tilde{\theta}(\mathcal{D}))$. To prove equality, we provide the following argument.

From Theorem 6.9 we know that $\|\tilde{\theta}(\mathcal{D}) - \theta^*\|_\infty \leq \frac{8\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}$. Additionally, by assumption the absolute value of the minimum non-zero entry of θ^* is at least $\Phi = \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}, \frac{20ks\sqrt{\log n}}{\Psi n}\right\}$. This means that the absolute value of the minimum non-zero entry of $\tilde{\theta}(\mathcal{D})$ is at least $\frac{10ks\sqrt{\log n}}{\Psi n}$. Recall that in Claim 6.26 we showed $\|\tilde{\theta}(\mathcal{D}) - \tilde{\theta}(\mathcal{D}')\|_\infty \leq \frac{10ks\sqrt{\log n}}{\Psi n}$. From this we can conclude that every coordinate where $\tilde{\theta}(\mathcal{D})$ is non-zero, $\tilde{\theta}(\mathcal{D}')$ is also non-zero.

Hence, $\text{supp}(\tilde{\theta}(\mathcal{D}')) = \text{supp}(\tilde{\theta}(\mathcal{D}))$. This concludes the proof. □

□

6.3.5.3 Proofs of Theorems 6.14 (k-stability (proxy version)) and 6.16 (Strongly-TYPICAL $\Rightarrow$ k-stability (proxy version))

Proof of Theorem 6.14

Proof of Theorem 6.14 (k-stability (proxy version)). The proof of this theorem directly follows from Lemma 6.28 and Claims 6.29, 6.30, and 6.31 below. We prove these statements after stating them.

Lemma 6.28. *If $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$ and $\Lambda > \frac{16s^2}{\Psi}$, then changing one entry in $\mathcal{D}$ does not change the support of $\hat{\theta}(\mathcal{D})$.*

In the following three lemmas we bound the local sensitivity (i.e., the amount by which the value of $g_i(\mathcal{D})$ changes when an entry is added or removed from $\mathcal{D}$) of the test functions $g_1, \cdots, g_4$ on a data set $\mathcal{D}$ when $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$.

Claim 6.29. *Following the definition in Table 6.1, if $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$ and $\Lambda > \frac{16s^2}{\Psi}$, then for any neighboring dataset $\mathcal{D}'$ (i.e.,having one entry more (less) compared to $\mathcal{D}$),*

$$\left|g_1(\mathcal{D}) - g_1(\mathcal{D}')\right| \leq \frac{12s^2}{\Psi} = \Delta_1$$

Claim 6.30. *If $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$ and $\Lambda > \frac{16s^2}{\Psi}$, then for any neighboring dataset $\mathcal{D}'$ (i.e.,having one entry more (less) compared to $\mathcal{D}$),*

$$\left|g_2(\mathcal{D}) - g_2(\mathcal{D}')\right| \leq s = \Delta_2$$

Claim 6.31. *If $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$ and $\Lambda > \frac{16s^2}{\Psi}$, then for any neighboring dataset $\mathcal{D}'$ (i.e.,having one entry more (less) compared to $\mathcal{D}$),*

$$n\|\hat{\theta}(\mathcal{D}) - \hat{\theta}(\mathcal{D}')\|_\infty \leq n\|\hat{\theta}(\mathcal{D}) - \hat{\theta}(\mathcal{D}')\|_2 \leq \frac{4s^{3/2}}{\Psi} = \Delta_3 = \Delta_4$$

Proof of Lemma 6.28. We prove the lemma via the following three claims (Claims 6.32, 6.33 and 6.34).

Claim 6.32. *If $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$ and $\Lambda > \frac{16s^2}{\Psi}$, if $\hat{\Gamma}$ is the support of $\hat{\theta}(\mathcal{D})$ and $\hat{\theta}(\mathcal{D})_{\hat{\Gamma}} = \arg\min_{\theta \in \mathcal{C}_{\hat{\Gamma}}} \frac{1}{2n}\|y - X\theta\|_2^2 + \frac{\Lambda}{n}\|\theta\|_1$ (where $\mathcal{C}_{\hat{\Gamma}} \subseteq \mathcal{C}$ is the convex subset of $\mathcal{C}$ restricted to support in $\hat{\Gamma}$), then $\hat{\theta}(\mathcal{D})_{\hat{\Gamma}}$ equals $\hat{\theta}$.*

Claim 6.33. *Let $\mathcal{D}' = (y', X'))$ be a data set formed by inserting (removing) one entry in $\mathcal{D}$. Let $z' = \arg\min_{\theta \in \mathcal{C}_{\hat{\Gamma}}} \frac{1}{2|\mathcal{D}'|}\|y' - X'\theta\|_2^2 + \frac{\Lambda}{|\mathcal{D}'|}\|\theta\|_1$. Then, if $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$ and $\Lambda > \frac{16s^2}{\Psi}$, then $z' = \hat{\theta}(\mathcal{D}')$, where $\hat{\theta}(\mathcal{D}') = \arg\min_{\theta \in \mathcal{C}} \frac{1}{2|\mathcal{D}'|}\|y' - X'\theta\|_2^2 + \frac{\Lambda}{|\mathcal{D}'|}\|\theta\|_1$.*

Claim 6.34. *If $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$ and $\Lambda > \frac{16s^2}{\Psi}$, then $\hat{\theta}(\mathcal{D})$ and $\hat{\theta}(\mathcal{D}')$ have the same support.*

The proof of these claims follow directly from the proofs of Lemmas 6.17, 6.18 and Claim 6.19 respectively.

□

Proof of Claim 6.29. W.l.o.g. we assume that the dataset $\mathcal{D}'$ has one entry more than $\mathcal{D}$ (call this entry d_{new}). First note that if $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$, then $(s+1)$-th coordinate of $\hat{\theta}(\mathcal{D})$ is zero. Additionally, note that by Lemma 6.28 the support of $\hat{\theta}(\mathcal{D})$ and $\hat{\theta}(\mathcal{D}')$ is the same. We now need to bound the following.

$$(n+1) \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}'); \mathcal{D}') = n \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}); \mathcal{D}) + n(\bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}'); \mathcal{D}) - \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}); \mathcal{D})) + \bigtriangledown \ell(\hat{\theta}(\mathcal{D}'); d_{new}) \tag{6.15}$$

For any $i \in [p] - \hat{\Gamma}$ (where $\hat{\Gamma}$ is the support of $\hat{\theta}(\mathcal{D})$), by triangle inequality the following is true.

$$\begin{aligned}\left|(n+1) \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}'); \mathcal{D}')_i - n \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}); \mathcal{D})_i\right| \leq & \underbrace{n\left|(\bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}'); \mathcal{D})_i - \bigtriangledown \hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}); \mathcal{D})_i)\right|}_{B} \\ & + \underbrace{\left|\bigtriangledown \ell(\hat{\theta}(\mathcal{D}'); d_{new})_i\right|}_{C}\end{aligned} \tag{6.16}$$

We can bound each of this terms (B and C) individually.

Bound on term $B \leq \frac{4s^2}{\Psi}$ in (6.16): The term B is upper bounded by $\|X^T X(\hat{\theta}(\mathcal{D}') - \hat{\theta}(\mathcal{D}))\|_\infty$. First notice that by definition every column of X has L_2-norm of at most $\sqrt{n}$. Thus it follows that every entry of the matrix $X^T X$ is at most n. Also note that $(\hat{\theta}(\mathcal{D}') - \hat{\theta}(\mathcal{D}))$ has only s-non-zero entries. Therefore, $\|X^T X(\hat{\theta}(\mathcal{D}') - \hat{\theta}(\mathcal{D}))\|_\infty \leq n\sqrt{s}\|\hat{\theta}(\mathcal{D}') - \hat{\theta}(\mathcal{D})\|_2$. From Claim 6.20 we already know that $\|\hat{\theta}(\mathcal{D}') - \hat{\theta}(\mathcal{D})\|_2 \leq \frac{4s^{3/2}}{\Psi}$. With this we get the relevant bound.

