THE UNIVERSITY OF CHICAGO

ROBUSTNESS AND MODEL ADAPTIVITY IN STOCHASTIC PROGRAMMING

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF STATISTICS

BY

XIALIANG DOU

CHICAGO, ILLINOIS

JUNE 2022

To my mom, Jing Chen

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Prof. Mihai Anitescu. His deep insight, extraordinary enthusiasm and patient guidance have always been inspiring me to keep improving myself during my life. Prof. Anitescu always encourages me to look for big pictures, pursue fundamental questions and take challenges. To be Prof. Anitescu's student is always my greatest honor.

I always would like to thank Prof. Tengyuan Liang for his helpful discussion, advice, encouragement over time, and the enlightening reading group he organized. Prof. Liang gave me a brand new view of theoretical side of machine learning.

Moreover, I would like to thank Wei Biao Wu for agreeing be in my committee, his helpful discussion and comment on improving the thesis as well as the suggestions and the wonderful advanced theory class regarding dependent data he gave during my PhD years.

I would also like to thank my friends, Changji Xu, Bumeng, Zhuo, Yi Wang, Yi Liu, You-Lin Chen, Jing Yu, Mingrui Zhang, Qingcan Wang, Ruying Bao, Zhaorong Jin, Weitao Shuai, Cong Ma, Xinyi Liu, Lunyang Huang, Kaizheng Wang, Lirong Xue, Zongxi Li, Shaolei Du, Tianyi Zhang, Yun Zhou, Peigen Zhou, Zhan Lin, Ruiyi Yang, Pinhan Chen, Zehao Niu, Fengyi Li. I can never finish this journey without them.

In the end, I would like to express my deepest gratitude to my mother for her endless love and support.

ABSTRACT

In the thesis, we study the problems regarding robustness and model adaptivity with stochastic optimization.

First, we formally address two robust concerns. 1. Finite sample cannot well represents the entire population. 2. Data modeling assumptions can be wrong (misspecified). For the first robust concern, we propose an alternative of the popular regularization method based on distributionally robust optimization and clarify their connection and derive finite dimensional computational formulation based on that. For the second robust concern we study Huber's loss within a modern non-asymptotic setting. We further study the second robust concern with the stochastic gradient descent algorithm and purpose how to amend SGD to defense possibly maliciously outlier attack (which can consider as a stronger version of second robust concern) and justify the statistical optimality.

We then study the model adaptivity of training neural network by gradient flow via a dynamic reproducing kernel Hilbert space (RKHS) approach. We show that when reaching any local stationarity, gradient flow learns an adaptive RKHS representation and performs the global least-squares projection onto the adaptive RKHS simultaneously. This approach gives intuition of the benefits of training neural network over only viewing the neural network as a neural tangent kernel.

CHAPTER 1

INTRODUCTION

In this thesis, we study the problem of data-driven approaches to solving the following stochastic programming (SP)[157] problem,

$$(sp) : \min_{\theta \in \Theta} \; F(\theta) = \mathbb{E}_{X \sim P} \left[f(\theta, X) \right] := \langle f(\theta, X), P(X) \rangle. \tag{1.0.1}$$

Here f is some functional we want to minimize on a population level. We will call it a loss for this chapter, and θ is our decision/action/estimation, and X is data-driven from specific data generating process P. This model has many applications in statistics (e.g., maximum likelihood estimation, M-estimator) and machine learning (e.g., training linear model, kernel method, deep neural network).

Many works are studying this problem with *known* distribution P, see, e.g., [22, 157], whereas the probability distribution is inferred from history, expert advice, or modeling (e.g., physics, mechanics, economics, etc.). This line of work assumes we have *full* knowledge of the distribution P to solve (1.0.1).

In another extreme case, we do not have any knowledge about the distribution P other than the support $\mathcal{X}$. Many works in (traditional) robust optimization [18, 15, 14] suggest that we make the decision w.r.t. the worst case scenario within the support of the distribution

$$\min_{\theta \in \Theta} \; F_{robust}(\theta) = \max_{X \in \mathcal{X}} f(\theta, X). \tag{1.0.2}$$

However, in many real-world applications, we encounter neither the optimistic case when we have the full knowledge of the distribution (1.0.1) nor the pessimistic case when we have essentially no knowledge about the distribution (1.0.2). A more common and realistic setting is that we only have 'data' collected, preferably, $i.i.d$ $X_1, X_2, \cdots, X_n$ from distribution P and aim to infer some properties of the distribution and try to give a reasonable solution

based on the properties.

Perhaps the most naive way to handle this problem is to substitute the population distribution P by the empirical distribution $P_n = \sum_{i=1}^{n} \delta_{X_i}$. Therefore, this simple approach gives us a finite approximation of the stochastic programming problem (1.0.1), which is often called sample average approximation (SSA) or empirical risk minimization (ERM),

$$(saa) : \min_{\theta \in \Theta} \ \hat{F}(\theta) = \mathbb{E}_{X \sim P_n} [f(\theta, X)] = \frac{1}{n} \sum_{i=1}^{n} f(\theta, X_i) = \langle f(\theta, X), P_n(X) \rangle \approx F(\theta). \quad (1.0.3)$$

However, a sufficient condition (and arguably the most successful one of the very few theoretical understandings we have) to guarantee this approach to work is that $\hat{F}(\theta)$ is a good approximation of $F(\theta)$ for all $\theta \in \Theta$ (or, in other words, empirical process theory [177]), which is often not the case, especially when we have limited data sample, or, when the Θ space is huge, e.g., deep neural network. Because of this issue, simple SAA can suffer from the problem of overfitting [195], i.e. the in-sample loss $\langle f(\theta, X), P_n \rangle$ cannot well represent the out-sample (population) loss $\langle f(\theta, X), P \rangle$ and thereby we select a sub-optimal $\hat{\theta}$ based on SAA. This raises our **first robust concern**, which is putting too much faith on P_n as an approximation of P.

Arguably the most successful and commonly used approach to resolve the overfitting problem is **regularization**, e.g. we penalize θ based on its 'complexity', which yields the regularized version of SAA,

$$(soft) : \min_{\theta \in \Theta} \ \tilde{F}_\lambda(\theta) = \frac{1}{n} \sum_{i=1}^{n} f(\theta, X_i) + \lambda p(\theta) = \hat{F}(\theta) + \lambda p(\theta), \quad (1.0.4)$$

and theoretically, we also study the closely related constrained version of the problem (the

above (1.0.4) can be seen as a 'soft' version of the 'hard' constraint version below),

$$(hard) : \min_{\theta \in \Theta} \tilde{F}(\theta) = \frac{1}{n} \sum_{i=1}^{n} f(\theta, X_i) \quad \text{s.t. } p(\theta) \leq R. \tag{1.0.5}$$

Here the penalty function p is some complexity measure of θ, e.g. L_2-norm $\|\theta\|_2^2$, Reproducing Kernel Hilbert Space (RKHS) norm $\|\theta\|_{\mathcal{H}}^2$ (where the parameter θ represents a functional in the RKHS space), sparsity $\|\theta\|_0 = \{i \mid o_i \neq 0\}$, etc. Here we briefly go through the reasoning of regularization from a learning theoretic point of view [119, 151]. When the loss function f satisfies certain condition e.g. Lipchitz, boundness, we have the following uniform convergence guarantee,

$$\forall\, R > 0, \theta \in \Theta \text{ s.t. } p(\theta) = R, \quad |\langle f(\theta, X), P(X) \rangle - \langle f(\theta, X), P_n(X) \rangle| \leq \lambda \cdot R = \lambda \cdot p(\theta). \tag{1.0.6}$$

The second inequality holds either in expectation or in high probability. In other words, $\lambda \cdot p(\theta)$ is an upper bound of the in-sample out-sample difference
$\lambda p(\theta) \geq \sup_{\theta' \leq p(\theta)} \langle f(\theta', X), P - P_n \rangle$ (in expectation or with high probability). Therefore the target of (1.0.4) can be understood as follows,

$$F(\theta) = \langle f(\theta, X), P \rangle = \underbrace{\langle f(\theta, X), P_n \rangle}_{\hat{F}(\theta) \text{ in-sample loss}} + \underbrace{\langle f(\theta, X), P - P_n \rangle}_{\leq \lambda p(\theta) \text{ excess loss}} \leq \hat{F}(\theta) + \lambda p(\theta) = F_\lambda(\theta). \tag{1.0.7}$$

Here we call the in-sample and out-sample difference excess loss or generalization. Thereby, the regularization method can be understood as optimizing our decision θ w.r.t. an upper bound surrogate of the population loss. However, for many decision spaces Θ, there is no obvious complexity measure $p(\theta)$, also, both the complexity measure $p(\theta)$ and the general-

ization estimate $\lambda \cdot p(\theta)$ can be too loose. To make the matter worse, the derivation of λ needs certain properties of f, e.g. (higher-order) smoothness, which cannot always be easily calculated for real-world problems. We, therefore, do not aim to derive an upper bound of the population loss based on a complexity measure of the functional space Θ, but rather characterize the in-sample out-sample distribution difference, i.e., $P - P_n$ differently.

Suppose we have a characterization of the difference of P and P_n with certain metric on distribution space $d(P, P_n) \leq \epsilon_1$. Then following the same procedure as in (1.0.7), we can derive the following robust decision making approach

$$F(\theta) = \langle f(\theta, X), P \rangle = \underbrace{\langle f(\theta, X), P_n \rangle}_{\hat{F}(\theta) \text{ in-sample loss}} + \underbrace{\langle f(\theta, X), P' - P_n \rangle}_{d(P,P_n) \leq \epsilon_1 \text{ excess loss}} \quad (1.0.8)$$

$$\leq \underbrace{\langle f(\theta, X), P_n \rangle}_{\hat{F}(\theta) \text{ in-sample loss}} + \underbrace{\langle f(\theta, X), P' - P_n \rangle}_{d(P',P_n) \leq \epsilon_1 \text{ excess loss upper bound}} = \max_{d(P',P_n) \leq \epsilon_1} \mathbb{E}_{P'} [f(\theta, X)] . \quad (1.0.9)$$

Putting the minimization over decision variable back into the formulation (1.0.8), we have the following *distributionally robust optimization* (DRO)

$$\min_{\theta \in \Theta} \max_{d(P,P_n) \leq \epsilon_1} \mathbb{E}_P [f(\theta, X)] . \quad (1.0.10)$$

Here, SAA (1.0.3) can be seen as the case $\epsilon_1 = 0$, RO (1.0.2) can be seen as the case $\epsilon_1 = \infty$. We note that, the inner target function $\langle f(\theta, X), P \rangle$ is always linear in P. Hence, even if it is an infinite dimensional problem, it is still a linear programming problem w.r.t. P, and roughly speaking, by duality [155] (exchanging the infinity from target to constraints), we can transform a infinite dimensional minimization linear programming into a semi-infinite programming problem, [59, 201]. Also, with further concave-convex assumption on the target function f, the problem with d as Wasserstein-distance, can further be reduced into a finite dimensional convex-concave tractable problem [47, 66, 59]

In real-world applications, we often have the following specific form, where X takes the

form of a data pair (x, y) and we know that x has predictive power of y, and our ultimate goal is to optimize the following stochastic program with conditional distribution target

$$\min_{\theta \in \Theta} F(\theta, x) = \mathbb{E}_{y \sim P_{y|x}} \left[f(\theta, y) \right]. \tag{1.0.11}$$

Estimating the transition kernel $P_{y|x}$ can be very hard. To simplify the task, we borrow wisdom from the statistical model,

$$y = m^*(x) + \xi. \tag{1.0.12}$$

Here $m(x)$ is a function of the conditional mean of y given x, and the random vector ξ includes the remaining uncertain. Note that in this general model, the distribution of ξ actually depends on x. To further simplify the model, we assume the noise $\xi \sim P_\xi$ is homogeneous, i.e., the distribution of ξ is independent of x. Suppose we have data pair $D = \{(x_i, y_i)\}_{i=1}^n$, we propose the following *joint* model estimation and DRO formulation.

1. We have the estimation $\hat{m}$ obtained from estimation from data D with confidence region $m^* \in \{d_1(m, \hat{m}) \leq \epsilon_1\}$, with some (possibly functional) measure d_1.

2. Given any feasible m, we have empirical residuals $\hat{\xi}_i = y_i - m(x_i)$.

3. Let $P_n(m) = \frac{1}{n} \sum_{i=1}^n \delta_{y_i - m(x_i)}$ be the empirical distribution of the residual. Then we have a confidence interval of actual distribution P_ξ characterized by $d_2(P, P_n(\beta^*)) \leq \epsilon_2$, with some distributional metric d_2.

4. By 1) and 3), we have a joint ambiguity set $\mathcal{C}$, which has the form $d_1(m, \hat{m}) \leq \epsilon_1$ and $d_2(P_n(m), P)) \leq \epsilon_2$, and with high probability will include the actual (m^*, P_ξ).

Therefore, we have the following distributionally robust optimization with a new observation

x_{n+1}.

$$\min_{\theta\in\Theta}\max_{m,P}\mathbb{E}_{\xi\sim P}\left[f(\theta, m(x_{n+1})+\xi)\right] \tag{1.0.13}$$

$$\text{s.t. } d_1(m,\hat{m})\leq\epsilon_1 \tag{1.0.14}$$

$$d_2\left(P, P_n(m)\right)\leq\epsilon_2. \tag{1.0.15}$$

The **second robust concern** we want to address is that the sampled data $X_1, X_2, \cdots, X_n$ may not all be collected from our desired distribution P, but rather, some small portion of the data are outliers. This is a common phenomenon in statistics in practice [147], especially in time series [115]. In the statistics literature, [92], arguably the best model for the outlier issue is the Huber's contamination model [91, 89]. In this model, we have the following characterization of the actual data generating process for the data we gathered,

$$X_i\sim(1-\alpha)P+\alpha Q. \tag{1.0.16}$$

The model states that each data point X_i is drawn from our desired distribution P with probability $1-\alpha$, but with a small probability α, it is drawn from other arbitrary distribution Q. A closely related adversarial contamination model recently received a lot of attention due to its success in machine learning applications [79, 168] and as a theoretical interest in the theoretical computer science community. In this model, similar to Huber's contamination, with probability $1-\alpha$, we get a sample from P, but with probability α, we get a sample that can be possibly maliciously designed (based on the actual model, previous data given or even our specific algorithm). Under those two types of contamination models, two natural questions to answer are a) what type of outlier can we detect? and b) after screening out the detectable outliers, how well can we learn P based on the remaining possibly contaminated sample?

Still we work on problem (1.0.11), with the modelling assumption (1.0.12). The first prob-

lem we are going to address about **the second robust concern** is robust linear regression, namely, we model

$$y_i = \underbrace{\beta^* x_i}_{m^*(x_i)} + \xi_i + o_i, \tag{1.0.17}$$

where ξ_i is some generic well-behaved noise (zero-mean, light-tail and independent of x), and o_i is the contamination, i.e. the majority of $o_i = 0$. In the classical statistics literature, there are many classical works that study this problem. Their focus are mainly on 1. breakdown point of the estimator [46], i.e., the minimal portion of $o_i \neq 0$ to make the estimator arbitrarily bad, and 2. the efficiency of the estimator [46], which is, the performance of the estimator under no contamination $o_i = 0$. Under Huber's contamination model (1.0.16), the estimators in include M-estimator [90], GM-estimators [114], S-estimators [145] and MM-estimator [196], among others. In our work, we study the modern finite sample non-asymptotic performance [186] of Huber's loss induced M-estimator under the adversarial contamination model. After obtaining $\hat{\beta}$, we need to build P_n in step 3 to characterize a feasible set including P_ξ, and it naturally leads to the questions a) what undetectable adversarial sample could mislead us and b) what property does the trustworthy data pair (x_i, y_i) have so that we can characterized the closeness of the empirical residual distribution P_n and P_ξ.

The previous approach can be considered as a 'batch' way to solve the data-driven problem of (1.0.1) (using all n samples in one shot). We also study another alternative 'streaming' approach (sequentially sampling new data D_t and updating our decision variable $\theta_{t+1} = f_t(\theta_t, D_t)$ at each iteration) for (1.0.1), also known as stochastic approximation (SA) [143], under the second robust concern. Stochastic gradient descent (SGD) as a specialized SA is arguably the most successful algorithmic building block for modern data science and machine learning [26]. Previous works studying this problem [136] rely on a relatively

straightforward analysis, which plug in a robust estimator of the gradient at each step of SGD and perform inexact gradient descent analysis with adversarial oracle [42]. However, this approach overlooks the concentration effect of uncontaminated samples and cannot be generalized to the streaming case. By carefully studying the sample that can be kept from robust outlier removal, we refined the analysis and proposed an algorithm that achieves optimality in the statistical sense.

Moving from the robustness concern with data-driven stochastic optimization, we are going to study the adaptivity of training dynamics on a neural network with target (1.0.1). The traditional statistical and empirical process-based learning theory understanding failed to successfully explain the success of neural networks with a huge number of parameters [121]. The traditional understanding of model generalization, as mentioned in (1.0.7), is based on the upper bound of the following quantity

$$\max_{\theta\in\Theta} \left|\mathbb{E}_P\left[f(\theta, X)\right] - \mathbb{E}_{P_n}\left[f(\theta, X)\right]\right| \leq \sqrt{\frac{Complexity(\Theta)}{n}}, \tag{1.0.18}$$

where Θ denotes the neural network space. However, this failed to explain the large neural network because the large $Complexity(\Theta)$ and the bound above always give a vacuous bound. Recently, there has been a line of works that establish the connection between the training dynamics of overparametrizatized neural networks and a fixed RKHS [93, 52, 50]. An intuitive way of understanding this connection is as follows. The kernel function gives an 'inner product' structure with implicit feature mapping of the data-data pair, i.e.

$$K(x, y) = \langle \phi(x), \phi(y) \rangle, \tag{1.0.19}$$

where $\phi(\cdot)$ is the implicit feature mapping, and the RKHS functional space is $f(\theta, x) = b + \langle \phi(x), \theta \rangle$. Here, overparametrized neural network with proper random initialization can

be approximated by

$$f(\theta, x) \approx f(\theta_0, x) + \langle \nabla_\theta f(\theta_0, x), \theta - \theta_0 \rangle = b_0 + \langle \nabla_\theta f(\theta_0, x), \theta \rangle. \tag{1.0.20}$$

This is by taking the Taylor expansion around its initialization. If we can justify that along the entire training trajectory, θ will always stay close to initialization θ_0, then this linear approximation with feature mapping $x \to \nabla_\theta f(\theta_0, x)$ is a valid approximation. However, the natural question is that if the success of neural networks can all be explained by RKHS, why not directly use the Kernel method? Another relation of the kernel method and the two-layer neural network is simply viewing first layer as a feature mapping (each neuron as a feature), $\psi(x, \theta_1) = \big(\psi_1(x, \theta_{1,1}), \psi_2(x, \theta_{1,2}), \cdots, \psi_m(x, \theta_{1,m})\big)$, and the second layer as a linear activation function of the features

$$f(\theta, x) = \langle \theta_2, \psi(x) \rangle. \tag{1.0.21}$$

Here, θ_i denotes the weights on the i-th layer, and $\theta_{1,i}$ denotes the weights of neuron i on the first (hidden) layer. The relation is a more generic way to understand neural networks as kernel machines and yields another RKHS. In this line of work, we can study the two spaces (mainly their difference to show the benefit of using a neural network), namely, the RKHS found by the training neural network (1.0.21) and the linear approximation RKHS at initialization (1.0.20).

This thesis contains material from two published papers by the author [47, 49]. In particular, Chapter 2 is based on [47], coauthored with Mihai Anitescu; and Chapter 5 is based on [49], coauthored with Tengyuan Liang.

1.1 Contributions of This Thesis

1.1.1 DRO with Dependent Data

In this work, we first study the problem (1.0.11)

$$\min_{\theta\in\Theta} F(\theta, x) = \mathbb{E}_{y\sim P_{y|x}}\left[f(\theta, y)\right], \tag{1.1.1}$$

with modeling assumption (1.0.12)

$$y = m^*(x) + \xi. \tag{1.1.2}$$

Here we assume that $\xi \sim P_\xi$ homogeneously. Also, we make linear model assumption $m^*(x) = A^*x$. Following the (1.0.13), we have the following DRO formulation.

$$\min_{\theta\in\Theta}\max_{A,P} \mathbb{E}_{\xi\sim P}\left[f(\theta, Ax_{n+1} + \xi)\right] \tag{1.1.3}$$

$$\text{s.t. } \|A - \hat{A}\| \le \epsilon_1 \tag{1.1.4}$$

$$W_p^p\left(P, P_n(A)\right) \le \epsilon_2. \tag{1.1.5}$$

Here W is the p-Wasserstein distance. By duality arguments, when function f is convex in the first argument and convex in the second argument, we show that this problem has a convex-concave finite-dimensional tractable formulation, which can be efficiently solved. We also applied this technique to the case when we observe a time series from vector autoregression (VAR), $z_{t+1} = Az_t + \xi_t$ [173].

1.1.2 DRO with Contamination

With the same target (conditional stochastic programming) and model as the work before, here we consider that case when we have contaminated data pair (x_i, y_i). We investigate

the performance of Huber's loss with finite sample for non-asymptotic guarantees. We also, design a data screening rule based on Huber's loss regression and study what type of contamination can be hidden from our screening rule and the distributional influence of such undetectable contamination on estimating P_ξ. On the DRO formulation side, we study the following problem

$$\min_{\theta\in\Theta}\max_{A,P}\mathbb{E}_{\xi\sim P}\left[f(\theta, Ax_{n+1}+\xi)\right] \tag{1.1.6}$$

$$\text{s.t. } \|A-\hat{A}\|\leq\epsilon_1 \tag{1.1.7}$$

$$W_p^p\left(P, P_n(A)\right)\leq\epsilon_2. \tag{1.1.8}$$

Based on the optimal transport theory, we show that for general continuous f, we can always approximate (1.1.6) with a finite-dimensional min-max optimization problem with a solution difference guarantee.

We also generalize this approach to time series setting with innovative outliers. The difficult part is that $z_{n+1}=x_{n+1}$ can be contaminated as well. Therefore, the formulation (1.1.6) needs to be adjusted to the last trust worth point z_{last}, but that will incur the noise residual to be the form $\xi_{n+1}+A\xi_n+\cdots+A^c\xi_{last}$. We propose two relaxations to handle this issue.

1.1.3 Making SGD Robust against Outlier Attack

In this work, we consider solving the stochastic programming problem (1.0.1) by SGD, i.e., at each step t, we get a (potentially contaminated) sample X_t, and get the noise gradient $g_t=\nabla f(\theta_t, X_t)$ and try to perform gradient descent,

$$\theta_{t+1}=\arg\min_{\theta\in\Theta}\left[\langle g_t, \theta-\theta_t\rangle+\frac{1}{2\eta_t}\|\theta-\theta_t\|^2\right]:=\prod_{\theta\in\Theta}\left[\theta_t-\eta_t g_t\right]. \tag{1.1.9}$$

Here we use $\prod$ as a projection operator. We develop a simple, robust screening rule based on the historical data. Using historical data, we develop an estimator of $\mu(\theta_t) \approx \nabla F(\theta)$, with confidence radius V and we rule out g_t if it doesn't fail into the confidence region. Note that the reuse of data will make the future trajectory dependent on the historical data, and this dependency can make certain martingale properties fail, which is crucial in analyzing the convergence of Vanilla SGD in the contamination-free setting. This dependency can also make the confidence radius V vacuously large, which will make the convergence rate sub-optimal. We handle both issues by studying this empirical process and proving this simple screening approach can be amended to achieve statistical optimality.

1.1.4 NN as Adaptive Kernel

Consider the problem: given data pair (x, y) drawn from a population with $f_*(x') = \mathbb{E}[y|x = x']$, specify a neural network model and run gradient flow on the weights over time until reaching any stationarity. How does f_t, the function computed by the neural network at time t, relate to f_* in terms of approximation and representation? What are the provable benefits of the adaptive representation by neural networks compared to the pre-specified fixed basis representation in the classical nonparametric literature? We answer the above questions via a dynamic reproducing kernel Hilbert space (RKHS) approach indexed by the training process of neural networks. We show that when reaching any *local* stationarity, gradient flow learns an adaptive RKHS representation and performs the *global* least-squares projection onto the adaptive RKHS simultaneously. Besides, we prove that as the RKHS is data-adaptive and task-specific, the residual for f_* lies in a subspace that is potentially much *smaller* than the orthogonal complement of the RKHS, formalizing the representation and approximation benefits of neural networks. To further substantiate the adaptive theory, we show that in the limit of vanishing regularization, the neural network function computed by gradient flow converges to the kernel ridgeless regression with the adaptive kernel.

CHAPTER 2

DISTRIBUTIONALLY ROBUST OPTIMIZATION WITH CORRELATED DATA FROM VECTOR AUTOREGRESSIVE PROCESSES

2.1 Introduction

A common formulation of optimization under uncertainty is the following stochastic program: [156]

$$\min_{x \in \mathcal{D}} \mathbb{E}_{\mathbf{y} \sim F}[h(x, \mathbf{y})].$$

Here the decision variable x has a convex feasible domain $\mathcal{D}$, h is a convex function in x, and y is data from an underlying generating process.

Much of the work in stochastic programming is carried out under the assumption that the distribution F is known [22, 156]. In many problems, however, other than some basic properties, we do not have the exact description of F. Using the empirical distribution F_n as a surrogate for F would overfit the data, especially with very few samples.

One way to overcome the uncertainty attached to the probability density F itself is to investigate *distributionally robust stochastic optimization* (DRSO). This problem is

$$\min_{x \in \mathcal{D}} \max_{F \in \mathcal{U}} \mathbb{E}_{\mathbf{y} \sim F}[h(x, \mathbf{y})],$$

where the distribution F is from a set $\mathcal{U}$. Significant research recently has been carried out concerning the choice of ambiguity set $\mathcal{U}$ by trying to balance out-of-sample performance and computational complexity. In [41], the authors proposed to specify the ambiguity set by the first and one-sided second moment constraints in order to preserve the convexity of the formulation. As mentioned in [67], however, the one-sided second-moment constraint may have no effect on the problem.

Other approaches identify the ambiguity set by considering distributions that are close to the empirical distribution in an appropriate measure. Different metrics include the Kullback-Leibler divergence [95], Burg entropy [187], total variation [167], χ^2-distance [102], and more generally ϕ-divergence [16] [9]. A drawback of ϕ-divergence formulations is that they may not be rich enough to capture distributions of interest [66].

Recent work has introduced DRSO formulations based on the Wasserstein distance [58, 66], which has both out-of-sample performance guarantees and computational efficient reformulations.

Most of the cited references study the problem in a setting where the data consists of copies of independent and identically distributed (i.i.d) random variables. However, in many applications, particularly when the data is formed by sequential entries in a time series, the i.i.d. assumption is not realistic. In this work, we study the case when we have times series data from a *vector autoregression* (VAR) process

$$\mathbf{y}_{t+1} = A\mathbf{y}_t + \boldsymbol{\xi}_t,\ t = 1, \ldots, n-1. \tag{2.1.1}$$

Here $\xi_1, \ldots, \xi_{n-1} \in \mathbb{R}^d$ are i.i.d random variables with a zero-mean residual. Realizations $y_1, \ldots, y_n \in \mathbb{R}^d$ are our observations, and the conditional expectation satisfies $\mathbb{E}_{t-1}[\mathbf{y}_t] = A\mathbf{y}_{t-1}$. For a more concise way to represent the model, let $Y_+ = [y_2, \ldots, y_n] \in \mathbb{R}^{d\times(n-1)}$, $Y_- = [y_1, \ldots, y_{n-1}] \in \mathbb{R}^{d\times(n-1)}$ and $E = [\xi_1, \ldots, \xi_{n-1}] \in \mathbb{R}^{d\times(n-1)}$. Thus, we have

$$Y_+ = AY_- + E. \tag{2.1.2}$$

The VAR model is widespread. It occurs in econometrics [160], control theory [105], and recent brain image analysis [64]. The ubiquity of times series models motivates us to generalize the DRSO to the VAR-dependent data setting.

Our contribution is to propose a DRSO formulation using Wasserstein distance techniques

for VAR data and to prove that if the original problem has the needed convexity features, then our DRSO formulation is a finite-dimensional convex-concave saddle point problem.

2.2 Model and Robust Formulation

Suppose we have data $y_1, y_2, \cdots, y_n \in \mathbb{R}^d$ from a time series and we need to make a decision that will be affected by the next outcome y_{n+1}. We assume that the time series is generated by a vector autoregression process (VAR(1)),

$$\mathbf{y}_{t+1} = \beta + A\mathbf{y}_t + \boldsymbol{\xi}_t.$$

Here A is a fixed transition matrix, and the noise terms $\xi_t \in \Xi \subseteq \mathbb{R}^d,\ t = 1, \ldots, n-1$ are i.i.d with zero mean. For notational simplicity, let $\tilde{\mathbf{y}}_t = [1, \mathbf{y}_t]^T$, $\tilde{\boldsymbol{\xi}}_t = [0, \boldsymbol{\xi}_t]^T$, and

$$\tilde{A} = \begin{bmatrix} 1 & 0 \\ \beta & A \end{bmatrix}.$$

Therefore, the data model becomes

$$\tilde{\mathbf{y}}_{t+1} = \tilde{A}\tilde{\mathbf{y}}_t + \tilde{\boldsymbol{\xi}}_t,\ t = 1, 2, \ldots, n-1.$$

To simplify notation, we will use, without loss of generality, $\mathbf{y}_t$ and A in the previous equation for the rest of the article; in other words, we will refer to the algebraic formalism of (2.1.1) and (2.1.2).

Consider the stochastic programming problem

$$\begin{array}{ll} \min\limits_{x} & \mathbb{E}_{\mathbf{y}_{n+1} \sim F}[h(x, \mathbf{y}_{n+1})] \\ \text{subject to} & x \in \mathcal{D}. \end{array} \tag{2.2.1}$$

Here F is the true conditional distribution of y_{t+1} given y_t within the model. For problems with real data, both A and F need to be estimated from the data. We can build confidence intervals of A and F under common regularity assumptions about the noise term ξ_t, which lead to our robust formulation. We consider the DRSO problem with decision variable x informed by incoming data from process (2.1.1):

$$\begin{aligned} \min_{x} \max_{A,F} \quad & \mathbb{E}_{\boldsymbol{\xi}_n \sim F}[h(x, Ay_n + \boldsymbol{\xi}_n)] \\ \text{subject to} \quad & d_1(A, \hat{A}) \leq \varepsilon_2 \\ & F \in \mathcal{U} \\ & x \in \mathcal{D}. \end{aligned} \tag{2.2.2}$$

Here $\hat{A}$ is a fixed matrix obtained by regression based on the matrix formulation (2.1.2):

$$\hat{A} = \arg\min_{B} \|Y_+ - BY_-\|_F^2 + \lambda \|B\|_F^2.$$

Here $\|\cdot\|_F$ is the Frobenius norm of the matrix. Concerning the structural matrix A, we impose an estimation accuracy constraint on $\hat{A}$, whereby A and $\hat{A}$ have to be relatively close. It is well known that the accuracy of the regression matrix $\hat{A}$ depends on the condition number of the design matrix Y_- [107, 77].

Later we will specify the choice of ε_2 such that, with high probability, the true matrix A satisfies that constraint. For each choice of A, we get the residual $\hat{\xi}_i = y_{i+1} - Ay_i$, $i = 1, \ldots, n-1$. Let F_n be the empirical distribution of $\hat{\xi}_i$; that is, $F_n = \frac{1}{n-1} \sum_{t=1}^{n-1} \delta_{\hat{\xi}_i}(\xi)$. Note that F_n depends on A, which is itself a variable in (2.2.2), but for simplicity of notation, we do not explicitly indicate that. The family of the distribution, $\mathcal{U}$, is specified by constraining the distribution F relative to the empirical distribution F_n by means of a specially chosen

distance function

$$\mathcal{U} = \left\{ F \;\middle|\; \begin{array}{l} F(\xi \in \Xi) = 1 \\ d_w(F, F_n) \leq \varepsilon_1 \end{array} \right\}. \tag{2.2.3}$$

The ambiguity set $\mathcal{U}$ is a subset of the distributions on the measurable space $(\mathbb{R}^d, \mathcal{B})$, where $\mathcal{B}$ is the σ-algebra of the Borel sets. The first condition constrains the support of the distribution to a known set Ξ, and the second constraint regulates the behavior of the noise term. In the following, we will assume this set to be bounded. The existence of a known set that contains the support of the distribution is a common assumption with other approaches [41, 66], at least when aiming for results comparable to ours, as well as a reasonable approach for most physical and economical processes.

Wasserstein Distance

The quantity d_w is the Wasserstein distance, which can be defined as follows.

Definition 2.2.1. Let $\mathbb{P}, \mathbb{Q}$ be two distributions on a metric space (X, d). The 2-Wasserstein distance can be defined by

$$d_w(\mathbb{P}, \mathbb{Q}) = \inf_{\pi \in \Pi(\mathbb{P}, \mathbb{Q})} \sqrt{\int_{X \times X} d^2(x, x') \pi(\mathrm{d}x, \mathrm{d}x')}.$$

Here $\Pi(\mathbb{P}, \mathbb{Q})$ is the family of distributions on $X \times X$ with marginal distributions $\mathbb{P}$ and $\mathbb{Q}$ [183].

This definition can be viewed as finding an optimal transport between two distributions, while the cost of moving probability mass is encoded by the distance $d(x, x')$ on the metric space X. Although this definition appears daunting, the key observation, also used in [58], is that when $\mathbb{Q}$ is the empirical distribution $F_n = \frac{1}{n-1} \sum_{i=1}^{n-1} \delta_{\hat{\xi}_i}(\xi)$, we can compute $d_w(\cdot, \cdot)$ relatively easily since we can always break $\mathbb{P}$ down to the sum of $n-1$ conditional distributions

$\mathbb{P}_i$. Subsequently, by utilizing duality, we will convert the resulting infinite-dimensional optimization problem (2.2.2) into a computable finite convex problem.

2.3 Problem Formulation and Dual Representation

We now formally state our DRSO version of (2.2.2):

$$\begin{aligned} \min_{x} \max_{A,F} \quad & \mathbb{E}_{\boldsymbol{\xi}_n \sim F} h(x, \mathbf{y}_{n+1}) \\ \text{subject to} \quad & d_w^2(F, F_n) \leq \varepsilon_1 \\ & A \in \Omega(\epsilon_2) \\ & x \in \mathcal{D}, \end{aligned} \tag{2.3.1}$$

where $F_n = \frac{1}{n-1} \sum_{i=1}^{n-1} \delta_{\tilde{\xi}_i}(\xi)$ is the empirical distribution of $\tilde{\xi}_i = y_{i+1} - A y_i$, $i = 1, \ldots n-1$, and Ω defines the uncertainty set of the matrix A,

$$\Omega(\epsilon_2) \doteq \left\{ A \in \mathbb{R}^{d \times d} \,\middle|\, \left\| A_i - \hat{A}_i \right\| \leq \varepsilon_{2,i} \text{ , for } i \in [d] \right\}. \tag{2.3.2}$$

The second constraint in (2.3.1) is the confidence interval of A for which we can choose $\varepsilon_{2,i}$ based on regression analysis [62] (see also the end of §2.4). Specifically, we denote here and in the following by A_i the i-th row of matrix $A \in \mathbb{R}^{d \times d}$ and by $\hat{A}_i$ the i-th row of matrix $\hat{A}$.