Bound on term $C \leq 2s^{3/2}$ in (6.16): By the definition of $\ell(\hat{\theta}(\mathcal{D}); \alpha_j)$ (where $\alpha_j = (y, x)$ is as defined in (6.6)), we have $\bigtriangledown \ell(\hat{\theta}(\mathcal{D}); \alpha_j) = -x(y - \langle x, \hat{\theta}(\mathcal{D}) \rangle)$. From the assumed bounds on y and $\|x\|_2$, we bound $|\bigtriangledown \ell(\hat{\theta}(\mathcal{D}); \alpha_j)_i|$ by $2s^{3/2}$. Now, it directly follows that the term C is bounded by $2s^{3/2}$. □

Proof of Claim 6.30. From Lemma 6.28 we know that the minimizers $\hat{\theta}(\mathcal{D})$ and $\hat{\theta}(\mathcal{D}')$ share the same support. Additionally, since if $g_i(\mathcal{D}) > t_i$ for all $i \in \{1, \cdots, 4\}$, we know that the size of the support of $\hat{\theta}(\mathcal{D})$ is less than or equal to s.

Now to prove Lemma 6.30, all we need to show is that restricted to any support Φ of size s, the minimum eigenvalue of the Hessian of $\hat{\mathcal{L}}(\hat{\theta}(\mathcal{D}); \mathcal{D})$ does not change by more than s when the dataset $\mathcal{D}$ is changed to a neighboring one $\mathcal{D}'$. Since, we are only concerned with linear regression, the Hessian of the loss function $\hat{\mathcal{L}}(\cdot; \mathcal{D})$ evaluated at any point is $X^T X$, where X is the design matrix. W.l.o.g. if we assume that $\mathcal{D}'$ has one entry more than $\mathcal{D}$ (and call that entry $d_{new} = (y, x)$, where $y \in \mathbb{R}$ and $x \in \mathbb{R}^p$, then the Hessian of $\hat{\mathcal{L}}(\cdot; \mathcal{D}')$ at any point is given by $X^T X + xx^T$.

Representing the minimum eigenvalue of a matrix A as $\eta(A)$ and A_Φ as the matrix formed by columns from the set Φ, we have the following.

$$\begin{aligned} \left| g_2(\mathcal{D}) - g_2(\mathcal{D}') \right| &= \left| \eta({X_{\hat{\Gamma}}}^T X_{\hat{\Gamma}}) - \eta({X_{\hat{\Gamma}}}^T X_{\hat{\Gamma}} + x_{\hat{\Gamma}} {x_{\hat{\Gamma}}}^T) \right| \\ &\leq \text{max. eigenvalue}(x_{\hat{\Gamma}} {x_{\hat{\Gamma}}}^T) \leq s \end{aligned}$$

The first inequality follows from Weyl's inequalities. This completes the proof. □

Proof of Claim 6.31. From Lemma 6.28 we know that the unique minimizers $\hat{\theta}(\mathcal{D})$ and $\hat{\theta}(\mathcal{D}')$ share the same support.

Now, from Claim 6.20, it follows that $\|\hat{\theta}(\mathcal{D}) - \hat{\theta}(\mathcal{D}')\|_2 \leq \frac{4s^{3/2}}{\Psi n}$. This in turn implies that $\|\hat{\theta}(\mathcal{D}) - \hat{\theta}(\mathcal{D}')\|_\infty \leq \frac{4s^{3/2}}{\Psi n}$ since L_∞-norm is less than or equal to L_2-norm. □

□

Proof of Theorem 6.16

*Proof of Theorem 6.16 (*Strongly-TYPICAL *⇒ k-stability (proxy version)).* From Assumption $(s, \Psi, \sigma, \Phi, k)$-Strongly-TYPICAL, it directly follows that $g_2(\mathcal{D}) > t_2 + (k-1)\Delta_2$. To argue about $g_3(\mathcal{D})$ and $g_4(\mathcal{D})$, notice that by Theorem 6.9 it follows that the absolute value of any non-zero entry of $\hat{\theta}(\mathcal{D})$ is in $\left(\frac{2(4+(k-1))s^{3/2}}{\Psi n}, 1 - \frac{2(4+(k-1))s^{3/2}}{\Psi n} \right)$. Hence, $g_3(\mathcal{D}) > t_3 + (k-1)\Delta_3$ and $g_4(\mathcal{D}) > t_3 + (k-1)\Delta_4$. To complete the proof, all we need to argue is about $g_1(\mathcal{D})$. Using similar proof technique of Claim 6.21 (more precisely (6.10)) and the *bounded noise* condition from Assumption $(s, \Psi, \sigma, \Phi, k)$-Strongly-TYPICAL (i.e., $\|X_{\Gamma^c}^T V w\|_\infty \leq 2\sigma\sqrt{n \log p} - 6(k-1)s^2/\Psi$) it follows that $g_1(\mathcal{D}) > t_1 + (k-1)\Delta_1$. □

6.3.6 Stability of LASSO in Stochastic Setting

In Section 6.3.3 we saw two variants of LASSO (*unmodified* and *huberized*) and a set of conditions (from Theorems 6.11 and 6.13) under which we argued that if the data

set size n is sufficiently large compared to $(s, k, \log p)$, then the minimizers of the two LASSO programs ((6.3) and (6.5)) are k-stable. In Section 6.3.4 we saw a strengthened set of assumptions (in Theorem 6.16) which implies that under these assumptions, the data set $\mathcal{D}$ will pass the efficient k-stability test designed in Section 6.3.4.

In this section we will see one specific stochastic setting for the data set $\mathcal{D} = (y, X)$, where the set of conditions (in Theorems 6.11, 6.13 and 6.16) are satisfied with high probability. The specific stochastic setting we consider here is the same we considered for consistency in Section 6.3.1. Consider each row of the design matrix X is drawn i.i.d. from $\mathcal{N}(0, \frac{1}{4}\mathbb{I}_p)$ and the entries in the noise vector w is drawn i.i.d. from a mean zero sub-Gaussian distribution with variance σ^2.

Analysis of unmodified LASSO in stochastic setting In order to make sure that Theorem 6.11 is applicable in the stochastic setting, we need to ensure two things: i) the data set $\hat{\mathcal{D}} = (\hat{y}, \hat{X})$ that gets used in Theorem 6.11 is from the domain $\mathcal{T}^*$, and ii) $(\hat{y}, \hat{X}, \theta^*)$ satisfy $(s, \Psi, \sigma, \Phi, k)$-Strongly-TYPICAL . This in particular implies that $(\hat{y}, \hat{X}, \theta^*)$ satisfy (s, Ψ, σ, Φ)-TYPICAL .

Given the data set $\mathcal{D} = (y, X)$ drawn from the distribution mentioned above, we first divide each entry in the design matrix X by $\sqrt{\log(ns)}$, where s is the sparsity parameter of the parameter vector θ^*. If the absolute value of any entry in X after dividing by $\sqrt{\log(ns)}$ exceeds 1, then just round it to -1 or 1 (whichever is closer). Call this design matrix $\hat{X}$. Similarly, if the absolute value of any entry in y exceeds s, then round it to $-s$ or s whichever is closer. By union bound and the tail property of Gaussian distribution it follows that once each entry of the design matrix X is divided by $\sqrt{\log(ns)}$, with high probability (i.e., with probability at least $1 - e^{-4}$) none of the columns which are in the support of θ^* gets truncated. Conditioned on this event, with probability at least 15/16, the design matrix $\hat{X}$ satisfies *column normalization* condition and *restricted strong convexity* condition in Assumption 6.15 with parameter Ψ' (as long as $n = \omega(ks \log n)$), where $\Psi' = \Psi/\sqrt{\log(ns)}$ and Ψ is the restricted strong convexity parameter corresponding to random Gaussian design matrix. Also by similar arguments as in the proof of Theorem 6.10, it follows that as long as $n = \omega(s \log p \log n, k^2 s^4 / \log p)$, with probability at least 7/8, the *incoherence* and *bounded noise* conditions are satisfied. Thus, we have the following stochastic analogue of Theorem 6.11. We do not need to argue about the truncation of the entries in y, since the truncation can be viewed as reducing the noise w.