Reformulation

Writing now expectations in integral form and recalling our specification of the choice of support ξ in (2.2.3) and of the objects in Definition 2.2.1, we have the following.

$$\begin{aligned} \min_{x\in\mathcal{D}} \quad \max_{A\in\Omega(\epsilon_2),F,\pi\in\Pi(F,F_n)} \quad & \int_{\Xi} h(x, Ay_n+\xi)\,\mathrm{d}F(\xi) \\ \text{subject to} \quad & \int_{\Xi} \mathrm{d}F(\xi) = 1 \\ & \int_{\Xi\times\Xi} \|\xi-\xi'\|^2 \mathrm{d}\pi(\xi,\xi') \le \varepsilon_1 \end{aligned} \tag{2.3.3}$$

Here the second constraint is a rewrite of the Wasserstein distance constraint using Definition 2.2.1. From the definition of $\Pi(F, F_n)$, the joint distribution $\pi \in \Pi(F, F_n)$ has marginal distributions F and F_n. Since F_n is the empirical distribution, by the rules of conditional distributions we have that $\pi(\xi,\xi') = \frac{1}{n-1}\sum_{i=1}^{n-1} \pi(\xi|\xi' = \hat{\xi}_i)\delta_{\hat{\xi}_i}(\xi')$, where $F_i \doteq \pi(\xi|\xi' = \hat{\xi}_i)$ is the conditional distribution of ξ given that ξ' takes the value $\hat{\xi}_i$.

Then we have $F = \sum_{i=1}^{n-1} \mathbb{P}(\xi' = \hat{\xi}_i)F_i = \frac{1}{n-1}\sum_{i=1}^{n-1} F_i$.

We note that, as a conditional distribution, F_i is constrained at this stage only by having the same support as F. Also note that since $\pi(F, F_n)$ can be used to define F (as one of its marginals), we substitute F as above and reformulate the optimization problem with the conditional probabilities as the variables (similar to [58]).

We obtain

$$\begin{aligned} \min_{x\in\mathcal{D}} \quad \max_{A\in\Omega(\epsilon_2),\{F_i\}_{i=1}^{i=n-1}} \quad & \frac{1}{n-1}\sum_{i=1}^{n-1}\int_{\Xi} h(x, Ay_n+\xi)\,\mathrm{d}F_i(\xi) \\ \text{subject to} \quad & \int_{\Xi} \mathrm{d}F_i(\xi) = 1,\ i = 1,2,\ldots,n-1 \\ & \frac{1}{n-1}\sum_{i=1}^{n-1}\int_{\Xi} \|\xi-\hat{\xi}_i'\|^2 \mathrm{d}F_i(\xi) \le \varepsilon_1. \end{aligned} \tag{2.3.4}$$

Now, we reduce the above distributional optimization problem into a finite-dimensional

problem.

Theorem 2.3.1. *Let $\Phi(x, A)$ denote the solution of the inner maximizing problem with fixed x and A in (2.3.4). When $h(x, y)$ is differentiable, convex in the first argument and concave in the second argument, we have the following identity:*

$$\Phi(x, A) = \min_{u \geq 0} u\varepsilon_1 + \max_{\xi_i \in \Xi, i \in [n-1]} \left\{ \frac{1}{n-1} \sum_{i=1}^{n-1} \Big[h(x, Ay_n + \xi_i) - u \cdot \ \|\xi_i - (y_{i+1} - Ay_i)\|^2 \Big] \right\}.$$

Proof. From Lagrangian duality, we get that $\Phi(x, A)$ equals

$$\max_{F_i, i \in [n-1]} \inf_{u \geq 0} u \left[\varepsilon_1 - \frac{1}{n-1} \sum_{i=1}^{n-1} \int_{\Xi} \|\xi_i - \tilde{\xi}_i'\|^2 F_i(\mathrm{d}\xi) \right] \frac{1}{n-1} + \sum_{i=1}^{n-1} \int_{\Xi} h(x, Ay_n + \xi) \mathrm{d}F_i(\xi)$$

$$= \inf_{u \geq 0} u\varepsilon_1 + \max_{F_i, i \in [n-1]} \left\{ \frac{1}{n-1} \sum_{i=1}^{n-1} \int_{\Xi} h(x, Ay_n + \xi_i) - u\|\xi_i - \tilde{\xi}_i\|^2 F_i(\mathrm{d}\xi_i) \right\}$$

$$= \inf_{u \geq 0} u\varepsilon_1 + \max_{\xi_i \in \Xi, i \in [n-1]} \left\{ \frac{1}{n-1} \sum_{i=1}^{n-1} [h(x, Ay_n + \xi_i) - u\|\xi_i - (y_{i+1} - Ay_i)\|^2 \Big] \right\}.$$

The second equality occurs from exchanging min and max, which is valid by strong duality. This can be proved by an extended version of a well-known strong duality result for

moment problems [155], similar to the argument in [58, Theorem 4.2].

The third equality stems from the fact that the maximum over distributions F_i with respect to the integral is equal to the maximum point of the integrand. □

From Theorem 2.3.1 our DRSO formulation (2.3.1) is equivalent to

$$\begin{aligned}&\inf_{x\in\mathcal{D}}\max_{A\in\Omega(\epsilon_2)}\inf_{u\geq 0}\max_{\xi_i\in\Xi,i\in[n-1]} u\varepsilon_1+\\&\left\{\frac{1}{n-1}\sum_{i=1}^{n-1}\left[h(x,Ay_n+\xi_i)-u\|\xi_i-(y_{i+1}-Ay_i)\|^2\right]\right\}.\end{aligned}\tag{2.3.5}$$

We can now state our main result.

Theorem 2.3.2. *Problems* (2.3.1) *and* (2.3.5) *are equivalent to the convex-concave minimax problem:*

$$\begin{aligned}\inf_{x\in\mathcal{D}}\max_{A,\xi_i\in\Xi,i\in[n-1]}\ &\frac{1}{n-1}\sum_{i=1}^{n-1}h(x,Ay_n+\xi_i)\\ s.t.\ &\frac{1}{n-1}\sum_{i=1}^{n-1}\|\xi_i-(y_{i+1}-Ay_i)\|\leq\varepsilon_1,\\ &\left\|A_i-\hat{A}_i\right\|\leq\varepsilon_{2,i},\ \textit{for } i\in[d].\end{aligned}\tag{2.3.6}$$

Proof.

$$\begin{aligned}&\Psi(x,u,A)\doteq\max_{\xi_i\in\Xi,i\in[n-1]} u\varepsilon_1+\\&\left\{\frac{1}{n-1}\sum_{i=1}^{n-1}\left[h(x,Ay_n+\xi_i)-u\|\xi_i-(y_{i+1}-Ay_i)\|^2\right]\right\}\end{aligned}\tag{2.3.7}$$

is both a maximum of affine functions in u and a maximum of functions *jointly concave* in $(A,\{\xi_i\})$. Therefore, it is convex in u and concave in A. Since the feasible set of A is

bounded, by Sion's minimax theorem [161, Thm.3.4], Equation (2.3.5) becomes

$$
\begin{aligned}
&\inf_{x\in\mathcal{D}} \max_{A\in\Omega(\epsilon_2)} \inf_{u\geq 0} \Psi(x,u,A) \\
&\overset{[161,\ \text{Thm.3.4}]}{=} \inf_{x\in\mathcal{D}} \inf_{u\geq 0} \max_{A\in\Omega(\epsilon_2)} \Psi(x,u,A) \\
&= \inf_{x\in\mathcal{D}} \inf_{u\geq 0} \max_{A\in\Omega(\epsilon_2),\xi_i\in\Xi,i\in[n-1]} \\
&u\varepsilon_1 + \frac{1}{n-1}\sum_{i=1}^{n-1}\left[h(x,Ay_n+\xi_i) - u\|\xi_i - (y_{i+1}-Ay_i)\|^2\right] \\
&\overset{[161,\ \text{Thm.3.4}]}{=} \inf_{x\in\mathcal{D}} \max_{A\in\Omega(\epsilon_2),\xi_i\in\Xi,i\in[n-1]} \inf_{u\geq 0} \\
&u\varepsilon_1 + \frac{1}{n-1}\sum_{i=1}^{n-1}\left[h(x,Ay_n+\xi_i) - u\|\xi_i - (y_{i+1}-Ay_i)\|^2\right].
\end{aligned}
$$

By strong duality applied to the innermost problem, the conclusion follows, after unfolding the definition of $\Omega(\epsilon_2)$ (2.3.2). □

The important consequence of Theorem 2.3.2 is that (2.3.1) can be solved efficiently by solving the equivalent problem (2.3.6) techniques such as [123].

2.4 Concentration Inequalities

We now aim to connect the relaxation parameters ε_1 and ε_2 to the probability of the true probability distribution satisfying the relaxed constraints. We assume that a bound for the support is known, similar to [41].

Assumption 2.4.1. *There exists an $R > 0$ such that for the noise term ξ, we have $\|\xi\| \leq R$.*

We note that the boundedness assumption can be relaxed by requiring square-exponential integrability. This would require techniques for unbounded distributions that involve the Wasserstein distance concentration, as presented in [58], and consistency of the transition

matrix estimation (A) (see, e.g., [27]). For brevity, we will focus on the bounded support case only. We also need the following boundness assumption on A.

Assumption 2.4.2. *There exists an $K > 0$ such that for the transition matrix A, we have $\|A\|_2 \leq K$.*

Note that this assumption also ensures the norm of each row of A satisfies,

$$\|A_i\| = \|e_i^T A\| \leq K. \tag{2.4.1}$$

Lemma 2.4.3. (Wasserstein metric concentration, specification of ε_1)
Suppose $\xi_1, \xi_2, \cdots, \xi_n \in \mathbb{R}^d$ are i.i.d samples from a distribution F with zero mean and that satisfy Assumption 1. Then, for the empirical distribution F_n, the following inequality holds:

$$\mathbb{P}(d_w^2(F, F_n) \geq \varepsilon) \leq C_0 \exp\left(-C_1 N \varepsilon^{d/2}\right). \tag{2.4.2}$$

Here C_0, C_1 depend only on R and d.

Proof. The result is an immediate consequence of [58, Theorem 3.4], where we chose $a = 2$ and used Assumption 1 for bounding A from that statement. □

We also note that C_0, C_1 are explicitly computable by using techniques such as in [66, Appendix B]. Now, we can select the right-hand side of (2.4.2) to the confidence level, for example, 0.05. This will be a conservative estimate, however, and we will use cross-validation in practice to compute a suitable ϵ_1, as we will discuss in §2.5. The important feature of Lemma 2.4.3, however, is the exponential decay of the failure probability with N and ϵ^2.

Lemma 2.4.4. (self-normalized process [1, Theorem 1])
Let $\{\mathcal{F}_t\}_{t=0}^{\infty}$ be a filtration. Let η_t be a real-valued stochastic process such that η_t is $\mathcal{F}_t$-

measurable and η_t is conditionally R-sub-Gaussian i.e.

$$\forall \lambda \quad \mathbb{E}\left[e^{\lambda \eta_t} \mid \mathcal{F}_{t-1}\right] \leq \exp\left(\lambda^2 R^2/2\right). \tag{2.4.3}$$

Let X_t be an $\mathbb{R}^d$-valued stochastic process such that X_t is $\mathcal{F}_{t-1}$ measurable. Assume that V is a $d \times d$ positive definite matrix. For any $t \geq 0$, define

$$\bar{V}_t = V + \sum_{s=1}^{t} X_s X_s^T \quad S_t = \sum_{s=1}^{t} \eta_s X_s. \tag{2.4.4}$$

Then, for any $\delta > 0$, with probability greater than $1-\delta$, for all $t \geq 0$, we have

$$\|\bar{V}_t^{-1/2} S_t\|_2^2 \leq 2R^2 \log\left(\frac{\det(\bar{V}_t)^{1/2} \det(V)^{-1/2}}{\delta}\right) \tag{2.4.5}$$

Specification of $\varepsilon_{2,i}$ Recall $\hat{A}$ is obtained by solving the following ridge regression problem,

$$\hat{A} = \arg\min_B \|Y_+ - BY_-\|_F^2 + \lambda \|B\|_F^2.$$

Due to the separable properties of the formulation, we have the following solution according to each row of $\hat{A}$,

$$\hat{A}_i = \arg\min_{B_i} \sum_{t=1}^{n-1} \|y_{t+1,i} - y_t^T B_i\|_F^2 + \lambda \|B_i\|_F^2. \tag{2.4.6}$$

Recall that $\hat{A}_i$ denotes the i-th row of $\hat{A}$ (as a column vector). By the normal equation, note

that $y_{t+1,i} = y_t^T A_i + \xi_{t,i}$, we have

$$\hat{A}_i = \left(\lambda I + \sum_{t=1}^{n-1} y_t y_t^T\right)^{-1} \left(\sum_{t=1}^{n-1} y_t y_{t+1,i}\right) \tag{2.4.7}$$

$$= \left(\lambda I + \sum_{t=1}^{n-1} y_t y_t^T\right)^{-1} \left(\sum_{t=1}^{n-1} y_t (y_t^T A_i + \xi_{t,i})\right) \tag{2.4.8}$$

$$= A_i - \underbrace{\lambda \left(\lambda I + \sum_{t=1}^{n-1} y_t y_t^T\right)^{-1} A_i}_{(I)} \tag{2.4.9}$$

$$+ \left(\lambda I + \sum_{t=1}^{n-1} y_t y_t^T\right)^{-1/2} \underbrace{\left(\lambda I + \sum_{t=1}^{n-1} y_t y_t^T\right)^{-1/2}}_{\bar{V}_{n-1}^{-1/2}} \underbrace{\left(\sum_{t=1}^{n-1} y_t \xi_{t,i}\right)}_{S_{n-1}} \tag{2.4.10}$$

By Assumption 1 and Hoeffding's lemma [86], we know that $\xi_{t,i}$ is bounded by R, therefore is R-sub-Gaussian, and by Assumption 2 and (2.4.1) applying on (I), with Lemma 2.4.4 applying on the last term, we have the following data-dependent bound on $\|\hat{A}_i - A_i\|$,

$$\|\hat{A}_i - A_i\| \leq \frac{\lambda K}{\lambda_{min}} + \sqrt{\frac{2R^2 \log\left(\det(V_{n-1})^{1/2} \det(\lambda)^{-1/2}/\delta\right)}{\lambda_{min}}} \tag{2.4.11}$$

Here λ_{min} is the minimal eigen-value of $\bar{V}_{n-1}$. With this bound, we can now specify the choice of $\varepsilon_{2,i}$.

Applying Lemma 2.4.4 to each column of $\hat{A}$ for confidence level $1 - \frac{\delta}{d}$, and using Boole's inequality to the complements,

we have, with probability greater than $1 - \delta = 1 - d\frac{\delta}{d}$, that $\|A_i - \hat{A}_i\| \leq \varepsilon_{2,i}$ with

$$\varepsilon_{2,i} \leq \frac{\lambda K}{\lambda_{min}} + \sqrt{\frac{2R^2 \log\left(d \det(V_{n-1})^{1/2} \det(\lambda)^{-1/2}/\delta\right)}{\lambda_{min}}}.$$

2.5 Experiments

We apply the DRSO approach (2.3.1) in the variant outlined in Theorem 2.3.2 to a portfolio optimization problem.

The decision variable x is constrained to the $(d-1)$-dimensional standard simplex $\mathcal{D} = \{x \in \mathbb{R}^d | x_1 + \cdots + x_d = 1,\, x_i \geq 0,\, i = 1, \ldots, d\}$. The variable x represents the portions of investment in different stocks. Here the data $y_t \in \mathbb{R}^d$ is the price of d different stocks at time t. In the framework of (2.2.1), the objective function is the (negative) return

$$h(x, y) = -\langle x, y \rangle.$$

We subsequently solve the distributionally robust problem (2.3.6) that is derived from our main result, Theorem 2.3.2, with the convex, spherical Ξ from Assumption 1. We report on those results in the rest of this section and label them as "DRO."

The DRSO problem (2.3.6) was solved with the saddle point algorithm from [123] implemented in Julia and run on a MacBook Pro, 2.4 GHz Intel Core i5, 8 GB 1600 MHz DDR3. The computation time of 100 experiments for either synthetic or real data cases below for $n = 21$, $d = 8$ (20 time periods) did not exceed 300 seconds.