Corollary 6.35. *Fix $k \geq 1$. Let $\Lambda = 4\sigma\sqrt{n \log p}$ and $n = \omega(s \log p \log n, \frac{s^4 k^2 \log n}{\log p}, ks \log n)$. There exists a constant Ψ such that under the assumptions $\|\theta^*\|_0 \leq s$, and the absolute value of any non-zero entry of θ^* is in $(\Phi, 1 - \Phi)$ (for $\Phi = \max\left\{\frac{\sigma}{\Psi}\sqrt{\frac{s \log n \log p}{n}}, \frac{ks^{3/2}\sqrt{\log n}}{\Psi n}\right\}$), with probability at least 3/4, the tuple $(\hat{y}, \hat{X}, \theta^*)$ satisfy $(s, \Psi, \sigma, \Phi, k)$-*Strongly-TYPICAL *assumption.*

The above theorem implies that as long as $\frac{n}{\log n} = \omega(s \log p, \frac{s^4 k^2}{\log p}, ks^{3/2})$, with high probability the support of the minimizer $\hat{\theta}(\hat{\mathcal{D}})$ is k-stable. The analysis for huberized LASSO is analogous and is omitted for brevity.

6.4 Private Support Selection for Sparse Linear Regression

In Section 6.2 we designed a generic framework to transform a model selection function $f : \mathcal{T}^* \to \mathcal{R}$ (which takes the data set $\mathcal{D}$ and outputs a model $\Gamma \in \mathcal{R}$) to an efficient differentially private algorithm for model selection, if we have proxy functions $\hat{f}$and $\hat{d}$ for f and its distance to instability (see Definition 6.3). In this section we use this framework for support selection in sparse linear regression. Note that in the context of linear regression (with sparsity parameter s for the underlying parameter vector θ^*) one can view the space of all possible models $\mathcal{R}$ to be all the $\binom{p}{s}$ sets of coordinates from $[p]$. Once a support of size s is chosen, one can restrict the linear regression problem to the set of s-coordinates chosen and then use algorithms (*e.g.,* objective perturbation) for private linear regression from Chapter 3 to obtain a parameter vector θ^{priv} such that $\hat{\mathcal{L}}(\theta^{\text{priv}}; \mathcal{D}) - \hat{\mathcal{L}}(\theta^*; \mathcal{D})$ scales as $O\left(\frac{s^2 \log(1/\delta)}{n\epsilon}\right)$. For more details, see Section 6.5.

The main challenge in designing a private support selection algorithm is to come up with effective proxies $\hat{f}$ and $\hat{d}$ for a given support selection function f. In the following two sections we design two different sets of proxies $\hat{f}$ and $\hat{d}$. Later we compare the sample complexities of the algorithms corresponding to both. In the current discussion, we set f to be the function that returns the support of the minimizer of unmodified LASSO program in (6.3). Although the results in this section can be easily extended to the huberized LASSO program in (6.5), we do not present it for brevity.

6.4.1 Support Selection via Sampling Stability

We use the same $\hat{f}$ and $\hat{d}$ used in Lemma 6.6 and use Algorithm 6.1 for support selection. In the current context, the non-private model selection function f in Algorithm 6.1 is the function that returns the support of the minimizer of unmodified LASSO program in (6.3). If the support has cardinality greater than s, then just pick the first s coordinates. By Theorem 6.5, the output is always (ϵ, δ)-differentially private. In order to argue that Algorithm 6.1 outputs the correct support, we make the following assumption (Assumption 6.36) about the data set $\mathcal{D}$ and the parameter vector θ^*. Under this assumption, we obtain the following utility guarantee (Corollary 6.37) for the support selection algorithm as a corollary to Theorem 6.5.

Assumption 6.36. *[(s, Ψ, σ, Φ)-Sub-sampled TYPICAL] Let $\hat{\mathcal{D}}$ be a random subset of $\mathcal{D} = (y, X)$ in which each element appears independently with probability $q = \frac{\epsilon}{32 \log(1/\delta)}$. The data set $\hat{\mathcal{D}}$ and parameter vector $\theta^* \in \mathbb{R}^p$ satisfy (s, Ψ, σ, Φ)-TYPICAL with probability at least $3/4$.*

It is important to note that the above assumption is satisfied by the stochastic setting in Section 6.3.6 with high probability.

Corollary 6.37 (Utility)**.** *Let $\Lambda = 8\sigma\sqrt{nq \log p}$ where $q = \frac{\epsilon}{32 \log(1/\delta)}$. If there exists a θ^* such that the data set $\mathcal{D} = (y, X)$ and θ^* satisfy Assumption 6.36 (Assumption (s, Ψ, σ, Φ)-Sub-sampled TYPICAL) with $\Phi \geq \frac{16\sigma}{\Psi}\sqrt{\frac{s \log p}{nq}}$, then w.p. at least $1 - 3\delta$, the current instantiation of Algorithm 6.1 outputs the correct support of θ^*.*

In order to analyze the sample complexity for support selection implied by the corollary above, first note that in expectation the sub-sampled data set $\hat{\mathcal{D}}$ will be of size nq, where n is the size of the original data set $\mathcal{D}$. Therefore, by sub-sampling we have blown up the sample complexity by a factor of $1/q$ with respect to the non-private sample complexity implied by Theorem 6.9. Hence, the sample complexity for consistent support selection, implied by the above corollary, is $(s \log p)/q$.

6.4.2 Support Selection via Stability of LASSO

In Section 6.3.3.1 we analyzed the stability properties of the unmodified LASSO program in (6.3). Moreover, in Section 6.3.4 we designed an efficient test for k-stability via defining four proxy conditions $g_1, \cdots, g_4$ in Table 6.1. In this section we transform it to a differentially private algorithm for outputting the support. In the language of Section 6.2 (and using the notation of Section 6.3.4) the proxy functions $\hat{f}$ and $\hat{d}$ we define are: i) $\hat{d}(\mathcal{D}) = \max\left\{\min_i \frac{g_i(\mathcal{D})-t_i}{\Delta_i} + 1, 0\right\}$ and ii) $\hat{f}(\mathcal{D})$ equals the support of the minimizer of the LASSO program in (6.3) when the data set $\mathcal{D}$ is stable, and $\perp$ o.w.

Lemma 6.38. *If $\Lambda > \frac{16s^2}{\Psi}$ (in (6.3)), then the proxy functions $\hat{f}$ and $\hat{d}$ defined above satisfy Definition 6.3.*

Proof. In Theorem 6.14 we saw that if for all i, $g_i(\mathcal{D}) > t_i + (k-1)\Delta_i$, then the data set $\mathcal{D}$ is k-stable. This straight away implies that if $\hat{d}(\mathcal{D}) > k$, then the data set $\mathcal{D}$ is k-stable w.r.t. the support of the minimizer.

To complete the proof, we need to show that the global sensitivity of $\hat{\mathcal{D}}$ is at most one. When $\hat{d}(\mathcal{D})$ is greater than or equal to zero, changing one entry in $\mathcal{D}$ changes $\hat{d}(\mathcal{D})$ by at most one, since one can show that in such a case each g_i changes by at most Δ_i. (See Claims 6.29, 6.30, and 6.31 in Section 6.3.5.3.) Now since $\hat{d}$ cannot be negative, global sensitivity of $\hat{d}$ is at most one. □

With Lemma 6.38 in hand, the algorithm for support selection follows from Section 6.2. Add $Lap(1/\epsilon)$ noise to $\hat{d}(\mathcal{D})$ and then test if it is greater than $\log(1/\delta)/\epsilon$. If the answer is "yes", then output the exact support of the minimizer $\hat{\theta}(\mathcal{D})$. By Proposition 6.2, the above algorithm is (ϵ, δ)-differentially private. Moreover, whenever $\hat{d}(\mathcal{D})$ is greater than $2\log(1/\delta)/\epsilon$, the algorithm outputs $f(\mathcal{D})$ with probability $1-\delta$. We obtain the following corollary.

Corollary 6.39. *Let $\mathcal{D} = (y, X)$ be a data set from $\mathcal{T}^*$ and let $\Lambda = 4\sigma\sqrt{n \log p}$. If there exists a θ^* such that (y, X, θ^*) is $(s, \Psi, \sigma, \Phi, k)$-*Strongly-TYPICAL *with $\Phi = \max\left\{\frac{16\sigma}{\Psi}\sqrt{\frac{s\log p}{n}}, \ \frac{16ks^{3/2}}{\Psi n}\right\}$, where $k = 2\log(1/\delta)/\epsilon$, then the above algorithm outputs the correct support of θ^* with probability at least $1-\delta$.*

Hence, it directly follows that the sample complexity for consistent support selection is $(s \log p, k^2 s^4/\log p, ks^{3/2})$, where $k = \log(1/\delta)/\epsilon$. It is important to note that the assumption in the above corollary is satisfied by the stochastic setting in Section 6.3.6 with high probability.

Comparing to the sample complexity obtained in Section 6.4.1, we find that when the sparsity parameter s is greater than $\log^2 p$, the *sampling* based approach has better sample complexity. When s is small (i.e., $s < \frac{\log^{2/3} p}{k^{1/3}}$), the *stability of LASSO* based approach has better sample complexity.

Note on optimal sample complexity. [Wainwright, 2006] mentioned that in the stochastic setting (i.e., the setting in Section 6.3.6) any non-private algorithm for consistent recovery of the support of θ^* will have sample complexity of at least $s \log p$. (See Section D in [Wainwright, 2006] for a detailed discussion.) Comparing to the sample complexities of our private algorithms we see that our *sampling* based algorithm matches the non-private lower bound on sample complexity up to factors in ϵ and $\log(1/\delta)$. Similarly, when $s < \frac{log^{2/3} p}{k^{2/3}}$ and $k < \log^{2/3} p$, the *stability of LASSO* based algorithm matches the sample complexity lower bound (without any dependence on the privacy parameters ϵ and δ).

6.5 Meta-algorithm for Sparse Regression

In sparse regression, we try to estimate

$$\theta^{\text{sp}} \in \arg \min_{\theta \in \mathcal{C}, \|\theta\|_0 \leq s} \hat{\mathcal{L}}(\theta; \mathcal{D}) \tag{6.17}$$

where $\mathcal{C}$ is a convex set and s is the sparsity parameter and $\hat{\mathcal{L}}$ is the corresponding *loss* (*risk*) function. For specific regression problems (like sparse linear regression), under suitable assumptions on the data set $\mathcal{D}$, θ^{sp} is close to θ^* and has similar generalization error. We are typically interested in the setting where $s < n$ and $n \ll p$. In order to obtain a private estimate for θ^{sp}, we use the following two stage approach (Algorithm 6.2):

Algorithm 6.2 Meta-algorithm for sparse linear regression

1: Output $\hat{\Gamma}$, an estimate of the support for the parameter vector θ^{sp}.
2: Privately minimize the loss function $\hat{\mathcal{L}}(\theta; \mathcal{D})$ over the convex set $\mathcal{C}$ restricted to support $\hat{\Gamma}$ using Algorithm 4.1.

One way to instantiate the meta-algorithm for sparse linear regression is by instantiating stage one of the algorithm with either of the algorithms developed in Section 6.4. Another approach is to use the *exponential mechanism* of [McSherry and Talwar, 2007] to select the support in stage one. Although the exponential mechanism based approach has weaker sample complexity bounds, but it has the advantage that it can be used to produce parameter vectors which have good generalization error, even when there is no single "best" underlying parameter vector θ^*. From a practical perspective, it is significant advantage compared to the algorithms developed in Section 6.4. In this section, we provide the formal empirical risk guarantees for the *subsampling stability* based instantiation of the meta-algorithm and then provide empirical risk guarantees for the exponential mechanism based algorithm.

6.5.1 Instantiation of the Meta-Algorithm (with Sub-sampling based Support Selection)

The current instantiation of the meta-algorithm is as follows.

- Run the sub-sampling based support selection algorithm on the data set $\mathcal{D} = (y, X)$ with privacy parameters $(\epsilon/2, \delta/2)$.
- If the return value is $\perp$, then "FAIL" and exit.
- Run the objective perturbation algorithm (Algorithm 4.1) with privacy parameter $(\epsilon/2, \delta/2)$ and the loss function $\hat{\mathcal{L}}(\hat{\theta}; \mathcal{D}) = \frac{1}{2n}\|y - X\theta\|_2^2$.

From Theorems 6.5 and 4.1 it follows that the above procedure is (ϵ, δ)-differentially private. In terms of utility, we get the following utility guarantee as a direct corollary to Theorem 4.2 and Corollary 6.37.

Theorem 6.40 (Utility)**.** *Under the conditions of Corollary 6.37 and setting the L_2 regularization parameter $\Delta = \Theta(s/\epsilon)$ (in Algorithm 4.1) , with probability at least $3/4$ we have*

$$\mathbb{E}_b\left[\hat{\mathcal{L}}(\theta^{priv}; \mathcal{D}) - \hat{\mathcal{L}}(\theta^*; \mathcal{D})\right] = O\left(\frac{s^2}{n\epsilon}\left(\frac{s^2 \log(1/\delta)}{n\epsilon\Psi} + 1\right)\right).$$

Here b is the noise vector in Algoritm 4.1.

6.5.2 Instantiation of the Meta-Algorithm (with Exponential Sampling based support selection)

At a high level, the algorithm (Algorithm Exp-mech (Algorithm 6.3)) finds a support $\hat{\Gamma}$ of size s such that restricted to this support, the minimum (non-private) loss is close the empirical loss incurred by the minimizer θ^{sp} (defined in (6.17)). In order to find $\hat{\Gamma}$ privately, Algorithm Exp-mech uses the *exponential mechanism* by [McSherry and Talwar, 2007]. Broadly speaking, the exponential mechanism first defines a score (or quality) function q for all possible outputs of the algorithm in the range space. (Algorithm Exp-mech defines the score function for any support Γ of size s as $q(\Gamma; \mathcal{D}) = \min_{\theta \in \mathcal{C}_\Gamma} \sum_{i=1}^n \ell(\theta; d_i)$ and the range space as all possible supports of size s.) It then picks a support Γ of size s with probability proportional to $\exp\left(-\epsilon q(\Gamma; \mathcal{D})\alpha\right)$, where α is an upper bound on $|\ell(\theta; d)|$ for all d in domain $\mathcal{T}$ and for all $\theta \in \mathcal{C}$ restricted to a support of size at most s. It is important to realize that Algorithm Exp-mech *may not* be computationally efficient.

6.5.2.1 Privacy Analysis

Theorem 6.41. *Algorithm* Exp-mech *(Algorithm 6.3) is ϵ-differentially private.*

Proof. In order to prove the theorem, we bound the *sensitivity* of the score function $q(\Gamma; \mathcal{D})$ (i.e.,the maximum absolute change in $q(\theta; \mathcal{D})$ when one entry of $\mathcal{D}$ is modified) via the following lemma.