2.5.1 Synthetic Data

For our experiment with synthetic data, the feasible set $\mathcal{D}$ is the $(d-1)$-dimensional standard simplex, and we set $d = 8$. The objective function is the inner product $-\langle x, y \rangle$. Here y_i is from the VAR(1) times series, with the transition matrix entrywise drawn from uniform distribution over $[0, 1]$, then scaled so that $\|A\| = 0.8$ and ξ_t is from $N(0, R^2 I)$, then truncated to 2-norm no greater than a preset radius R. The metric we use in the Wasserstein distance constraint is the 2-norm in Euclidian space $(\mathbb{R}^d, \|\cdot\|)$. The radii of confidence intervals from §2.4 are conservative. For better performance, we shrink the parameters ϵ_1, ϵ_2 by factors

Setup	Median	75th Perc.	90th Perc.
8-4-5	**0.49**/0.59/0.52	0.91/**0.74**/0.78	1.27/**0.92**/1.09
8-16-5	**0.42**/0.55/0.47	0.81/**0.72**/0.75	1.14/**0.86**/1.05
16-4-5	**0.34**/0.57/0.41	0.72/0.72/**0.64**	1.08/**0.86**/**0.86**
16-16-5	**0.38**/0.57/0.43	0.76/0.70/**0.68**	1.03/**0.82**/0.92
16-4-8	**0.43**/0.53/0.48	0.75/**0.66**/**0.66**	0.99/**0.76**/0.80
16-16-8	**0.43**/0.54/0.46	0.73/**0.65**/0.68	1.00/**0.77**/0.83

Table 2.1: Comparison of MLE, SAA, and DRO for several standard percentiles, synthetic data. Setup is "sample size-$(n-1)$-noise radius-(R)-dimension(d)." Different statistics are normalized by noise radius R and given by "MLE/DRO/SAA." Lowest regret among methods is boldfaced.

$1, 0.5$ on the first 40 data points and choose the combination with the best outcome. In [58], the authors tried different confidence levels δ, which fundamentally resulted in the same effect. We compare the solution of DRO x^d and the solution of sample average approximation (SAA) x^s with the empirical residuals.

Here our SAA solution is obtained by

$$\begin{aligned} \min_x \max_A \sum_{i=1}^{n-1} h(x, Ay_n + \hat{\xi}_i) \\ \text{s.t.} \|A_i - \hat{A}_i\| \leq \varepsilon_{2,i},\ i \in [d]. \end{aligned} \tag{2.5.1}$$

We also calculate the solution of the deterministic version of (2.3.1) obtained by plugging in the maximum likelihood estimator (MLE) of y_n, $\hat{A}y_{n-1}$. Let $x^* = \arg\min h(x, y_{n+1})$ (solution with perfect information). We report the "regret" $h(x, y_{n+1}) - h(x^*, y_{n+1})$ for x given by the different approaches. Some empirical quantiles are given in Table 2.1, and two histograms are given in Figures 2.1 and 2.2. As we can see from the results, with more training samples or lower noise levels, the estimated transition matrix $\hat{A}$ becomes more accurate, so regression results in a decision closer to the perfect one most of the time. In all scenarios, however, DRSO has a lighter tail (see also Figures 2.1 and 2.2), which demonstrates the robustness of our decision. In particular, in Table 2.1, DRSO exhibits the smallest regret for all experiments at the 90th quantile.

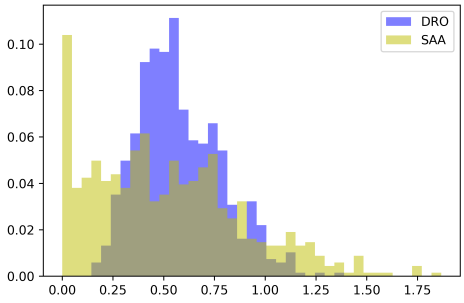


Figure 2.1: Comparison of DRO and SAA, synthetic data. All data are normalized by noise radius R. Noise radius is 16. Sample size is 8. Problem dimension is 5.

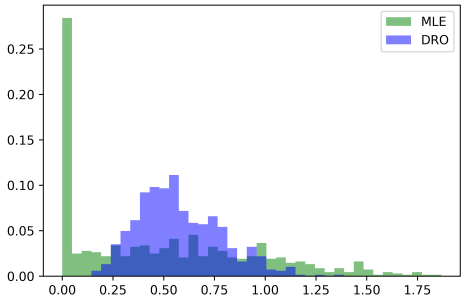


Figure 2.2: Comparison of DRO and MLE, synthetic data. All data are normalized by noise radius R. Noise radius is 16. Sample size is 8. Problem dimension is 5.

	Mean	Median	25th Perc.	10th Perc.
DRO	0.822	0.987	**-3.888**	**-9.758**
SAA	0.955	**1.032**	-3.896	-10.013
MLE	**1.433**	0.733	-6.032	-13.704
Independent DRO	0.819	0.988	-3.889	-9.891

Table 2.2: Comparison of statistics of daily return for real stock data.

2.5.2 Real Data

We perform our real data analysis with the asset price of nine tech companies from the S&P 500 (INTC, AMZN, FB, MSFT, GOOGL, IBM, ORCL, ADBE, AAPL) from January 2013 to January 2018 with our model on the log price $y_t = \log(p_i)$ at the end of each day and the objective function the approximated return $\sum_{i=1}^{d} x_i(\frac{p_{i+1}}{p_i} - 1) \approx \langle x, y_T - y_{T-1} \rangle$ (where one uses the approximation $e^r \approx 1 + r$, which is very accurate in the range of successive daily price ratios) [149]. We again compare the DRO and SAA models. In addition, we run the algorithm assuming independence between the samples (which we call "Independent DRO"). For each day, the algorithms are allowed to use data from the previous 15 days. The parameters are chosen by experimenting on the first three months of the dataset with, in reference to Theorem 2.3.2 and §2.4, $\delta = 0.05, 0.1$, and $R = 1\%, 4\%, 10\%$ and selecting the combination with the best accumulated return.

The results are shown in Table 2.2 for some quantiles of actual returns if we invest $10,000$ dollars each day, $10000 \sum_{i=1}^{d} x_i(\frac{p_{i+1}}{p_i} - 1)$, where x is, in turn, the solution for the four approaches. We see that both robust methods have a significantly lighter tail than does either the SAA or MLE approach, that our AR-based DRO on performs better than the independent DRO (except only slightly for the median), and that ignoring the uncertainty results in a significant degradation (MLE).

2.6 Conclusion

We present a distributionally robust formulation of a stochastic optimization problem for non-i.i.d vector autoregressive data. We use the Wasserstein distance to define robustness in the space of distributions. The resulting optimization problem is a finite concave-convex saddle point problem that can be solved efficiently. On a portfolio problem whose objective contains a linear term, we demonstrate that the approach results in lighter tails compared with an MLE or an SAA formulation for both synthetic and real data and that DRO problems ignoring sample dependence perform worse.

CHAPTER 3

OUTLIER-ROBUST, DATA-DRIVEN, DISTRIBUTIONALLY ROBUST OPTIMIZATION

3.1 Introduction

In this paper, we consider a robust framework for stochastic programming (SP) problem [157]:

$$\min_{a\in\mathcal{A}} H(a) = \mathbb{E}_{x\sim F}\left[h(a,x)\right]. \tag{3.1.1}$$

Here a is our decision to make within some decision set $\mathcal{A}$. This model has a lot applications in statistics [31], machine learning [119], and operations research [191].

In a lot of studies, the distribution F is assumed to be known [22]. However, in real-world applications, we do not often have the data generating process F or only have some knowledge that F belongs to the certain distribution family. Often case, we have data $x_1, x_2, \cdots, x_n$ $i.i.d$ sampled from F. One simple approach is to replace F by the empirical distribution $F_n = \sum_{i=1}^{n} \delta_{x_i}$. However, this simple approach may raise the issue of overfitting the data set [84].

To resolve the issue of overfitting, we consider a distributionally robust optimization (DRO) approach [41]. Whereas, given sample empirical distribution F_n we have certain characterization of the actual distribution F being relatively close to F_n with certain uncertainty quantification $F \in \mathcal{U}(F_n)$. In some cases the uncertainty set has the form $\mathcal{U}(F_n) = \{F | d(F, F_n) \leq \epsilon(n)\}$. Here d is some metric of distributions. Then DRO propose to solve the following minmax robust optimization programming problem,

$$\min_{a\in\mathcal{A}} \max_{F\in\mathcal{U}(F_n)} \mathbb{E}_{y\sim F}\left[h(a,y)\right]. \tag{3.1.2}$$

In other words, we are making our decision w.r.t. to the worst-case scenario within the distributional uncertainty set.

Some choices of specifying the uncertainty set $\mathcal{U}$ include. 1) moment characterization [41, 203], which suffers from lack of consistency. e.g., given infinite sample size, the uncertainty set doesn't converge to a singleton of true distribution $\{F\}$. This can be easily seen because finite moments (or even infinite moments) are not sufficient to characterize a distribution. 2) statistical divergence [95, 187, 167, 102, 16, 9], which suffers from the fact that the distributions within $\mathcal{U}(F_n)$ always has the same support as F_n, therefore $\mathcal{U}$ is not rich enough to capture the true distribution. 3) Kernel-based maximum mean discrepancy (MMD) [164, 201], which results in semi-infinite programming and by far doesn't have a finite-dimensional approximation with theoretical guarantees. We choose 4) Optimal transport/Wasserstein distance-based metric [66, 58], which resolves all of the concerns aforementioned, and in some settings, it yields a convex tractable optimization formulation [58].

In this work we consider the following regression based problem, which is more general and practical. In a lot of data-driven problems, we often encounter the following form of problem. Given some observation x which has predictive power for the upcoming y. We want to solving the following decision making problem,

$$\min_{a \in \mathcal{A}} H(a \mid x) = \mathbb{E}_{y \sim F_{y|x}} \left[h(a, y) \right]. \tag{3.1.3}$$

Here $F_{y|x}$ is the conditional distribution of y given x. Conditional distribution $F_{y|x}$ can be quite hard to estimate without any model assumption. Here, we pose a linear model with homogeneous noise assumption, where

$$y = f(x) + \xi = Ax + \xi. \tag{3.1.4}$$

We assume $f(x)$ can be parameterized by a linear model $f(x) = Ax$ and $\xi \sim G$ homogeneously (not depending on x) and independently. Then purpose a regression-dro framework to robustly solve the problem (3.1.3).

Another common problem in the real-world application is data contamination [2], where the majority of the data are actually from the underlying data generating process, but a few of the samples are not. A popular model in statistics literature for modeling this issue is Huber's contamination model [89, 91],

$$(1-\alpha)F + \alpha Q, \tag{3.1.5}$$

where a large portion $(1-\alpha)$ of data are sampled from F, while a small portion α of data are from arbitrary distribution, we call this small portion of data contamination. It is also worth mentioning in modern statistics and machine learning literature, we study a stronger contamination model, whereas the contamination can be adaptive to your algorithm design or previous data generated [198].

For the linear regression case with contamination concern, there are methods include M-estimator [91], and many other trimmed/weighted loss-based estimators, see, e.g., [159, 145, 146]. The properties of the minimizer are studied, but in practice, only Huber's loss can be computed efficiently and in scale. Recently, there has been a resurgence of research study on Huber's loss from a sparsity induced ℓ_1 penalty angle [172, 108, 158]. From this point of view, many generalizations using ℓ_0 based algorithm, e.g., Iterative Hard Thresholding [23], can be applied for outlier robust regression [19]. In our case, we still use Huber's loss because of its simple form and scalability. We also study Huber's loss's finite sample non-asymptotic performance and the performance under stronger adaptive contamination (not necessarily draw random sample totally independent of the actual data, so-called 'oblivious').

We further extend our study to time-dependent data, in which we study the data generating from vector autoregressive model (VAR) [80].

3.2 Problem Set-up

Suppose we have data pair $\{(x_i, y_i)\}_{i=1}^n$ from a linear model with contamination,

$$y_i = \beta^{*T} x_i + \xi_i + o_i. \tag{3.2.1}$$

Here ξ_i is the generic noise from observation. We assume that ξ_i are $i.i.d$ from distribution F with zero mean and supports on $\mathcal{B}$ with radius B_1. Here the outlier is o_i. We assume that the outliers are oblivious. Here 'oblivious' means that the choice of the corruption o is independent with x and ξ. We also assume sparse contamination namely, $\{i|o_i \neq 0\} = k = n\alpha$ with $\alpha < 1/8$. It is worth noting that there are regression algorithms and frameworks with stronger outlier models (for example, outlier models dependent on x and ξ). Our DRO-after-regression approach can be extended to those cases; however, for simplicity of exposition on this first endeavor, we only proceed with the oblivious model.

Our goal is to make robust decision w.r.t. to the upcoming y_{n+1} given observation x_{n+1}. To be specific, let a be our action variable, we aim to solve the following stochastic programming problem,

$$\min_a \mathbb{E}_{\xi \sim F}[h(a, \beta^{*T} x_{n+1} + \xi)]. \tag{3.2.2}$$

Since we do not have distribution F and parameter β^*, we can only estimate them from data. The nature of distributionally robust optimization (DRO) is to make decisions while taking the uncertainty of our estimation $\hat{F}$ and $\hat{\beta}$ into account. Namely, we have an ambiguity characterization of what we have learned from data, and normally, the set is a confidence interval,

$$(\beta, F) \in \mathcal{C}. \tag{3.2.3}$$

Here the ambiguity set $\mathcal{C}$ with high probability will include true (β^*, F). Then we optimize our decision based on the worst case scenario in the ambiguity set,

$$\min_{a} \max_{(\beta,F)\in\mathcal{C}} \mathbb{E}_{\xi\sim F}[h(a, \beta^{*T} x_{n+1} + \xi)]. \tag{3.2.4}$$

We further consider the problem in a Vector Autoregression (VAR) setting, where

$$\tilde{z}_{t+1} = A^* \tilde{z}_t + \xi_t,\ t = 0, 1, \cdots, n-1, \tag{3.2.5}$$

but we observe a contaminated sequence z_t, where $z_t = \tilde{z}_t + o_t$. In this case still still write $(x_i, y_i) = (z_i, z_{i+1})$ for $i = 0, 2, \cdots, n-1$. Similarly, we assume ξ_i are sampled $i.i.d$ from distribution F with zero mean supports on set $\mathcal{B}$ with radius B_1. We further assume the true time series is bounded $\|\tilde{z}_t\| \leq B_2$, we will justify that this is a direct results from mixing condition of time series and bounded support assumption on F. For the ease of presentation, here we assume $o_i = 0$ with at least $(1-\alpha)n$ number of data point. Also the choice of contaminated i is independent of time series $\tilde{z}$, but once index set $\{i \mid o_i \neq 0\}$ is chosen, with $|\{i \mid o_i \neq 0\}| = k = n\alpha$, the choice of o_i can be adaptive to data set $\tilde{z}$ and generic noise ξ. We note that this is already stronger than the 'oblivious' assumption. We also assume that the ratio of contaminated sample $\alpha < 1/8$.

3.3 Huber DRO Framework

3.3.1 A Two-step Computational Formulation

We first consider the regression Problem (3.2.1). The basic idea is to reduce this robust optimization to the type of problem studied in [47]. Note that, even though [47] is studying a problem in a time series setting, all of the techniques also apply to regression and robust decision marking based on prediction as we are considering here. The strategy in this work

with the un-contaminated model ($o = 0$) is as follows.

- 1. We have the estimation $\hat{\beta}$ obtained from regression with confidence region $\beta^* \in \{\|\beta - \hat{\beta}\| \leq \epsilon_1\}$.
- 2. Given any feasible β, we have empirical residual $\hat{\xi}_i = y_i - \beta^T x_i$.
- 3. Let $F_n(\beta) = \frac{1}{n}\sum_{i=1}^n \delta_{y_i - \beta^T x_i}$ be the empirical distribution of the residual. Then we have a confidence interval of actual distribution F characterized by $d(F, F_n(\beta^*)) \leq \epsilon_2$, with some distributional metric d.
- 4. By 1) and 3), we have a joint ambiguity set $\mathcal{C}$, which has the form $\|\beta - \hat{\beta}\| \leq \epsilon_1$ and $d(F(\beta), F)) \leq \epsilon_2$, and with high probability will include the actual (β^*, F).

Therefore, we have the following robust optimization.

$$\min_a \max_{\beta, F} \mathbb{E}_{\xi \sim F}\left[h(a, \beta^T x_{n+1} + \xi)\right] \tag{3.3.1}$$

$$\text{s.t. } \|\beta - \hat{\beta}\| \leq \epsilon_1 \tag{3.3.2}$$

$$d\left(F, F_n(\beta)\right) \leq \epsilon_2. \tag{3.3.3}$$

We will use Wasserstein distance as our distribution space metric $d(F, G)$. The p-Wasserstein distance can be defined as follows.

Definition 3.3.1. Let $\mathcal{X} \in \mathbb{R}^d$ with metric d, and F and G be two distributions support on $\mathcal{X}$, then the p-th power of p-Wasserstein distance ($p \geq 1$) is,

$$W_p{}^p(F, G) = \min_{\pi \in \Pi(F,G)} \int_\pi d^p(x_1, x_2)\, d\pi(x_1, x_2), \tag{3.3.4}$$

$$= \mathbb{E}_{(X,Y)}[d^p(X, Y)], \text{ where the marginal distribution of } X \text{ is } F \text{ and } Y \text{ is } G. \tag{3.3.5}$$

Here $\Pi(F, G)$ denotes the distribution on $\mathcal{X} \times \mathcal{X}$ with marginal F and G.

We also need the following convex property of W_p.