Algorithm 6.3 Exponential Mechanism based support selection (Exp-mech)

Require: data set: $\mathcal{D} = \{d_1, \ldots, d_n\}$, privacy parameters: (ϵ, δ), loss function: $\hat{\mathcal{L}}(\theta; \mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta; d_i)$, dimensionality of the problem: p, number of data points: n, L_2 penalization parameter: Δ, support size of θ^*: s, closed convex set: $\mathcal{C}$, α: bound on $|\ell(\theta; d)|$ restricted to any support of size s and for any $d \in \mathcal{T}^*$

1: For any s-sparse subspace Γ, let score function $q(\Gamma; \mathcal{D}) = \min_{\theta \in \mathcal{C}_\Gamma} \sum_{i=1}^{n} \ell(\theta; d_i)$, where $\mathcal{C}_\Gamma$ refers to the vectors in $\mathcal{C}$ with support in Γ. Pick a subspace $\hat{\Gamma}$ w.p. $\propto e^{-\frac{\epsilon}{2\alpha} q(\Gamma; \mathcal{D})}$.

2: **return** $\hat{\Gamma}$

Lemma 6.42. *Sensitivity of the score function* $q(\Gamma; \mathcal{D}) = \min_{\theta \in \mathcal{C}_\Gamma} \sum_{i=1}^{n} \ell(\theta; \mathcal{D})$ *is bounded by* $\alpha \geq \max_{\theta \in \mathcal{C}_\Gamma, d \in \mathcal{T}} \ell(\theta; d)$*, where* Γ *is any* s*-sparse subset and* $\mathcal{T}$ *is the domain from which the data entries are drawn.*

Proof. Let $\mathcal{D}'$ be any data set which either has one entry more (less) than $\mathcal{D}$. W.l.o.g. we assume that $\mathcal{D}'$ has one entry more as compared to $\mathcal{D}$ (i.e.,$\mathcal{D}'$ has entry d_{n+1} which $\mathcal{D}$ does not). To bound the sensitivity of q we need to bound $|q(\Gamma; \mathcal{D}') - q(\Gamma; \mathcal{D})|$ for any database pairs $\mathcal{D}$ and $\mathcal{D}'$, and any subset Γ of size at most s. The bound is as follows.

$$
\begin{aligned}
|q(\Gamma; \mathcal{D}') - q(\Gamma; \mathcal{D})| &= \left| \min_{\theta \in \mathcal{C}_\Gamma} n\hat{\mathcal{L}}(\theta; \mathcal{D}') - \min_{\theta \in \mathcal{C}_\Gamma} n\hat{\mathcal{L}}(\theta; \mathcal{D}) \right| \\
&= \left| \min_{\theta \in \mathcal{C}_\Gamma} \left(n\hat{\mathcal{L}}(\theta; \mathcal{D}) + \ell(\theta; d_{n+1}) \right) - \min_{\theta \in \mathcal{C}_\Gamma} \left(n\hat{\mathcal{L}}(\theta; \mathcal{D}) \right) \right| \\
&\leq \left| \min_{\theta \in \mathcal{C}_\Gamma} n\hat{\mathcal{L}}(\theta; \mathcal{D}) + \max_{\theta \in \mathcal{C}_\Gamma} \ell(\theta; d_{n+1}) - \min_{\theta \in \mathcal{C}_\Gamma} n\hat{\mathcal{L}}(\theta; \mathcal{D}) \right| \\
&= \max_{\Gamma, \theta \in \mathcal{C}_\Gamma, d \in \mathcal{T}} \ell(\theta; d) \leq \alpha
\end{aligned}
$$

With this the bound in the above lemma follows. □

Now for two data sets $\mathcal{D}$ and $\mathcal{D}'$, the ratio of the probabilities for picking any support of size s is as follows.

$$
\begin{aligned}
\frac{\Pr[\hat{\Gamma}(\mathcal{D}) = \Gamma]}{\Pr[\hat{\Gamma}(\mathcal{D}') = \Gamma]} &\leq \frac{e^{-\frac{\epsilon q(\Gamma; \mathcal{D})}{2\alpha}}}{e^{-\frac{\epsilon q(\Gamma; \mathcal{D}')}{2\alpha}}} \cdot \frac{\sum_\Gamma e^{-\frac{\epsilon q(\Gamma; \mathcal{D}')}{2\alpha}}}{\sum_\Gamma e^{-\frac{\epsilon q(\Gamma; \mathcal{D})}{2\alpha}}} \\
&\leq e^{\epsilon}
\end{aligned}
$$

The lower bound of $e^{-\epsilon}$ also follows symmetrically. With this the proof is complete.

□

6.5.2.2 Utility Analysis

The main step in the utility analysis of Algorithm Exp-mech is that a "good" support has high weight in the exponential sampling. Also the utility guarantee relies on the

parameter α which essentially bounds the change in the score function for any support Γ when one entry is added or removed from the data set $\mathcal{D}$. We defer the complete proof till Section 6.5.2.3.

Theorem 6.43 (Theorem 6.46, special case)**.** *Assume that $|\ell(\theta; d)| \leq \alpha$ (for all $\theta \in \mathcal{C}_\Gamma$, for all $d \in \mathcal{T}$ and for all support Γ of size at most s). We have $\mathbb{E}\left[\hat{\mathcal{L}}(\phi; \mathcal{D}) - \hat{\mathcal{L}}(\theta^{sp}; \mathcal{D})\right] = O\left(\frac{\alpha s \log p}{\epsilon n}\right)$. Here $\phi = \arg\min_{\theta \in \mathcal{C}_{\hat{\Gamma}}} \hat{\mathcal{L}}(\theta; \mathcal{D})$ and $\hat{\Gamma}$ is the output of Algorithm* Exp-mech *and θ^{sp} is as defined in* (6.17)*.*

Instantiation with Sparse Linear Regression: For linear regression, if we instantiate the first step of Algorithm 6.2 (Algorithm Meta-Alg) with the exponential sampling described above, for outputting support $\hat{\Gamma}$ while preserving $\epsilon/2$-differential privacy, and execute Algorithm Obj-Pert (Algorithm 4.1) in the second step with privacy parameters $(\epsilon/2, \delta)$, then we obtain an (ϵ, δ)-differentially private algorithm.

In order to state the utility guarantee in the context of linear regression, we need the following set of assumptions. In the following, we define what it means to be a "well-behaved" data set. We use this definition to precisely state our assumptions.

Definition 6.44 ((s, Ψ)-well behaved)**.** *A matrix $X_{n\times p} \in \mathbb{R}^{n\times p}$, $y \in \mathbb{R}^p$ are (s, Ψ)-*well behaved *if:*

1. *$\forall i, \| M_i|_s\|_2 \leq \sqrt{s}\log p$, where $M_i|_s$ is the largest s entries of the i-th row of M.*
2. *The response vector $y \in [-s\log p, s\log p]^n$.*
3. ***Restricted Strong Convexity (RSC):** Given a set of indices $\Gamma \subset [p]$, let $C(\Gamma) = \{\theta \in \mathbb{R}^p : \|\theta_{\Gamma^c}\|_1 \leq 3\|\theta_\Gamma\|_1\}$. Here θ_Γ (respectively, θ_{Γ^c}) denotes θ restricted to entries in Γ (respectively, $\Gamma^c = [p] \setminus \Gamma$). We require that for all Γ of size $|\Gamma| = s$ and for all $\theta \in C(\Gamma)$: $\|M\theta\|_2^2 \geq n\Psi\|\theta\|_2^2$.*

Remark: If w is i.i.d. sub-gaussian with mean zero and variance σ^2 and if the design matrix $X_{n\times p}$ is generated by sampling the rows i.i.d. from a Gaussian ensemble $\mathcal{N}(0, \Sigma)$, then under reasonable assumptions on n, s, p and Σ, the tuple (X, w) is (s, σ, Ψ)-well-behaved with high probability. (Roughly, a sufficient condition is that the eigenvalues of Σ lie strictly between 0 and 1 and that n grows at least as fast as $s\log p$.) See [Negahban et al., 2010] for discussion and references.