Proposition 3.3.2. *The p-th power of p-Wasserstein distance $W_p{}^p(F, G)$ is convex (jointly in F, G).*

Proof. For any $\epsilon > 0$. We have $\pi_1 \in \Pi(F_1, G_1)$ and $\pi_2 \in \Pi(F_2, G_2)$ such that, for $i = 1, 2$,

$$\mathbb{E}_{\pi_i}\left[d^p(x_1, x_2)\right] \leq W_p{}^p(F_i, G_i) + \epsilon. \tag{3.3.6}$$

Then for any $0 \leq \lambda \leq 1$, $\pi_3 = \lambda\pi_1 + (1-\lambda)\pi_2 \in \Pi(\lambda F_1 + (1-\lambda)F_2, \lambda G_1 + (1-\lambda)G_2)$. Then we have

$$W_p{}^p(\lambda F_1 + (1-\lambda)F_2, \lambda G_1 + (1-\lambda)G_2) \tag{3.3.7}$$

$$\leq \mathbb{E}_{\pi_3}\left[d^p(x_1, x_2)\right] \leq \lambda W_p{}^p(F_1, G_1) + (1-\lambda)W_p{}^p(F_2, G_2) + \epsilon. \tag{3.3.8}$$

Let $\epsilon \to 0$; we get the desired result. □

3.4 Independent Setting

3.4.1 Robust Regression

Under the oblivious contamination model, we have a convex formulation [172] for performing regression and outlier sample detection. Here we explain the idea for the formulation of the convex programming, and how it can be reduced to the well-known Huber's loss. It is known that the ℓ_1 penalty can encourage solution sparsity, and we assume that the outlier vector o is sparse. Therefore, we formulate the problem as a quadratic loss on the residual $y_i - \beta^T x_i - o_i$, along with a ℓ_1 penalty on the outlier vector o,

$$\min_{\beta, o} \sum_i (y_i - x_i^T\beta - o_i)^2 + \lambda\|o\|_1. \tag{3.4.1}$$

By solving the above regression problem, we can screen out the outlier by throwing away the sample corresponding to $i : o_i \neq 0$, and then use the distributionally regression framework [47]. If we partially solve (3.4.1), that is solve it first with respect to o with fixed β, then β is a solution for the following equivalent regression formulation with Huber's loss function with some r [65]

$$\min_{\beta} \sum_{i} h_r(y_i - x_i^T \beta). \tag{3.4.2}$$

Here the Huber's loss function is,

$$h_r(x) = \begin{cases} x^2/2 & \text{if } |x| \leq r, \\ r\left(|x| - r/2\right) & \text{o.w.} \end{cases}$$

To simplify our presentation, we assume design covariate x_i are sampled *i.i.d.* from a distribution with zero mean, identity covariance matrix with bounded radius B_2, though our approach can be easily extended to the general subGaussian case (with subGaussian noise). Then we derive the characterization of true parameter $\|\beta^* - \hat{\beta}\| \leq \epsilon_1$ as in (3.3.1).

Theorem 3.4.1. *Let $\hat{\beta}$ denote the minimizer of the convex program* (3.4.2) *with $r = 2B_1$, with the assumptions of design distribution on x_i and noise distribution ξ_i mentioned before. When n is large enough, with probability geater than $1 - \delta$, we have the following,*

$$\|\hat{\beta} - \beta^*\| \leq \left(16 C_0 B_1 B_2 \sqrt{\log(1/\delta)}\right) / \sqrt{n} := K_0 B_1 B_2 / \sqrt{n}.$$

Here K_0 is some constant independent of n.

Proof. For the robust regression problem,

$$\min_{\beta} \; L(\beta) = \sum_{i=1}^{n} h_r(y_i - x_i^T \beta). \tag{3.4.3}$$

The assumptions guarantee that $\|\xi_i\| \le B_1$, $\|x_i\| \le B_2$. Let S_{good} denote the set of indexes corresponding to uncontaminated samples, and S_{bad} vice versa. Let β^* be the true parameter and $\hat{\beta}$ be the **local** minimizer of the above function within radius around β^*,

$$R := (r - B_1)/B_2 = B_1/B_2, \tag{3.4.4}$$

that is, $\hat{\beta} := \arg\min_{\|\beta-\beta^*\|\le R} L(\beta)$. *Note that $\|\beta - \beta^*\| \le R$ guarantees that all of the good samples $y = \beta^T x + \xi$ are falling in the quadratic (smooth) part of the Huber's loss function.* We will later verify that, with high probability, the $\hat{\beta}$ is also the **global** minimizer of $L(\beta)$. Note that to justify this, we only need to show that $\hat{\beta}$ lies in the interior of the ball (3.4.4) since this is a convex program. By the convexity of Huber's loss function, we have

$$L(\hat{\beta}) \le L(\beta^*) \text{ , by the optimality of } \hat{\beta}, \tag{3.4.5}$$

$$L(\beta^*) \ge L(\hat{\beta}) \ge L(\beta^*) + \langle \nabla L(\beta^*), \hat{\beta} - \beta^* \rangle + \sum_{i \in S_{good}} \frac{1}{2}(\hat{\beta} - \beta^*)^T H(h_{r,i})(\hat{\beta} - \beta^*) \tag{3.4.6}$$

$$\ge L(\beta^*) - \|\nabla L(\beta^*)\|\|\hat{\beta} - \beta^*\| + \frac{\lambda_{\min}(X_{good}X_{good}^T)}{2}\|\hat{\beta} - \beta^*\|^2. \tag{3.4.7}$$

Here $H(f)$ denotes the Hession matrix of function f w.r.t. β, also $h_{r,i} := h(y_i - \beta^T x_i)$. Inequality (3.4.6) follows from the fact that the tangent approximation to $h_{r,i}$ is an underestimator due to convexity for all i, and since for $i \in S_{good}$, $h_{r,i}(\beta)$ is a quadratic function for $\|\beta - \beta^*\| \le R$. In the latter case $H(h_{r,i}) = x_i x_i^T$. We use λ_{min} as a abbreviation of $\lambda_{min}\left(X_{good}X_{good}^T\right)$. From (3.4.6) and (3.4.7), subtracting $L(\beta^*)$ from all sides we get

$$-\|\nabla L(\beta^*)\|\|\hat{\beta} - \beta^*\| + \frac{\lambda_{min}}{2}\|\hat{\beta} - \beta^*\|^2 \le 0 \Longrightarrow \|\beta - \beta^*\| \le \frac{2\|\nabla L(\beta^*)\|}{\lambda_{min}}. \tag{3.4.8}$$

Next, we derive a upper bound of $\|\nabla L\|$ and a lower bound for λ_{min}. The upper bound is based on concentration of *good* sample and boundness of the derivative of the Huber's loss on contaminated samples,

$$\|\nabla L(\beta^*)\| = \left\| \sum_{i \in S_{good}} \xi_i x_i + \sum_{i \in S_{bad}} h_r'(\xi_i + o_i) x_i \right\|. \tag{3.4.9}$$

Notice that by the oblivious condition, ξ_i, o_i are independent of x_i. Note that $\|\xi_i\| \leq B_1 \leq r$ and $\|h_r'\| \leq r$. This implies that all random variables appearing to the right of (3.4.6) are bounded and each term is less than rB_2. Then by Hoeffding's inequality for the bounded vector [133, Theorem 3.5] we have, with probability greater than $1 - \delta$,

$$\|\nabla L(\beta^*)\| \leq C_0 r B_2 \sqrt{n \log(1/\delta)}. \tag{3.4.10}$$

Then from (3.4.8) we have

$$\|\beta - \beta^*\| \leq \frac{2C_0 r B_2 \sqrt{n \log(1/\delta)}}{\lambda_{min}}. \tag{3.4.11}$$

It remains to show the lower bound of λ_{min} is scale as $\Omega(n)$. Let $|S_{good}| = m$. By [182, Remark 5.40], we have with probability greater than $1 - \delta$, we have

$$\frac{1}{m}\lambda_{min} \geq 1 - CB_2^2 \max\left(\sqrt{\left(d + \log(1/\delta)\right)/m}, \left(d + \log(1/\delta)\right)/m \right). \tag{3.4.12}$$

We know that $m = n - k = (1 - \alpha)n$ (see §3.2 for definitions), and when n large enough, we have $\lambda_{min} \geq n/2$. Then for large enough n, such that $\left(16 C_0 r B_2 \sqrt{\log(1/\delta)}\right)/\sqrt{n} < B_1/B_2$, then (3.4.4) and (3.4.11) hold simultaneously, which verifies that the constrained minimizer $\hat{\beta}$ actually lies strictly in the interior of the ball (3.4.4). Since $\hat{\beta}$ is a strict local minimizer

of a convex program, it is also the global minimizer. □

Remark 3.4.2. From the proof, we know that the global minimizer $\hat{\beta}$ satisfies $\|\beta^* - \hat{\beta}\| \leq R$ with probability greater than $1 - 2\delta$. When that occurs, the proof states that all of the good samples will fall in the quadratic part of $h_{2B_1}(y_i - \hat{\beta}x_i)$. Moreover, for any (x_i, y_i) that falls in the quadratic part we know $|y_i - \beta^{*T}x_i| = |\xi_i + o_i| \leq |y_i - \hat{\beta}^T x_i| + |(\beta^* - \hat{\beta})^T x_i| \leq r + B_2 R = 2B_1 + B_2 \cdot \frac{B_1}{B_2} = 3B_1$, where the second term bound comes from (3.4.4).

3.4.2 DRO Formulation

Once we finish the robust regression, we can eliminated the samples fail outside the Huber's loss quadratic part. Keep the remaining m samples and let $F(\beta) = \frac{1}{m}\sum_{i=1}^{m} \delta_{y_i - \beta^T x_i}$ and perform the distributionally robust optimization:

$$\begin{aligned} V_{dro} := \min_{a} \max_{F,\beta} \mathbb{E}_F \left[h(a, \beta^T x_{n+1} + \xi_i) \right] \\ \text{s.t. } \|\beta - \hat{\beta}\| \leq \epsilon_1 \\ W_p^p(F, F(\beta)) \leq \epsilon_2. \end{aligned} \tag{3.4.13}$$

This is a difficult problem to solve due to the infinite dimensionality of the Wasserstein constraint. We will try instead to solve the problem.

$$
\begin{aligned}
V_{dro_n} := \min_{a} \max_{\beta, \{\xi_i'\}_{i=1}^m} & \sum_{i=1}^m h(a, \beta^T x_{n+1} + \xi_i') \\
\text{s.t. } & \|\beta - \hat{\beta}\| \le \epsilon_1 \\
& \frac{1}{m} \sum_{i=1}^m \|\xi_i - \xi_i'\|^p \le \epsilon_2 \\
& \xi_i' \in \mathcal{B} \\
& \xi_i = y_i - \beta^T x_i.
\end{aligned} \tag{3.4.14}
$$

Proposition 3.4.3. *When h is convex in the first argument and concave in the second argument, formulation* (3.4.13) *is equivalent to formulation* (3.4.14)*, and the latter is tractable.*

Proof. Follows from [47, Theorem 3.2]. □

For general h the convex-concave reduction no longer exists. However, we always have a finite dimensional reformulation of the problem.

$$
\min_{a} \max_{\|\beta - \hat{\beta}\| \le \epsilon_1} \min_{\lambda \ge 0} m\lambda\epsilon_2 + \sum_{i=1}^m \max_{\xi_i' \in \mathcal{B}} \left[h(a, \beta, \xi_i') - \lambda \|\xi_i - \xi_i'\|^p \right]. \tag{3.4.15}
$$

Solving this multiple min/max/min/max formulation is still very hard, and we will therefore aim to solve (3.4.14) anyways. Note that formulation (3.4.14) can be seen as replacing the distribution F by another $\tilde{F}_m$ with m support with $W_p^p(F(\beta), F_m) \le \epsilon_2$ by [132, Proposition 2.1] (actually here are some subtlety, namely the best π coupling distribution has to be a one-to-one matching, but it is known that the best Wasserstein distance coupling for n-to-n empirical distribution is precisely the perfect matching problem presenting here by an argument that the doubly stochastic matrix is the convex hull of permutation matrix).

We thus propose solving (3.4.14) as a finite approximation of the infinite dimensional distributional problem. Actually, we have a finite approximation guarantee for this approach

from [66, Corollary 2 iv)]. With our compact support assumption, we have the following results from [66].

Proposition 3.4.4. (*Corollary 2 from [66]) For the he DRO problem, suppose we have continuous function* $|g(\xi)| \leq B$, *and* $P_n = \frac{1}{n}\sum_{i=1}^n \delta_{\xi_i}$ *supports on compact region* $\mathcal{B}$, *the infinite dimensional problem,*

$$\begin{aligned} dro : \max_{F} \ & \mathbb{E}_{\xi\sim F}\left[g(\xi)\right] \\ s.t. \ & W_p^p(P_n, F) \leq \epsilon \\ & F \text{ supports on } \mathcal{B}, \end{aligned} \tag{3.4.16}$$

can be approximated by the finite dimensional programming

$$\begin{aligned} dro_n : \max_{\{\xi_i'\}_{i=1}^n} \ & \frac{1}{n}\sum_{i=1}^n g(\xi_i') \\ s.t. \ & \frac{1}{n}\sum_{i=1}^n \|\xi_i - \xi_i'\|^p \leq \epsilon \\ & \xi_i' \in \mathcal{B}. \end{aligned} \tag{3.4.17}$$

with guarantee that $dro_n \leq dro \leq dro_n + 2B/n$.

The proof is almost verbatim of the proof of [66, Corollary 2 iv)], which relies on Corollary 2 iii), which relies on the existence of the worst-case distribution in the Wasserstein uncertainty set. We only need to justify the existence of the worst-case distribution. First note that $W_p(\mathcal{B}) = \{\mu \mid \mu \text{ supports on } \mathcal{B}\}$ is a compact metric space [130, Corollary 2.2.5], so is the closed ball $W_p(P_n, F) \leq \epsilon^{1/p}$, then it is sequentially compact, therefore, the worst-case distribution always exists.

With Proposition 3.4.4, we can now quantify the performance guarantee of the finite approximation (3.4.14).

Theorem 3.4.5. *Assume our target function $h(a, y)$ is bounded by B and continuous, let*

$$\begin{aligned} \Psi(a,\beta) &:= \max_F \mathbb{E}_F\left[h(a, \beta^T x_{n+1} + \xi_i)\right] \\ &\text{s.t. } W_p^p(F, F(\beta)) \le \epsilon_2 \end{aligned} \tag{3.4.18}$$

$$\begin{aligned} \Psi_n(a,\beta) := \max_{\{\xi_i' \in \mathcal{B},\, \xi_i = y_i - \beta^T x_i\}_{i=1}^m} & \sum_{i=1}^m h(a, \beta^T x_{n+1} + \xi_i') \\ & \frac{1}{m}\sum_{i=1}^m \|\xi_i - \xi_i'\|^p \le \epsilon_2 \end{aligned} \tag{3.4.19}$$

Let $\Phi(a) := \max_{\beta, \|\beta - \hat{\beta}\| \le \epsilon_1} \Psi(a, \beta)$ *and* $\Phi_n(a) := \max_{\beta, \|\beta - \hat{\beta}\| \le \epsilon_1} \Psi_n(a, \beta)$. *Finally, let* $a^* \in \operatorname{argmin}_a \Phi(a)$ *and* $a_n^* \in \operatorname{argmin}_a \Phi_n(a)$. *Then we have*

$$\Phi(a_n^*) \ge \Phi(a^*) \ge \Phi_n(a^*) \ge \Phi_n(a_n^*) \ge \Phi(a_n^*) - \frac{2B}{n} \ge \Phi(a^*) - \frac{2B}{n}. \tag{3.4.20}$$

Proof. By Proposition 3.4.4, we know that

$$\Psi(a,\beta) \ge \Psi_n(a,\beta) \ge \Psi(a,\beta) - \frac{2B}{n}. \tag{3.4.21}$$

Let $b(a) \in \arg\max_\beta \Psi(a, \beta)$ and $b_n(a) \in \arg\max_\beta \Psi_n(a, \beta)$ (we use $\in$ for arg max since there may be multiple solution to the respective optimization problems; the statements we obtain here will occur for any selection of the solution). We have that

$$\Psi(a, b(a)) \ge \Psi(a, b_n(a)) \ge \Psi_n(a, b_n(a)) \ge \Psi_n(a, b(a)) \tag{3.4.22}$$

$$\overset{(3.4.21)}{\ge} \Psi(a, b(a)) - \frac{2B}{n} \ge \Psi(a, b_n(a)) - \frac{2B}{n}. \tag{3.4.23}$$

By definition we have $\Phi(a) = \Psi(a, b(a))$ and $\Phi_n(a) = \Psi(a, b_n(a))$. Then we have from

(3.4.22) that

$$\Phi(a) \geq \Phi_n(a) \geq \Phi(a) - \frac{2B}{n}. \tag{3.4.24}$$

From the definition of a^* and a_n^*, we have that

$$\Phi(a_n^*) \geq \Phi(a^*) \geq \Phi_n(a^*) \geq \Phi_n(a_n^*) \geq \Phi(a_n^*) - \frac{2B}{n} \geq \Phi(a^*) - \frac{2B}{n}. \tag{3.4.25}$$

which completes the proof □

Remark 3.4.6. Since the value of the (3.4.13) is $\Phi(a^*)$ and the value of the (3.4.14) is $\Phi_n(a_n^*)$, this theorem guarantees that the performance of the solution of the finite approximation problem (3.4.14) a_n^* will not be degrade too much from the solution of the possibly continuous formulation (3.4.13) a^*. In particular, we have that $\Phi(a^*) \leq \Phi(a_n^*) \leq \Phi(a^*) + 2B/n$.

ϵ specification

Let the index set remaining after the Huber filtering be S_{trust}. Note that from the proof, we know that with probability greater than $1 - 2\delta$, all of the good samples will be included in S_{trust}, also we have that $\|\hat{\beta} - \beta^*\| \leq \frac{B_1}{B_2}$. We assume this event is true for the remaining of the section. Also, for any contaminated $i \in S_{trust}$, we know that $\|\xi_i + o_i\| \leq 3B_1$ from Remark 3.4.2.

Here we discuss the selection of the parameters. The choice of ϵ_1 is based upon Theorem 3.4.1. Note that $\alpha = k/n$ be the bad sample portion, and there are at most $\alpha' \leq \alpha$ portion of bad samples within the m kept samples (we keep all the good samples). Therefore we can write $F(\beta) = (1 - \alpha')F_{good}(\beta) + \alpha' F_{bad}(\beta)$. We have that $W_p^p(F(\beta^*), F) \leq (1 - \alpha')W_p^p(F_{good}(\beta^*), F) + \alpha' W_p^p(F_{bad}(\beta^*), F)$, by the convexity properties of W_p^p Proposition 3.3.2. For the first term as a convergence result of sample empirical distribution converges to population distribution, we apply [61, Theorem 2], which gives

$CB_1^p n_{good}^{-p/d}$ for $n_{good} = (1-\alpha)n$ large enough. For the second term we have upper bound $\alpha(4B_1)^p$ which can be easily seen from the Definition 3.3.1, using the bound of the support for $F_{bad}(\beta^*)$ in Remark 3.4.2 and F is supports on set with radius B_1. Then we can select $\epsilon_2 = CB_1^p((1-\alpha)n)^{-p/d} + \alpha(4B_1)^p$.

Remark 3.4.7. *The need for a decontamination approach* Observe that any model, particularly least-squares can give arbitrary bad estimates under the contamination model [46], e.g. $\|\beta - \beta^*\| \geq C$ for arbitrary large C. As an example, suppose for an uncontaminated model that x_i are taken from $Unif(\pm 1)$, and $y = \beta^* \cdot x_i$. Then least square estimation is the sample average $\hat{\beta} = Mean(x_i \cdot y_i)$. If we contaminate the model and perturb one data point by adding $o_i = n \cdot C$ (a contamination is made independent of the dataset), where C extremely large, then we will clearly have $\|\hat{\beta} - \beta^*\| \geq \frac{|C|}{2}$. As C is arbitrary, a contamination model can alter the estimates arbitrarily. Therefore an outlier resistant approach is needed to prevent such an occurrence.

The unavoidability of a bias term

Observe that our choice of ϵ_2 includes a bias term, $\alpha(4B_1)^p$ that does not go to zero for increasing n if the contamination probability is fixed. That means that, while our decontamination approach prevents the estimate of β being arbitrarily far away, it cannot prevent the estimation of F from being biased, even in the limit of n large. For completeness, in this section, we explain, using results from other references, why there will be an intrinsic $O\left(\alpha B_1^2\right)$ term that cannot be fundamentally improved. Our setting in this section is slightly different in the sense that 1) we consider a subGaussian variable with variance proxy B_1 variable rather than bounded noise 2) we consider Huber's contamination model, and 3) we use the 2-Wasserstein distance. Our argument relies on the following facts.

Proposition 3.4.8. *There exists $c_1 > 0$, such that, for any two Gaussian distributions*

$P_1 = N(0, B_1^2)$ and $P_2 = N(0, (1+\delta)^2 B_1^2)$ with $0 \leq \delta \leq 1/2$, we have that

$$TV(P_1, P_2) \leq c_1 \delta. \tag{3.4.26}$$

Proof. By Pinsker's inequality, [175, 2.20]

$$TV(P_1, P_2) = TV(P2, P1) \overset{[175,\ 2.20]}{\leq} \sqrt{\frac{1}{2} KL(P_2 \| P_1)} = \sqrt{\frac{1}{2}\left((1+\delta)^2 - 1 - 2\log(1+\delta)\right)}.$$

The conclusion follows from the series expansion of $\log(1+\delta)$ at $\delta = 0$ whose convergence radius is 1 and whose first three terms are $\delta - \frac{1}{2}\delta^2 + \frac{1}{3}\delta^3$. □

Proposition 3.4.9. *For two Gaussian distributions $P_1 = N(0, B_1^2)$ and $P_2 = N(0, (1+\delta)^2 B_1^2)$ with $0 \leq \delta \leq 1/2$, the 2-Wasserstein distance between them satisfies*

$$W_2^2(P_1, P_2) = (\delta B_1)^2. \tag{3.4.27}$$

Proof. From *[76, Proposition 7]* we have that

$$W_2^2(P_1, P_2) = B_1^2 + (1+\delta)^2 B_1^2 - 2B_1^2(1+\delta) = B_1^2(1+\delta-1)^2 = B_1^2\delta^2$$

which proves the claim. □

Proposition 3.4.10. *For two distributions P_1 and P_2, if $TV(P_1, P_2) \leq \alpha/(1-\alpha)$, then there exists a pair of contamination distributions Q_1, Q_2, such that*

$$(1-\alpha)P_1 + \alpha Q_1 = (1-\alpha)P_2 + \alpha Q_2 = U. \tag{3.4.28}$$

Then given any data $X_1, \cdots, X_n$ from the contaminated distribution, for any test function

defined on the n-copy of the output space $\phi : Y^n \to \{1, 2\}$*, we have that*

$$U(\phi(X_1, X_2, \ldots, X_n) = 1) + U(\phi(X_1, X_2, \ldots, X_n) = 2) = 1. \tag{3.4.29}$$

Therefore, we have

$$\max_{i=1,2} U(\phi = i) \geq 1/2. \tag{3.4.30}$$

Proof. Let $C = (1 - \alpha) \cdot TV(P_1, P_2) \leq \alpha$, then take $Q_1 = \frac{1}{\alpha}(P_2 - P_1)_+ + \frac{\alpha - C}{\alpha}\delta_0$, $Q_2 = \frac{1}{\alpha}(P_1 - P_2)_+ + \frac{\alpha - C}{\alpha}\delta_0$. Here δ_0 is the distribution that puts all the mass at zero. □

The significance of Proposition 3.4.10 is the following. Under our contamination model, if U is the contaminated distribution then either P_1 or P_2 could have equally well produced the data. Therefore if our relaxation were consistent, it must allow for *both* P_1 and P_2 to be in the feasible set with high probability. We will now show that, if we choose ϵ_2 too aggressively (too small) at least one of these two distributions will be excluded by the Wasserstein distance constraint $W_2^2(F, F(\beta)) \leq \epsilon_2$ in (3.4.13) with high probability.

Let $\delta = \frac{\alpha}{c_1(1-\alpha)}$. Let now $P_1 = N(0, B_1^2)$ and $P_2 = N(0, (1 + \delta)^2 B_1^2)$ be two normal distributions. From Proposition 3.4.8 it follows that $TV(P_1, P_2) \leq c_1\delta = \frac{\alpha}{1-\alpha}$. We can subsequently apply Proposition 3.4.10 to identify contaminating distributions Q_1 and Q_2 such that $U = (1-\alpha)P_1 + \alpha Q_1 = (1-\alpha)P_2 + \alpha Q_2$. *Assume now*, that there is an algorithm that, given any sample from contaminated distribution $X_1, \cdots, X_n \sim U$, can output a distribution $F := F(X_1, X_2, \ldots, X_n)$ such that $W_2(F, P_i) < \delta B_1/2 = \frac{\alpha}{1-\alpha} B_1/2c_1$, $i = 1, 2$ with probability strictly greater than $0.5 + \varepsilon > 0.5$ (better than random guess). We define

$$\phi(X_1, X_2, \ldots, X_n) = \begin{cases} 1 & \text{if } W_2(F(X_1, X_2, \ldots, X_n), P_1) \leq W_2(F(X_1, X_2, \ldots, X_n), P_2) \\ 2 & \text{o.w.} \end{cases} \tag{3.4.31}$$

Then from (3.4.30), we know,

$$\max\left([(1-\alpha)P_1+\alpha Q_1]\left(\phi=2\right),\ \ [(1-\alpha)P_2+\alpha Q_2]\left(\phi=1\right)\right)\geq 1/2,\Rightarrow \tag{3.4.32}$$

$$\max\Bigg(\ [(1-\alpha)P_1+\alpha Q_1]\left(W_2(F,P_1)\geq\frac{\alpha}{1-\alpha}B_1/2c_1\right), \tag{3.4.33}$$

$$[(1-\alpha)P_2+\alpha Q_2]\left(W_2(F,P_2)\geq\frac{\alpha}{1-\alpha}B_1/2c_1\right)\Bigg)\geq 1/2, \tag{3.4.34}$$

The statement (3.4.33) follows from a triangle inequality and Proposition 3.4.9 . Indeed, observe that

$$\phi=2\Rightarrow W_2(F,P_2)\leq W_2(F,P_1) \tag{3.4.35}$$

$$\Rightarrow 2W_2(F,P_1)\geq W_2(F,P_2)+W_2(F,P_1)\overset{\text{triangle}}{\geq}W_2(P_1,P_2)\overset{\text{Proposition 3.4.9}}{=}\delta B_1. \tag{3.4.36}$$

Therefore, we know

$$\phi=2\Rightarrow W_2(F,P_1)\geq\delta B_1/2. \tag{3.4.37}$$

Therefore, when presented with an α-contaminated distribution U, we cannot find a screening procedure that produces an empirical density approximation F from the samples $X_1,X_2,\ldots,X_n$, that satisfies $W_2\left(P,F\right)\leq\frac{\alpha}{1-\alpha}B_1/2c_1$ *for all* true distributions P that may have produced U. Applying this rationale to our formulation (3.4.14) for our screening residual-based distribution P_n, it follows that we cannot satisfy $W_2\left(P_n,F\right)=\frac{1}{m}\sum_{i=1}^{m}\|\xi_i-\xi_i'\|^p\leq\frac{\alpha}{1-\alpha}B_1/2c_1$ for *some potentially true distribution* F with probability greater than 0.5. Therefore, for our formulation (3.4.14) to contain the true distribution with high enough probability the parameter ϵ_2 has to be greater than $\frac{\alpha}{1-\alpha}B_1/2c_1$. This "bias" needs to be large enough, and, in particular, cannot go to zero as $n\to\infty$ as we expressed in the beginning

of §3.4.2.

3.5 Vector Autoregression

In this section, we study the problem within a time series setting. For notational simplicity, in this section we assume,

$$B_2 = 1 \Rightarrow \|\tilde{z}_t\| \leq 1 \quad \forall t = 0, 1, \ldots, T. \tag{3.5.1}$$

Note that once we assume the true sequence is bounded, we can rescale it to make sure (3.5.1) holds, so this assumption results in no loss of generality. Remember from §3.2 that we denote $(x_i, y_i) = (z_i, z_{i+1})$. This time, we study vectorized robust regression problem.

$$\min_A L(A) = \sum_{i \in S_{trust}} h_r(\|y_i - Ax_i\|). \tag{3.5.2}$$

Function h_r is monotone and convex. Therefore the above program is convex as well. We will specify the trust set S_{trust} later.

3.5.1 Robust Regression

In this section, we analyze Huber's regression with time series. A by-product of this section is that we analyze Huber's loss performance under an adaptive type of contamination and the bias induced by the contamination. We still denote the data pair (x_i, y_i) as good, if both x_i and y_i are not contaminated. Let $S_{con} \in \{0, 1, \cdots, n-1\} = \{i_1, i_2, \cdots, i_k\}$ be the set of index of contaminated z_i's with $k = n\alpha$. Let $S_{bad} = \{i \mid 0 \leq i \leq n-1, i \in S_{con} \text{ or } i+1 \in S_{con}\}$. Since S_{con} is independent of $\tilde{z}_i, \xi_i$, and S_{bad} is built using S_{con}, then S_{bad} is independent of $\tilde{z}_i, \xi_i$. Then $S_{good} = \{0, 1, \cdots, n-1\}/S_{bad}$, indicating all of the index i with both $x_i = z_i$ and $y_i = z_{i+1}$ uncontaminated, is still independent of $\tilde{z}_i, \xi_i$. Given that

$|S_{con}| \leq 2k = 2\alpha n$, we know that $|S_{good}| \geq (1-2\alpha)n$. We first throw out the pair (x_i, y_i) with $\|x_i\| > B_2 = 1$. We call the data set index after this screening as S_{trust}.

Using Remark 3.4.2 we have that with probability at least $1-2\delta$ the screening does not eliminates a good data pair. Therefore, $S_{good} \in S_{trust}$ with probability at least $1-2\delta$. Then perform the Huber regression on the remaining data (3.5.2).

We discuss how we adapt the proof to the time series setting. Our proof for Theorem 3.4.1 is relying on a) upper bound of $\left\|\sum_{i\in good}\xi_i x_i^T + \sum_{i\in trust/good}\nabla h_r(\|\xi_i + o_i\|)x_i^T\right\|_F$ and b) lower bound of the minimal eigenvalue $\lambda_{min}(X_{good}X_{good}^T)$, which gives us,

$$\|\hat{A} - A^*\|_F \leq \frac{2\left\|\sum_{i\in good}\xi_i x_i^T + \sum_{i\in trust/good}\nabla h_r(\|\xi_i + o_i\|)x_i^T\right\|_F}{\lambda_{min}(X_{good}X_{good}^T)}. \tag{3.5.3}$$

Here $\nabla h_r(\cdot)$ denotes the gradient of $h_r(\|x\|)$ as a function of x and

$$\nabla_x h_r(\|x\|) = \begin{cases} x & \|x\| \leq r \\ r\frac{x}{\|x\|} & \text{o.w.} \end{cases} \tag{3.5.4}$$

We know that $\|\nabla h_r(\cdot)\| \leq r$.

Lemma 3.5.1. *For the regression problem, with probability greater that $1-2\delta$, we have*

$$\left\|\sum_{i\in good}\xi_i x_i^T + \sum_{i\in trust/good}\nabla h_r(\|\xi_i + o_i\|)x_i^T\right\|_F \leq 2B_1\sqrt{2n\log(2/\delta)} + 2\alpha n r. \tag{3.5.5}$$

Proof. To proof this, let I_i denotes the indicator of the event that the index i and $i+1$ are not in the index set S_{con}, i.e. $I_i = I_{i\notin S_{con}} \cdot I_{i+1\notin S_{con}}$. Note that we call a data pair $(x_i, y_i) = (z_i, z_{i+1})$ good if both samples are not contaminated. By the assumption of random contaminated, S_{con} is independent of $\tilde{z}_i$ and ξ_i, and then all of I_i are independent of data $\tilde{z}_i$ and ξ_i. Then we define the filtration, $\mathcal{F}_{-2}$ is the trivial σ-field, and $\mathcal{F}_{-1} = \sigma(\{I_i\}_{i=0}^{n-1})$ and $\mathcal{F}_t$ to be the σ-field containing $\mathcal{F}_{t-1}$ and the σ-field $\sigma(\tilde{z}_t, \xi_{t-1})$. Then we have (note that

$x_i = z_i$ and $x_i I_i = z_i I_i = \tilde{z}_i I_i$),

$$\sum_{i \in good} \xi_i x_i^T = \sum_{i=0}^{n-1} \xi_i \tilde{z}_i^T I_i := 0 + 0 + \sum_{i=0}^{n-1} m_i. \tag{3.5.6}$$

It remains to show that m_i is a martingale difference sequence, i.e.

$$\mathbb{E}[m_i \mid \mathcal{F}_i] = 0, \tag{3.5.7}$$

$$m_i \in \mathcal{F}_{i+1}. \tag{3.5.8}$$

The second condition is by construction of $\mathcal{F}_t$. To see the first condition, we note that $I_i \in \mathcal{F}_i$ (see the definition of $\mathcal{F}_{-1}$), and from its independence from $\tilde{z}_i$ and ξ_i we get

$$\mathbb{E}[m_i \mid \mathcal{F}_i] = I_i \cdot \mathbb{E}[\xi_i \tilde{z}_i \mid \mathcal{F}_i], \tag{3.5.9}$$

$$\mathbb{E}[m_i \mid \mathcal{F}_i] = \begin{cases} 0 & I_i = 0 \\ \mathbb{E}[\xi_i \tilde{z}_i \mid \mathcal{F}_i] & I_i = 1. \end{cases} \tag{3.5.10}$$

Note that $\xi_i \perp \mathcal{F}_i$, $\tilde{z}_i \in \mathcal{F}_i$,

$$\mathbb{E}[\xi_i \tilde{z}_{i-1} \mid \mathcal{F}_i] = \mathbb{E}[\xi_i] \tilde{z}_{i-1} = 0 \cdot \tilde{z}_{i-1} = 0. \tag{3.5.11}$$

Then martingale difference condition (3.5.7) is justified. It is obvious that, with $B_2 = 1$, we have,

$$\|m_i\|_F = \|\xi_i \tilde{z}_{i-1} I_i\|_F \leq \|\xi_i\| \|\tilde{z}_{i-1}\| \ \leq B_1. \tag{3.5.12}$$

Then we have, by the bounded vector martingale concentration [133, Theorem 3.5], with

probability greater than $1-\delta$,

$$\left\| \sum_{i \in good} \xi_i x_i^T \right\|_F \leq 2B_1 \sqrt{2n \log(2/\delta)}, \tag{3.5.13}$$

and a simple bound $\left\| \nabla h_r(\|\xi_i + o_i\|) {x_i}^T \right\|_F \leq r$ on the *trust/good* part and the fact that $|S_{trust}/S_{good}| \leq 2\alpha n$ gives the desired result. □

For deriving the lower bound of the minimal eigenvalue, we provide a simplified proof based on [197] relying on mixing conditions. We make the following two assumptions on the time series. The first one guarantees that the time series is mixing to the stationary distribution, the second one ensures that the stationary distribution is non-degenerate.