From the above assumption one can easily conclude that $|\langle x, \theta^*\rangle| \leq s\log p$, where x is any row of the design matrix X. This means, if we truncate the responses $y_1, \cdots y_n$ (in the data set $\mathcal{D}$) to have values in $[-s\log p, s\log p]$, then the utility of the algorithm will not worsen. Therefore, w.l.o.g. we assume that $y_1, \cdots, y_n$ lie in $[-s\log p, s\log p]$.

From Theorems 3.20 and 6.43, we directly obtain the utility guarantee for the current instantiation of Meta Algorithm 6.2. We defer the complete proof till Section 6.5.2.3.

Theorem 6.45 (Theorem 6.47, special case)**.** *If θ^{sp} is as defined in* (6.17) *and the data set $\mathcal{D} = (y, X)$ is (s, Ψ)-well behaved and $\Delta = \Theta\left(s\log^2 p/\epsilon\right)$, then we have*

$$\mathbb{E}\left[\hat{\mathcal{L}}(\theta^{priv}; \mathcal{D}) - \hat{\mathcal{L}}(\theta^{sp}; \mathcal{D})\right] = O\left(\frac{s^2\log^2 p}{n\epsilon}\left(\frac{s^2\log^2 p\log(1/\delta)}{n\epsilon\Psi} + s\log p\right)\right).$$

Assuming ϵ, δ and Ψ to be constants, empirical risk goes to zero as $n \to \infty$ as long as $n = \omega(s^3 \log^3 p)$. Comparing to the sub-sampling based approach we see that the sample complexity in this case is much higher. However, as mentioned earlier, the subsampling based approach only performs well if there is single best underlying model parameter. But since the exponential mechanism based algorithm does not need any such assumption, it is expected to perform well in scenarios where there are multiple model parameters that explain the data set $\mathcal{D}$ equally well.

6.5.2.3 Proofs of Utility Theorems (Theorems 6.43 and 6.45)

Proof of Theorem 6.43 In order to prove Theorem 6.43, we prove a slightly more general version stated below. Since in Theorem 6.43 we are dealing with expected error, we ignore the term γ.

Theorem 6.46. *Assume that $|\ell(\theta; d)| \leq \alpha$ (for all $\theta \in \mathcal{C}_\Gamma$, for all $d \in \mathcal{T}^*$ and for all support Γ of size s). With probability $\geq 1 - \gamma$, we have*

$$\hat{\mathcal{L}}(\phi; \mathcal{D}) - \hat{\mathcal{L}}(\theta^{sp}; \mathcal{D}) = \frac{2\alpha s \log(p/\gamma)}{\epsilon n}$$

where $\phi = \arg\min_{\theta \in \mathcal{C}_{\hat{\Gamma}}} \hat{\mathcal{L}}(\theta; \mathcal{D})$ and $\hat{\Gamma}$ is the support selected by Algorithm Exp-mech *(Algorithm 6.3).*

Proof. Let $\Gamma_{\min}$ be the support of size $\leq s$ which minimizes $\min_{\theta \in \mathcal{C}} \hat{\mathcal{L}}_\Gamma(\theta; \mathcal{D})$ w.r.t. Γ. Recall that $\hat{\Gamma}$ is the support output by exponential sampling. Based on the distribution used for exponential sampling, we have the following for any $\kappa > 0$.

$$\Pr\left[\min_{\theta \in \mathcal{C}_{\hat{\Gamma}}} \hat{\mathcal{L}}(\theta; \mathcal{D}) \geq \min_{\theta \in \mathcal{C}_{\Gamma_{\min}}} \hat{\mathcal{L}}(\theta; \mathcal{D}) + \frac{\kappa}{n}\right] \leq \binom{p}{s} \exp\left(-\frac{\epsilon\kappa}{2\alpha}\right)$$

$$\Rightarrow \Pr\left[\min_{\theta \in \mathcal{C}_{\hat{\Gamma}}} \hat{\mathcal{L}}(\theta; \mathcal{D}) \geq \hat{\mathcal{L}}(\theta^{\text{sp}}; \mathcal{D}) + \frac{\kappa}{n}\right] \leq \binom{p}{s} \exp\left(-\frac{\epsilon\kappa}{2\alpha}\right)$$

The last inequality follows from the fact that $\hat{\mathcal{L}}(\theta^{\text{sp}}; \mathcal{D}) = \min_{\theta \in \mathcal{C}} \hat{\mathcal{L}}_{\Gamma_{\min}}(\theta; \mathcal{D})$. Setting the R.H.S. $\leq \gamma$, we have $\kappa \leq \frac{2\alpha s}{\epsilon} \log \frac{p}{\gamma}$. Thus w.p. $\geq 1 - \gamma$ we have

$$\hat{\mathcal{L}}(\phi; \mathcal{D}) - \hat{\mathcal{L}}(\theta^{\text{sp}}; \mathcal{D}) \leq \frac{2\alpha s}{n\epsilon} \log \frac{p}{\gamma}$$

This completes the proof. □

Proof of Theorem 6.45 In order to prove Theorem 6.45, we prove a slightly more general version stated below. Setting $\alpha = 4s^2 \log^2$, $\zeta = 2s^{3/2} \log^2 p$, $\lambda = s \log^2 p$ and $\|\phi\|_2 \leq \sqrt{s}$, and substituting $\Delta = \Theta\left(\frac{s \log^2 p}{\epsilon}\right)$ in Theorem 6.47 we obtain the required bound for Theorem 6.45. Since in Theorem 6.45 we are dealing with expected error, we ignore the term γ.

Theorem 6.47. *Let $\hat{\mathcal{L}}(\theta;\mathcal{D})$ be Ψ-strongly convex for a given data set $\mathcal{D}$ when the support of $\theta \in \mathcal{C}$ is restricted to any set Γ of size $\leq s$. Assuming that $\|\ell(\theta;d)\|_2 \leq \zeta$, $|\ell(\theta;d)| \leq \alpha$, λ is the bound on the maximum eigenvalue of $\bigtriangledown^2\ell$ (for all $\theta \in \mathcal{C}_\Gamma$, for all $d \in \mathcal{T}^*$ and for all support Γ of size s), with probability $\geq 1-\gamma$, the following is true.*

$$\hat{\mathcal{L}}(\theta^{priv};\mathcal{D}) - \hat{\mathcal{L}}(\theta^*;\mathcal{D}) \leq \frac{16s\zeta^2(8\log\frac{2}{\delta}+2\epsilon)\log(2/\gamma)}{n\epsilon^2(\Delta+n\Psi)} + \frac{4\alpha s}{n\epsilon}\log\frac{2p}{\gamma} + \frac{\Delta}{2n}\|\phi\|_2^2$$

where $\phi = \arg\min_{\theta\in\mathcal{C}_{\hat{\Gamma}}} \hat{\mathcal{L}}(\theta;\mathcal{D})$ and $\hat{\Gamma}$ is the support chosen by Algorithm Exp-mech *(Algorithm 6.3).*

Proof. We bound $\hat{\mathcal{L}}(\theta^{\text{priv}};\mathcal{D}) - \hat{\mathcal{L}}(\theta^*;\mathcal{D})$ in two parts A and B mentioned below.

$$\hat{\mathcal{L}}(\theta^{\text{priv}};\mathcal{D}) - \hat{\mathcal{L}}(\theta^*;\mathcal{D}) = \underbrace{\hat{\mathcal{L}}(\phi;\mathcal{D}) - \hat{\mathcal{L}}(\theta^*;\mathcal{D})}_{A} + \underbrace{\hat{\mathcal{L}}(\theta^{\text{priv}};\mathcal{D}) - \hat{\mathcal{L}}(\phi;\mathcal{D})}_{B}$$