Assumption 3.5.2. ***(Mixing assumption)*** *We assume there exits* Γ *and* $\gamma < 1$ *such that,*

$$\|A^{*k}\| \leq \Gamma \gamma^k.$$

This is a direct result of assuming the spectral radius $\rho(A^*) < 1$, whereas the spectral radius is the largest absolute value of eigenvalues of A^*. For more details, we refer to [200]. With the assumptions above, we know the time series is stationary. Let $\mathcal{D}$ denote the stationary distribution, i.e., the distribution of $\sum_{i=0}^{\infty} A^{*i} \xi_i$.

Remark 3.5.3. From our assumption about F, we know that $\|\xi_i\| \leq B_1$. With the mixing assumption above, assuming that z_{-1} is drawn from the stationary distribution, we know that $\|\tilde{z}_t\| \leq B_2 \leq \frac{\Gamma B_1}{1-\gamma}$. Therefore boundedness of the time series follows from the mixing assumption.

Next, we make the following non-degenerate assumption on the stationary distribution.

Assumption 3.5.4. ***(Boundedness assumption)*** *Let* Σ *be the covariance matrix of the*

stationary distribution $\mathcal{D}$, i.e. the solution of

$$A^{*T}\Sigma A^* - \Sigma + cov(\xi) = 0.$$

We assume that,

$$\Sigma \geq \ell I := \frac{I}{\kappa}.$$

Here we use the notation κ, because effectively, it is a proxy of the condition number of the covariance matrix (due to the upper boundedness assumption $B_2 = 1$ (3.5.1)). Next we couple a subsequence $(\tilde{z}_0, \tilde{z}_l, \cdots, \tilde{z}_{ml})$ to a *i.i.d* sample sequence from stationary distribution. We show that they are entry-wise exponentially close w.r.t l.

Lemma 3.5.5. *Suppose $\tilde{z} = (\tilde{z}_0, \tilde{z}_l, \tilde{z}_{2l}, \cdots, \tilde{z}_{ml})$ is a subsequence from the uncontaminated vector autoregression process with $\tilde{z}_0 \sim \mathcal{D}$. Under the assumptions of mixing 3.5.2 and boundedness 3.5.4, there exist the vectors of random variables $z' = (z'_0, z'_l, \cdots, z'_{ml})$, with each component an i.i.d. sample from the stationary distribution $\mathcal{D}$, and $u = (0, u_l, \cdots, u_{ml})$ such that*

$$z'_0 = \tilde{z}_0, \tag{3.5.14}$$

$$(\tilde{z}_0, \tilde{z}_l, \tilde{z}_{2l}, \cdots, \tilde{z}_{ml}) \overset{(d)}{=} z' + u, \text{ and} \tag{3.5.15}$$

$$\|u_{il}\| \leq \frac{2\Gamma\gamma^l}{1-\gamma} B_1 := D(l). \tag{3.5.16}$$

Proof. Let $v_j^l = \sum_{i=0}^{l-1} A^{*i}\xi_i^j$ and $v_j^\infty = \sum_{i=1}^{\infty} A^{*i}\xi_i^j$ correspondingly. Here ξ_i^j are *i.i.d.* sample from F, from $j = 1, 2, \ldots$. Then $v_1^l, v_2^l, \cdots$ are *i.i.d.* distributed and $v_1^\infty, v_2^\infty, \cdots$ are *i.i.d* sample from $\mathcal{D}$. By a geometric series argument, we have

$$\|v_j^\infty - v_j^l\| \leq \frac{\Gamma\gamma^l}{1-\gamma} B_1, \ \forall j. \tag{3.5.17}$$

Let

$$z'_0 = \tilde{z}_0 \tag{3.5.18}$$

$$z'_l = \tilde{z}_l - \underbrace{(A^{*l}\tilde{z}_0 - v_1^\infty + v_1^l)}_{:=u_l} := \tilde{z}_l - u_l = \sum_{i=0}^{l-1} A^{*i}\xi_{l-1-i} + \sum_{i=l}^{\infty} A^{*i}\xi_i^1 \tag{3.5.19}$$

$$\cdots \tag{3.5.20}$$

$$z'_{ml} = \tilde{z}_{ml} - \underbrace{(A^{*l}\tilde{z}_{(m-1)l} - v_m^\infty + v_m^l)}_{:=u_{ml}} := \tilde{z}_{ml} - u_{ml} = \sum_{i=0}^{l-1} A^{*i}\xi_{ml-1-i} + \sum_{i=l}^{\infty} A^{*i}\xi_i^m. \tag{3.5.21}$$

Then we have $(z'_0, z'_l, \cdots, z'_{ml})$ has same distribution as $i.i.d$ sample from $\mathcal{D}$, since all of they z's are using different ξ's. Then we have the following bound on u's,

$$\|u_{il}\| \le \|A^{*l}z_{(i-1)l}\| + \|v_i^\infty - v_i^l\| \le \Gamma\gamma^l B_2 + \frac{\Gamma\gamma^l}{1-\gamma}B_1 \le \frac{2\Gamma\gamma^l}{1-\gamma}B_1, \tag{3.5.22}$$

where we use the fact $B_2 \le \frac{\Gamma B_1}{1-\gamma}$ and (3.5.17). □

Applying the lemma on shifted subsequence, we have the following corollary.

Corollary 3.5.6. *Suppose $\tilde{z} = (\tilde{z}_j, \tilde{z}_{l+j}, \tilde{z}_{2l+j}, \cdots, \tilde{z}_{ml+j})$, for some $0 \le j < l$, is a subsequence from the uncontaminated vector autoregression process with some $\tilde{z}_j \sim \mathcal{D}$. Under the assumptions of mixing 3.5.2 and boundedness 3.5.4, there exist the random vectors $z' = (z'_j, z'_{l+j}, \cdots, z'_{ml+j})$, with each component an i.i.d. sample from the stationary distribution $\mathcal{D}$, and $u = (0, u_{l+j}, \cdots, u_{ml+j})$ such that*

$$z'_j = \tilde{z}_j, \tag{3.5.23}$$

$$(\tilde{z}_j, \tilde{z}_{l+j}, \tilde{z}_{2l+j}, \cdots, \tilde{z}_{ml+j}) \overset{(d)}{=} z' + u, \text{ and} \tag{3.5.24}$$

$$\|u_{il+j}\| \le \frac{2\Gamma\gamma^l}{1-\gamma}B_1 = D(l). \tag{3.5.25}$$

With the coupling (between sequences $\tilde{z}$ and z') argument above, we obtain that a subsequence of fixed stride of the uncontaminated process behaves close to $i.i.d$ samples z' from stationary distribution $\mathcal{D}$, and that this behavior is closer the larger the stride.

The next proposition gives the lower bound of the minimal eigenvalue of data matrix $Z' = [z'_1, z'_2, \cdots, z'_m]$. This is a direct result of [182, Remark 5.40].

Proposition 3.5.7. *Suppose $Z' \in \mathbb{R}^{d\times m}$ is the data matrix with each column i.i.d sampled from $\mathcal{D}$, with probability greater than $1-\delta$,*

$$\lambda_{min}(Z'Z'^T) \geq m\ell - C_1 \max\left(\sqrt{m\left(d+\log(1/\delta)\right)}, d+\log(1/\delta)\right).$$

Since our data is contaminated randomly, we derive the bound on the lower eigenvalue of the coupling sequence corresponding to the good index.

Proposition 3.5.8. *Suppose $Z' \in \mathbb{R}^{d\times m}$ is the data matrix with each column i.i.d sampled from $\mathcal{D}$, given any random index set $S \in [m]$ chosen independently of Z', then we have that there exists a C_1 independent of m, d, δ and fixed, such that, with probability greater than $1-\delta$, we have*

$$\lambda_{min}(Z'_S Z'^T_S) \geq |S|\ell - C_1 \max\left(\sqrt{m\left(d+\log(1/\delta)\right)}, d+\log(1/\delta)\right) := le(|S|, m, \delta).$$

Proof. The proof essentially follows [182, Remark 5.40], except that here it is extended to subsets of variable length (and thus both sides of the inequality conclusion of the proposition are now random variables). Let I_i denote the indicator of $i \in S$. We are going to bound the spectral norm of the following random matrix,

$$\Delta = \sum_{i=1}^{m} I_i \cdot \left(z'_i z'^T_i - \Sigma\right). \tag{3.5.26}$$

Define

$$
\begin{aligned}
f := \|\Delta\| &= \max_{v \in S^{d-1}} \left| \sum_{i=1}^{m} v^T I_i \cdot \left(z_i' z_i'^T - \Sigma \right) v \right| = \max_{v \in S^{d-1}} \left| \sum_{i=1}^{m} q_i(v) \right|, \\
q_i(v) &:= I_i \cdot v^T \left(z_i' z_i'^T - \Sigma \right) v, i = 1, 2, \ldots, m.
\end{aligned} \tag{3.5.27}
$$

We construct a $\frac{1}{4}$-cover of S^{d-1}, $\mathcal{N} = \{v_i\}_{i=1}^{C}$ [182, Def. 5.1]. That is, the points v_i are such that, for any $v \in S^{d-1}$ there exists an $i \leq C$ such that $\|v - v_i\| \leq \frac{1}{4}$. It is known that such a cover exists with $C \leq 9^d$ by [182, Lemma 5.2]. We assume that the cover $\mathcal{N}$ has this property for the rest of the proof. We define the finite approximation of f by

$$
f_n := \max_{v \in \mathcal{N}} \left| v^T \Delta v \right| \overset{(3.5.26)}{=} \max_{v \in \mathcal{N}} \left| \sum_{i=1}^{m} v^T I_i \cdot \left(z_i' z_i'^T - \Sigma \right) v \right| \overset{(3.5.27)}{=} \max_{v \in \mathcal{N}} \left| \sum_{i=1}^{m} q_i(v) \right|. \tag{3.5.28}
$$

We define the mapping $m : S^{d-1} \to \mathcal{N}$ as $m(v) = v_j$, with $j := \arg\min_{i \in 1,2 \ldots C} \|v - v_i\|$. That is $m(v)$ is the vector closest to v in the cover set $\mathcal{N}$ and in case of ties we chose the one with the lowest index. We then have that $\|v - m(v)\| \leq \frac{1}{4}$, $\forall v \in S^{d-1}$. We then obtain, for any $v \in S^{d-1}$,

$$
\begin{aligned}
|v^T \Delta v - m^T(v) \Delta m(v)| &= |v^T \Delta (v - m(v)) + (v - m(v))^T \Delta m(v)| \\
&\leq \|\Delta\| \|v\| \|v - m(v)\| + \|\Delta\| \|m(v)\| \|v - m(v)\| \leq \|\Delta\|/4 + |\Delta\|/4 \\
&= f/2.
\end{aligned}
$$

Subsequently, we obtain that

$$
\left| \sum_{i=1}^{m} q_i(v) \right| \overset{(3.5.27)}{=} \left| v^T \Delta v \right| \leq \left| m(v)^T \Delta m(v) \right| + \left| v^T \Delta v - m^T(v) \Delta m(v) \right| \overset{(3.5.28)}{\leq} f_n + f/2 \tag{3.5.29}
$$

Taking maximum over v, we obtain

$$f \leq f_n + f/2 \Longrightarrow f/2 \leq f_n \leq f.$$

To complete our proof, we then only need to show $f_n \leq C_3\sqrt{m(dC_2 + \log(1/\delta))}$ for some appropriate, constant C_2, C_3. Note that $|v^T I_i \cdot \left(z_i' z_i'^T - \Sigma\right) v| \leq (v^T z_i')^2 + v^T \Sigma v \leq 2$, given that $B_2 = 1$ (3.5.1).

Define $\mathcal{F}_{-1}$ to be the trival σ-field and $\mathcal{F}_0 = \sigma(\{I_i\}_{i=1}^m)$, $\mathcal{F}_t$ to be the σ-field containing $\mathcal{F}_{t-1}$ and $\sigma(z_t')$. Note that S is chosen independently from Z', by the proposition's hypothesis. Then we know that $I_i \in \mathcal{F}_0 \subseteq \mathcal{F}_{i-1}$, $z_i' \perp \mathcal{F}_{i-1}$ and $z_i' \in \mathcal{F}_i$, and we have

$$q_i(v) = I_i v^T \left(z_i' z_i'^T - \Sigma\right) v, \quad \mathbb{E}[q_i(v) \mid \mathcal{F}_{i-1}] = I_i \cdot v^T \mathbb{E}[z_i' z_i'^T - \Sigma] v = 0, \quad q_i(v) \in \mathcal{F}_i. \tag{3.5.30}$$

Therefore, $q_i(v)$ is a bounded martingale difference sequence w.r.t $\mathcal{F}_i$. By martingale concentration [133, Theorem 3.5], for any $v \in \mathcal{N}$ we have, with probability greater than $1 - \delta/9^d$, that

$$\left|\sum_{i=1}^m q_i(v)\right| \leq C_1\sqrt{m(d\log 9 + \log(1/\delta))}.$$

Here C_1 is a constant with respect to all stated parameters, particularly m. Using a union bound on probability sets applied to the complement of the set on which the above is true, and noting that $|\mathcal{N}| \leq 9^d$ we obtain that,

$$P\left(\left|\sum_{i=1}^m q_i(v)\right| \leq C_1\sqrt{m(d\log 9 + \log(1/\delta))},\ \forall v \in \mathcal{N}\right) \geq 1 - 9^d \cdot \delta/9^d = 1 - \delta.$$

Then we have, using (3.5.28), that $P(f_n \leq C_1\sqrt{m(d\log 9 + \log(1/\delta))} \geq 1 - \delta$ and thus that,

with probability at least $1-\delta$, the following statement holds

$$f \leq 2f_n \leq 2C_1\sqrt{m(d\log 9 + \log(1/\delta))}. \tag{3.5.31}$$

Given the definition of f in (3.5.27) this implied that with probability greater than $1-\delta$, we have that

$$\left| v^T Z'_S {Z'_S}^T v - \sum_{i=1}^{m} I_i v^T \Sigma v \right| \leq 2C_1\sqrt{m(d\log 9 + \log(1/\delta))}, \quad \forall v \in S^{d-1}$$

Then with probability at least $1-\delta$, the following sequence of inequalities holds:

$$\begin{aligned}
\forall v \in S^{d-1}: \quad & v^T Z'_S {Z'_S}^T v \geq \sum_{i=1}^{m} I_i v^T \Sigma v - 2C_1\sqrt{m(d\log 9 + \log(1/\delta))} \\
\forall v \in S^{d-1}: \quad & v^T Z'_S {Z'_S}^T v \overset{\text{A3.5.4}}{\geq} \sum_{i=1}^{m} I_i \ell - 2C_1\sqrt{m(d\log 9 + \log(1/\delta))} \\
\forall v \in S^{d-1}: \quad & v^T Z'_S {Z'_S}^T v \geq |S|\ell - 2C_1\sqrt{m(d\log 9 + \log(1/\delta))}
\end{aligned}$$

Taking the minimum over all v in the last inequality we obtain $\lambda_{min}\left(Z'_S {Z'_S}^T\right) \geq |S|\ell - 2C_1\sqrt{m(d\log 9 + \log(1/\delta))}$ which concludes the proof after the relabeling $2C_1\sqrt{\log 9} \to C_1$. □

Combining the coupling results and the lower bound on $\lambda_{min}(Z'_S Z'^T_S)$, we have the following.

Lemma 3.5.9. *Let S_{good} be the good index set. Let Z be the design matrix corresponding to the good data part of the regression problem $Z_{S_{good}} = [x_i]_{i\in S_{good}} = [z_i]_{i\in S_{good}} = [\tilde{z}_i]_{i\in S_{good}}$.*

Then we have, with probability greater than $1-2\delta$, for n is large enough, we have,

$$\lambda_{min}(Z_{S_{good}}Z_{S_{good}}^T) \geq 0.49(1-2\alpha)\ell n. \tag{3.5.32}$$

Proof. For notational simplicity we assume p, n/p are integers. We first divide $\{0,1,\cdots,n-1\}$ into p blocks $B_0 = \{0,p,2p,\cdots n\},\cdots,B_{p-1} = \{p-1,2p-1,\cdots,n-1\}$, with block size n/p. We denote the blocks' intersection with S_{good} as $S_1,S_2,\cdots,S_p$, with size $|S_i| = m_i$. Note that all the index set S_i are independent of $\tilde{z}_i,\xi_i$. Applying the coupling results, let z'_j,u_j be the coupling correspondingly to Corollary 3.5.6 i.e. $\tilde{z}_j = z'_j + u_j$ on the corresponding block. We then have that

$$\begin{aligned}
\forall v \in S^{d-1}: \sum_{j\in S_{good}}(v^T z_j)^2 = \sum_{j\in S_{good}}(v^T\tilde{z}_j)^2 &= \sum_{j\in S_{good}}\left[v^T(z'_j+u_j)\right]^2 \\
&\geq \frac{1}{2}\sum_{j\in S_{good}}(v^T z'_j)^2 - \sum_{j\in S_{good}}(v^T u_j)^2 \\
&\overset{\textbf{Corollary}3.5.6}{\geq} \frac{1}{2}\sum_{i=0}^{p-1}\sum_{j\in S_i}(v^T z'_j)^2 - nD^2(p) \\
&\geq \frac{1}{2}\sum_{i=0}^{p-1}\lambda_{min}\left(Z'_{S_j}Z'^T_{S_j}\right) - nD^2(p)
\end{aligned}$$

The second line follows from Jensen's inequality applied to x^2 as: $(a+b)^2 + (-b)^2 \geq \frac{1}{2}(a+b-b)^2 = \frac{1}{2}a^2$ which implies $(a+b)^2 \geq \frac{1}{2}a^2 - b^2$. We now apply Proposition 3.5.8 to each of the p terms of the outer sum of the last displayed inequality with probability $1-\frac{\delta}{p}$, and use an union bound to have its conclusion apply simultaneously to all of them,

to conclude that, with probability $1 - p\frac{\delta}{p} = 1 - \delta$ the following holds.

$$\forall v \in S^{d-1} : \sum_{j \in S_{good}} (v^T z_i)^2 \geq \frac{1}{2} \sum_{i=0}^{p-1} le(m_i, n/p, \delta/p) - nD^2(p)$$

$$\geq \frac{1}{2} \sum_{i=0}^{p-1} m_i \ell - \sum_{i=0}^{p-1} \left[C_1 \max(\sqrt{(n/p)(d + \log(p/\delta)}, d + \log(p/\delta)) \right] \tag{3.5.33}$$

$$- nD^2(p)$$

$$\geq \frac{1}{2} |S_{good}| \ell - C_1 \max\left(\sqrt{np(d + \log(p/\delta))}, p(d + \log(p/\delta)) \right) \tag{3.5.34}$$

$$- nC_2 \gamma^{2p}. \tag{3.5.35}$$

In applying proposition 3.5.8 recall that, by Corollary 3.5.6, Z'_{B_i} are $i.i.d$ copy from $\mathcal{D}$ and Z'_{S_j} plays the role of matrix Z'_S in Proposition 3.5.8 with index mapping $i \to \lfloor i/p \rfloor + 1$ on each block. The expression of $D(p)$ we use to obtain the final bound originates in Corollary 3.5.6.