Let us first concentrate on part A. From Theorem 6.46, w.p. $\geq 1-\frac{\gamma}{2}$ we have

$$A \leq \frac{4\alpha s}{n\epsilon}\log\frac{2p}{\gamma}$$

Notice that after selecting $\hat{\Gamma}$, the problem has reduced to a s-dimensional subspace. Now invoking Theorem 3.30 restricted to the support $\hat{\Gamma}$, setting the failure probability to $\gamma/2$ and plugging $\epsilon/2$, w.p. $\geq 1-\frac{\gamma}{2}$ we have

$$B \leq \frac{16s\zeta^2(8\log\frac{2}{\delta}+2\epsilon)\log(2/\gamma)}{n\epsilon^2(\Delta+n\Psi)} + \frac{\Delta}{2n}\|\phi\|_2^2$$

Using the bounds for A and B above, Theorem 6.46 follows. □

Chapter 7

Concluding Remarks

In this dissertation we saw three different aspects of private convex empirical risk minimization (ERM). First, we saw private ERM algorithms which either transform the minimizer of the non-private problem or transform the objective function corresponding to the ERM problem. Second, we saw a private ERM algorithm that uses online learning as a tool to obtain the private minimizer via "online to batch" conversion. We also saw a generic transformation to convert any online convex programming (OCP) algorithm to its differentially private variant. Third, we saw the first differentially private algorithms for sparse regression in high-dimensions. Some of the algorithms we presented match the optimal non-private sample complexity of $s \log p$ for sparse linear regression. Along the way, we saw various stability bounds for LASSO, a popular algorithm for sparse linear regression. In the following, we list some of the problems left unanswered in this dissertation. We think that understanding these problems is necessary to explore the complete landscape of differentially private ERM (and more generally differentially private learning).

1. We noticed that the privacy guarantees of both *output* and *objective* perturbation rely on the fact that the underlying optimization problem reaches its true minimum. However, due to their iterative nature, in practice it rarely happens that convex optimization algorithms reach the true minimum. *Can we provide privacy guarantees for output or objective perturbation when the true minimum of the objective function is not reached?* One possible way to answer this question by arguing privacy in a small enough region around the true minima. The privacy guarantee will then be conditioned on the optimization algorithm reaching this region.

2. [Williams and McSherry, 2010] mentioned an iterative algorithm for empirical risk minimization which, in the experiments they conducted, performed better than the objective perturbation algorithm. However, they did not provide any formal utility guarantees for their algorithm. The question is, *can we provide a formal utility guarantee for the algorithm in [Williams and McSherry, 2010]?* One possible way to answer this question is by applying analysis similar to the analyses of private online learning algorithms.

3. We saw that for linear regression, one can have better algorithms than the standard objective perturbation algorithm. The idea there is to use a quadratic regularization which depends on the data to control the strong convexity of the objective function. However, the current privacy analysis heavily uses the properties of linear regression. More precisely, the privacy analysis uses the fact that the hessian of the loss function for linear regression is the same at all points. One question is, *can we extend the idea of data dependent regularization to other ERM problems to obtain better utility guarantees?*

4. Our results for private online learning require the individual cost functions to be strongly convex. Under the strong convexity assumption, the regret bounds for our private algorithms scale roughly as $O(\sqrt{T})$, where T is the number of iterations. [Hazan et al., 2007] showed that for strongly convex cost functions there exist non-private algorithms that have regret $O(\log T)$. The question is, *can we bridge this exponential gap between the private regret and non-private regret? Alternatively, can we show that this gap is necessary?* Showing the exponential separation is going to be tricky, since from [Jain et al., 2012] we know that for linear regression one can have $O(poly \log T)$ regret. So, any argument for showing the exponential separation has to rule out linear regression.

5. All our private online convex programming (OCP) algorithms need to keep the previous non-private iterates in the memory, in order to predict the new iterate. A question that arises is, *can we design private online algorithms with sub-linear regret that do not need to store any non-private iterate?* For the case when the cost functions are linear, one can answer this question in the affirmative via the *follow the perturbed leader algorithm* [Kalai and Vempala, 2005].

6. Our private OCP algorithms are for problems in the *full information* setting. *Can we extend them to the bandit setting?* For the bandit setting, the main challenge is to estimate a proper gradient for the individual cost functions. [Flaxman et al., 2005] mentioned an approach for addressing this challenge (via randomization). *Using the ideas existent in the bandit learning literature, can we design differentially private algorithms for bandit problems which have sublinear regret?*

7. In the high-dimensional setting, we instantiated our model selection algorithms with LASSO as the model selector for sparse linear regression. However, our *bootstrapping* based model selector is not specific to LASSO and can work with any reasonable non-private model selector. We saw that in certain settings, the generic bootstrapping based algorithm has higher sample complexity compared to the algorithms directly based on the stability of LASSO instances. The question is, *can we design algorithms for other model selection problems which have better sample complexity compared to the bootstrapping based algorithm?*

8. Our bootstrapping based model selection algorithm will only provide a reasonable estimate when there is one underlying model that performs better than all others. In typical learning problems this may not be the case and there can be multiple

competing models. In such a case, an obvious choice is to use the exponential sampling discussed in Chapter 6. But we saw that the sample complexity of exponential sampling based algorithm is much worse than the other algorithms for sparse linear regression. *Can we come up with exponential sampling based algorithms with better sample complexity for sparse regression problems?*

9. We analyzed the stability properties of LASSO and concluded that under the consistency assumptions, the support of the minimizer of the LASSO program is actually stable under addition or removal of any constant number of entries in the training data set $\mathcal{D}$. *Can we analyze stability properties of other sparse estimation algorithms like* least angle regression *[Efron et al., 2004] and various* message passing *based algorithms?* More generally, *can we analyze stability properties of various high-dimensional regression problems like* nuclear norm *regularized regression and* trace norm regularized *regression [Negahban et al., 2010]?*

10. The most important property of any learning algorithm is *generalizability*. This in turn means that the output of the algorithm cannot depend too much on any particular entry in the training data. On the other hand, for an algorithm to be differentially private, we require that the output of the algorithm should not change too much if one entry in the data set is changed. So, philosophically the notions of stability of learning algorithms and differential privacy are very similar. It will be interesting to show a formal connection between the two. *Can we formalize the relation between generalizability of learning algorithms and differential privacy?*

Bibliography

Francis R. Bach. Bolasso: model consistent lasso estimation through the bootstrap. In *ICML*, 2008.

S. Ben-David, U. Von Luxburg, and D. Pál. A sober look at clustering stability. *Learning Theory*, 2006.

Patrick Billingsley. *Probability and Measure*. John Wiley and Sons, New York, NY, USA, 1995. ISBN 9780471007104.

Avrim Blum, Katrina Ligett, and Aaron Roth. A learning theory approach to non-interactive database privacy. In *STOC*. ACM, 2008.

Olivier Bousquet and André Elisseeff. Stability and generalization. *JMLR*, 2002.

Joseph A. Calandrino, Ann Kilzer, Arvind Narayanan, Edward W. Felten, and Vitaly Shmatikov. "you might also like: " privacy risks of collaborative filtering. In *IEEE Symposium on Security and Privacy*, 2011.

K. Chaudhuri and D. Hsu. Convergence rates for differentially private statistical estimation. In *ICML*, 2012.

Kamalika Chaudhuri and Daniel Hsu. Sample complexity bounds for differentially private learning. *JMLR*, 2011.

Kamalika Chaudhuri, Claire Monteleoni, and Anand D. Sarwate. Differentially private empirical risk minimization. *JMLR*, 2011.

Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. Online passive-aggressive algorithms. *JMLR*, 2006.

Sanjoy Dasgupta and Leonard Schulman. A probabilistic analysis of em for mixtures of separated, spherical gaussians. *JMLR*, 2007.

Cynthia Dwork. Differential privacy. In *ICALP*, 2006.

Cynthia Dwork. Differential privacy: A survey of results. In *TAMC*. Springer, 2008.

Cynthia Dwork and Jing Lei. Differential privacy and robust statistics. In *STOC*, 2009.

Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, 2006a.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, 2006b.

Cynthia Dwork, Parikshit Gopalan, Huijia Lin, Toniann Pitassi, Guy Rothblum, Adam Smith, and Sergey Yekhanin. An analysis of the Chaudhuri and Monteleoni algorithm. *Innovations in Computer Science (poster)*, 2009. Available as [Dwork et al., 2012].

Cynthia Dwork, Moni Naor, Toniann Pitassi, and Guy N. Rothblum. Differential privacy under continual observation. In *STOC*, 2010a.

Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. Boosting and differential privacy. In *FOCS*, 2010b.

Cynthia Dwork, Parikshit Gopalan, Huijia Lin, Toniann Pitassi, Guy Rothblum, Adam Smith, and Sergey Yekhanin. An analysis of the Chaudhuri and Monteleoni algorithm. Technical Report NAS-TR-0156-2012, Network and Security Research Center, Pennsylvania State University, USA, February 2012.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of statistics*, 2004.

A.D. Flaxman, A.T. Kalai, and H.B. McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *SODA*, 2005.

Srivatsava Ranjit Ganta, Shiva Prasad Kasiviswanathan, and Adam Smith. Composition attacks and auxiliary information in data privacy. In *KDD*, 2008.

Anupam Gupta, Aaron Roth, and Jonathan Ullman. Iterative constructions and private data release. In *TCC*, 2012.

Moritz Hardt and Guy N. Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. In *FOCS*, 2010.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Mach. Learn.*, 69:169–192, 2007. ISSN 0885-6125.

Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V. Pearson, Dietrich A. Stephan, Stanley F. Nelson, and David W. Craig. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS Genet*, 2008.

Peter Huber. *Robust Statistics*. Wiley, 1981.

Prateek Jain, Pravesh Kothari, and Abhradeep Thakurta. Differentially private online learning. In *COLT*, 2012.

Sham Kakade and Shai Shalev-Shwartz. Mind the duality gap: Logarithmic regret algorithms for online optimization. In *NIPS*, 2008.

Sham M. Kakade and Ambuj Tewari. On the generalization ability of online strongly convex programming algorithms. In *NIPS*, 2008.

Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer System and Science*, 71, 2005.

Vishesh Karwa, Sofya Raskhodnikova, Adam Smith, and Grigory Yaroslavtsev. Private analysis of graph structure. *PVLDB*, 2011.

Daniel Kifer and Ashwin Machanavajjhala. No free lunch in data privacy. In *SIGMOD*, 2011.

Daniel Kifer, Adam Smith, and Abhradeep Thakurta. Private convex empirical risk minimization and high-dimensional regression. In *COLT*, 2012.

Aleksandra Korolova. Privacy violations using microtargeted ads: A case study. In *ICDM Workshops*, 2010.

Brian Kulis and Peter L. Bartlett. Implicit online learning. In *ICML*, 2010.

Y. Lee, S. N. MacEachern, and Y. Jung. Regularization of case-specific parameters for robustness and efficiency. Technical report, Statistics Deepartment, Ohio State University, 2011.

Ashwin Machanavajjhala, Daniel Kifer, John M. Abowd, Johannes Gehrke, and Lars Vilhuber. Privacy: Theory meets practice on the map. In *ICDE*, 2008.

Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *FOCS*, 2007.

M. Meilă. The uniqueness of a good optimum for k-means. In *ICML*, 2006.

Nicolai Meinshausen and Peter Bhlmann. High dimensional graphs and variable selection with the lasso. *Annals of Statistics*, 2006.

Arvind Narayanan and Vitaly Shmatikov. De-anonymizing social networks. In *IEEE Symp. Security and Privacy*, 2009.

Sahand Negahban, Pradeep Ravikumar, Martin J. Wainwright, and Bin Yu. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *CoRR*, abs/1010.2731, 2010.

Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. Smooth sensitivity and sampling in private data analysis. In *STOC*, 2007.

Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2000. ISBN 0387987932.

Manas Pathak, Shantanu Rane, and Bhiksha Raj. Multiparty differential privacy via aggregation of locally trained classifiers. In *NIPS*, 2010.

Tomaso Poggio, Stephen Voinea, and Lorenzo Rosasco. Online learning, stability, and stochastic gradient descent. *CoRR*, abs/1105.4701, 2011.

A. Rakhlin and A. Caponnetto. Stability of k-means clustering. In *NIPS*, 2007.

Garvesh Raskutti, Martin J. Wainwright, and Bin Yu. Minimax rates of estimation for high-dimensional linear regression over ℓ_q-balls. *IEEE Transactions on Information Theory*, 2011.

W.H. Rogers and T.J. Wagner. A finite sample distribution-free performance bound for local discrimination rules. *The Annals of Statistics*, 1978.

Stéphane Ross and J. Andrew Bagnell. Stability conditions for online learnability. *CoRR*, abs/1108.3154, 2011.

Benjamin I. P. Rubinstein, Peter L. Bartlett, Ling Huang, and Nina Taft. Learning in a large function space: Privacy-preserving mechanisms for svm learning. *CoRR*, abs/0911.5708, 2009.

S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Learnability, stability and uniform convergence. *JMLR*, 2010.

Shai Shalev-Shwartz, Ohad Shamir, Nathan Srebro, and Karthik Sridharan. Stochastic Convex Optimization. In *COLT*, 2009.

Jun Shao. Bootstrap model selection. *Journal of the American Statistical Association*, 1996.

Adam Smith. Privacy-preserving statistical estimation with optimal convergence rates. In *STOC*, 2011.

Adam Smith and Abhradeep Thakurta. Differentially private feature selection via stability arguments, and the robustness of the lasso. *Manuscript*, 2012.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 1996.

Martin J. Wainwright. Sharp thresholds for high-dimensional and noisy recovery of sparsity using ℓ_1-constrained quadratic programs, 2006.

Oliver Williams and Frank McSherry. Probabilistic inference and differential privacy. In *NIPS*, 2010.

H. Xu, C. Caramanis, and S. Mannor. Robust regression and lasso. *IEEE Transactions on Information Theory*, 2010.

Huan Xu, S. Mannor, and C. Caramanis. Sparse algorithms are not stable: A no-free-lunch theorem. In *46th Annual Allerton Conference on Communication, Control, and Computing*, 2008.

A. H. Zemanian. *Distribution theory and transform analysis: an introduction to generalized functions, with applications*. Dover Publications, Inc., New York, NY, USA, 1987.

P. Zhao and B. Yu. On model selection consistency of lasso. *JMLR*, 2007.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *ICML*, 2003.

Vita

Abhradeep Guha Thakurta

Abhradeep Guha Thakurta is a PhD candidate in the Department of Computer Science and Engineering at The Pennsylvania State University at University Park. Abhradeep joined Penn State in 2008 after completing his Bachelor of Technology in Electrical and Electronics Engineering in 2007 from National Institute of Technology Karnataka, Surathkal. Between 2007 and 2008, he worked as a Software Development Engineer in Microsoft India Development Center, Hyderabad. Abhradeep's main research interest lies in understanding the privacy properties of various large scale machine learning algorithms. He is also interested in problems that involve understanding the relationship between various formalisms of statistical data privacy. Abhradeep was selected as the *best teaching assistant* in Spring 2010 for Data Structures and Algorithms class. He also received the *Yahoo! Key Scientific Challenges award* for the year 2011 for his work on differentially private sparse regressions. His research works have been published in SIGKDD, Asiacrypt, SIGMOD, COLT, Approx/Random and TCC.