Next we take $p = 10 \log n/(1 - \gamma)$. When $n \to \infty$, it easy to see the second term in (3.5.35) becomes $O(\sqrt{n \log(n) \log\log(n)})$. For the third term, note that, from our choice of p we obtain

$$2p \log(\gamma) = 2p \log(1 + (\gamma - 1)) \leq 2p(\gamma - 1) \leq -5 \log n.$$

Raising the end terms above as exponents of γ, and multiplying by n we obtain $n\gamma^{2p} \leq n^{-4}$. Therefore the third term in (3.5.35) is $O(n^{-4})$ and the sum of the second and third term in (3.5.35) is $o(n)$, which will be negligible compared to the first term $\Omega(n)$ due to our contamination assumption. Therefore, when n large enough we have from (3.5.35), by taking the minimum over the left hand side and dividing by n (and, recall $0 \leq \alpha < 1/8$), that

$$\frac{1}{n} \lambda_{\min}(Z_{S_{good}} Z_{S_{good}}^T) \geq 0.5(1 - 2\alpha)\ell - o(n)/n > 0.49(1 - 2\alpha)\ell. \tag{3.5.36}$$

This completes the proof. □

Theorem 3.5.10. *Consider the VAR model and taking the Huber's loss with $r = 2B_1$. If $11\alpha/\ell < 1$, when n is large enough, with probability greater than $1 - 2\delta$, we have*

$$\|\hat{A} - A^*\|_F \leq \frac{C_3 B_1 \sqrt{\log(1/\delta)}}{\sqrt{n}\ell} + \frac{4\alpha B_1}{0.49(1-2\alpha)l}. \tag{3.5.37}$$

Proof. The proof essentially follows from Theorem 3.4.1, other then inner product becomes the matrix inner product and the norm becomes the matrix Frobenius norm. With similar procedure, first, we assume that $\|\hat{A} - A^*\|_2 \leq \|\hat{A} - A^*\|_F \leq r - B_1 = B_1$, than all of the good sample will be fail in the quadratic part of Huber's loss. Note that $B_2 = 1$. Plug our Lemma 3.5.1 and Lemma 3.5.9 into (3.4.8), we have

$$\|\hat{A} - A^*\|_F \leq \frac{C_2 B_1 \sqrt{\log(1/\delta)}}{\sqrt{n}\ell} + \frac{2\alpha r}{0.49(1-2\alpha)\ell}. \tag{3.5.38}$$

Therefore, with $\alpha < 1/8$, we have $\|\hat{A} - A^*\|_F < 5.34\alpha r/\ell = (10.68\alpha/\ell)\, B_1 < B_1$ for n large enough. It obvious with $r = 2B_1$, $\hat{A}$ is also in the interior of the ball $\|\hat{A} - A^*\| < B_1$, which justify the local minimizer $\hat{A}$ is also the global minimizer of the convex program. □

Remark 3.5.11. With the above lemmas, we can show that with high probability our estimator $\hat{A}$ will be close to true A, $\|A - \hat{A}\|_F \leq \epsilon_1 < B_1$ specified in (3.5.37). Also, with high probability, we know that all of the good samples will fall into the quadratic part of Huber's loss when A takes $\hat{A}$. Following the exact argument as in Remark 3.4.2, we have that for bad sample that kept from robust screening and falls into the quadratic part, the corresponding residual $\|z_{i+1} - A^* z_i\| = \|\xi_i + o_{i+1} - A^* o_i\| \leq \|z_{i+1} - \hat{A} z_i + (\hat{A} - A^*) z_i\| = \|z_{i+1} - \hat{A} z_i\| + \|(\hat{A} - A^*) z_i\| \leq 2B_1 + 1 \cdot B_1 = 3B_1$.

3.5.2 DRO Formulation

For this section, we slightly change the robust screening and regression procedure. It is not set up as follows.

1. Process time series data $(z_0, \cdots, z_n)$. If any $\|z_t\| \geq B_2$, we substitute z_t by arbitrary v with $\|v\| \leq B_2$ and add t to $\mathcal{S}_{untrust}$. Also, we denote $(x_i, y_i) = (z_i, z_{i+1})$.
2. Perform robust regression with Huber's loss to get $\hat{A}$.
3. If the residual $y_i - \hat{A}x_i$ is mapped to the linear (outlier) part of Huber's loss, we add i and $i+1$ into $\mathcal{S}_{untrust}$.
4. Let $\mathcal{S}_{trust} = \{0, \cdots, n\}/\mathcal{S}_{untrust}$.

Note that, we obtain an identical bound for $\|A - \hat{A}\|$ as in Theorem 3.5.10; the proof is virtually identical for the new procedure. After the robust regression, we get the estimated $\hat{A}$. We then (3) can drop the data of $(x_i, y_i) = (z_i, z_{i+1})$ that maps to the linear (outlier) part of the Huber loss. We redefine the trust set to be the set after robust replacement and the screening based on Huber's loss. Observe that, from Remark 3.4.2 applied to the autoregressive case, where a data point consists of two consecutive time series entries, we have that with probability at least $1 - 2\delta$ that

F1: If $(z_{i-1}, z_i, z_{i+1}, z_{i+2})$ are all uncontaminated, then $i, i+1 \in \mathcal{S}_{trust}$ and (z_i, z_{i+1}) will be kept for the purpose of evaluating residuals.

F2: If (z_{i-1}, z_i, z_{i+1}) are all uncontaminated, then i will be kept.

F3: If z_i is contaminated, then $i-1, i, i+1$ can all potentially be in $\mathcal{S}_{untrust}$ even if z_{i-1} and z_{i+1} are uncontaminated.

We define *last trust* in under different assumptions that belongs to $\mathcal{S}_{trust}$. Note that $z_{\text{last trust}}$ can still be contaminated. Nevertheless, we have the following deviation guarantee for it.

Lemma 3.5.12. *Under one of these assumptions,*

A1: When random contamination occurs, no consecutive two points $\tilde{z}_i, \tilde{z}_{i+1}$ are contaminated.

A2: If i is the last contaminated index, then the data at indices $i-1$, $i-2$, and $i-3$ are uncontaminated.

A3: The data at indices $n-1$ and n are uncontaminated.

We define $1 \leq$ "last trust" $\leq$ to be

L1: The largest index i among $1, 2, \ldots, n$ where both $i-1$ and i are in $\mathcal{S}_{trust}$ under assumption A1.

L2: The largest index i among $1, 2, \ldots, n$ in $\mathcal{S}_{trust}$ under assumptions A2 or A3.

We then have the following guarantee for $z_{\text{last trust}}$,

$$\|z_{\text{last trust}} - \tilde{z}_{\text{last trust}}\| \leq 4B_1. \tag{3.5.39}$$

Remark 3.5.13. Note that these three assumptions do not violate the independent contamination assumption since the contamination index is just sampling with restrictions but still independent of $\tilde{z}_i, \xi_i$.

Proof. Recall that $o_t := z_t - \tilde{z}_t$. We first state a useful result: If we have that $i, i+1 \in \mathcal{S}_{trust}$ *and* we assume at least one of them is uncontaminated, then we have,

$$\|z_{i+1} - \tilde{z}_{i+1}\| = \|o_{i+1}\| \leq I_{o_{i+1}=0} \cdot 0 + I_{o_{i+1}\neq 0, o_i = 0} \left(\|o_{i+1} + \xi_i - A^* \cdot 0\| + \|\xi_i\|\right) \tag{3.5.40}$$

$$\leq \|o_{i+1} + \xi_i - A^* o_i\| + \|\xi_i\| \overset{\textbf{Remark } 3.5.11}{\leq} 3B_1 + B_1 = 4B_1. \tag{3.5.41}$$

Next we derive (3.5.39) for the different assumptions. Under assumption A1, let (z_{n-c-1}, z_{n-c}) be the last data point **pair** that is kept and thus, by the definition L1 of

the "last trust" index we get $z_{\text{last trust}} = z_{n-c}$. Then we can apply the useful result above to obtain from (3.5.11) that $\|\tilde{z}_{n-c} - z_{n-c}\| \leq 4B_1$ to prove (3.5.39) in this case. Under assumption A3, we know that $z_n = z_{\text{last trust}}$ is kept and uncontaminated. Thus $\|z_{\text{last trust}} - \tilde{z}_{\text{last trust}}\| = 0 \leq 4B_1$ which proves (3.5.39) in this case as well.

The proof under Assumption A2 requires a bit more discussion. Let last trust $= n - c$ and thus $z_{\text{last trust}} = z_{n-c}$. We discuss the following cases.

Case 1: If the last contamination occurs before index $i \leq n - 2$ we know that z_n is kept and $\|z_n - \tilde{z}_n\| = 0 \leq 4B_1$, and (3.5.39) holds.

Case 2: If the last contamination occurs at index $n - 1$ we have two cases. (i) If z_n is kept by the screening procedure then $\|z_n - \tilde{z}_n\| = 0 \leq 4B_1$. (ii) If z_n is not kept by the screening procedure, then from A2 we know that $z_{n-2}, z_{n-3}, z_{n-4}$ are uncontaminated. Then we know at least one of z_{n-1}, z_{n-2} or z_{n-3} will be kept using F2 and with guarantee $\|z_{\text{last trust}} - \tilde{z}_{\text{last trust}}\| \leq 4B_1$ (the difference is zero if "last trust" is either $n - 2$ or $n - 3$, whereas if "last trust" is $n - 1$ then both $n - 1$ and $n - 2$ are kept and the latter is uncontaminated and the conclusion follows from (3.5.40)).

Case 3: If the last contamination occurs at index n, then, either (i) z_n is kept, last trust $= n$, and, from Assumption A2, z_{n-1} is uncontaminated. Therefore, using (3.5.40), we have $\|z_n - \tilde{z}_n\| \leq 4B_1$. Otherwise, (ii) since $z_{n-1}, z_{n-2}, z_{n-3}$ are uncontaminated, we know, using F3, that at least one of z_{n-1}, z_{n-2} will be kept which guarantees, since they are uncontaminated, that $\|z_{\text{last trust}} - \tilde{z}_{\text{last trust}}\| = 0 \leq 4B_1$.

□

Remark 3.5.14. We note that some assumptions about constraining the contamination on the tail part is needed to obtain a good behavior for the relaxation (3.4.13). Otherwise, the contamination index can be $\{n - \alpha n + 1, \cdots, n - 1, n\}$ (the entire fraction α of the end sequence is contaminated), which is deterministic and therefore independent of $\tilde{z}_i, \xi_i$. Such a contamination can basically 'design' an arbitrarily undetectable tail sequence, (e.g., draw

another independent sample sequence starting from $z_{n-\alpha n}$). Let the new independent tail sequence starting from $\tilde{z}_{n-\alpha n}$ be $z'_{n-\alpha+1} = A^* \tilde{z}_{n-\alpha n} + \xi'_{n-\alpha n}, \cdots z'_n$. Then we know

$$z'_n - (A^*)^{\alpha n} \tilde{z}_{n-\alpha n} \perp \tilde{z}_n - (A^*)^{\alpha n} \tilde{z}_{n-\alpha n}, \tag{3.5.42}$$

which is almost irrelevant to the actual (uncontaminated) tail sequence (after subtracting an exponentially small term in $k = n\alpha$, see, e.g., the proof of Lemma 3.5.5).

Note that, if last trust $= n - c$, we have the characterization of $\|z_{n-c} - \tilde{z}_{n-c}\| \leq 4B_1$ from Lemma 3.5.12. From our model, we have $\tilde{z}_{n+1} = A^{*c+1} \tilde{z}_{n-c} + \sum_{k=0}^{c} A^{*k} \xi_{n-k}$, which will allow us to control the true objective of the original problem. Let $F(A)$ be the empirical distribution of the residual given transition matrix A, and $\hat{A}$ be the regression estimation. With p-Wasserstein distance as our distribution robustness metric, the DRO formulation is

$$\begin{aligned} \min_{a} \max_{A,z,F} \mathbb{E}_{\underbrace{F \otimes F \otimes \cdots \otimes F}_{c+1}} [h(a, A^{c+1} z + \sum_{k=0}^{c} A^k \xi'_i)] \\ \text{s.t. } \|A - \hat{A}\|_F \leq \epsilon_1 \\ W_p^p(F, F(A)) \leq \epsilon_2 \\ \|z - z_{n-c}\| \leq 4B_1. \end{aligned} \tag{3.5.43}$$

Since the tensor product cannot be approximated by a finite formulation, we will relax it. We first define the p-Wasserstein distance on the product space $\mathcal{X} \times \mathcal{X}$. Suppose Wasserstein distance W_p is defined on $\mathcal{X}$ with norm $\|\cdot\|$. Then we can define a distance (actually a norm) on the product space $\mathcal{X} \times \mathcal{X}$ with metric $d_p^p((x_{1,1}, x_{1,2}), (x_{2,1}, x_{2,2})) = \|x_{1,1} - x_{2,1}\|^p + \|x_{1,2} - x_{2,2}\|^p$. Then W_p^p on $\mathcal{X} \times \mathcal{X}$ is defined using this metric. Let $F \otimes F$ denote the (independent) product distribution on $\mathcal{X} \times \mathcal{X}$, i.e., $F \otimes F(A_1 \times A_2) = F(A_1)F(A_2)$ with A_i measurable on $\mathcal{X}$. Then we have that $W_p^p(F \otimes F, G \otimes G) \leq 2W_p^p(F, G)$ with equality for the case $p = 2$ by [129, §2,pg.412].

To justify this statement, we write Wasserstein distance in the expectation form

$$W_p^p(F,G) = \inf_{(X,Y)\sim\pi,\pi\in\Pi(F,G)} \mathbb{E}[\|X-Y\|^p]. \tag{3.5.44}$$

Recall that $\Pi(F,G)$ denotes the joint distribution with marginal F and G. For any $\epsilon > 0$, from (3.5.44) we can then select $(X,Y) \sim \pi$, where π has marginal F, G, such that

$$\mathbb{E}_{(X,Y)\sim\pi}[\|X-Y\|^p] \leq W_p^p(F,G) + \epsilon. \tag{3.5.45}$$

Then let $(X_i, Y_i)_{i=1}^2$ be two $i.i.d$ copies from (X, y). We have that $(X_1, X_2) \sim F \otimes F$ and $(Y_1, Y_2) \sim G \otimes G$, and

$$W_p^p(F\otimes F, G\otimes G) \leq \mathbb{E}_{\pi\times\pi}\left[\|X_1-Y_1\|^2 + \|X_2-Y_2\|^p\right] \leq 2W_p^p(F,G) + 2\epsilon. \tag{3.5.46}$$

Let $\epsilon \to 0$, we get the desired inequality.

In practice, c always takes the value 0 or 1. Since the case $c = 0$ is essentially equivalent to the case from [47], we use the case $c = 1$ as an example to demonstrate the idea. The problem (3.5.43) can be relaxed to

$$\begin{aligned} \min_a \max_{A,z,F_2} \; & \mathbb{E}_{(\xi_1',\xi_2')\sim F_2}[h(a, A^2 z + \xi_1' + A\xi_2')] \\ \text{s.t. } & \|A - \hat{A}\|_F \leq \epsilon_1 \\ & W_p^p(F_2, F(A)\otimes F(A)) \leq 2\epsilon_2 \\ & \|z - z_{n-c}\| \leq 4B_1. \end{aligned} \tag{3.5.47}$$

Here the second constraint stems from (3.5.46) $W_p^p(F\otimes F, G\otimes G) \leq 2W_p^p(F,G)$. At this stage, we use our Proposition 3.4.4 to approximate this problem by a finite dimensional

min-max problem:

$$\begin{aligned}\min_{a} \max_{A,z,\{\xi'_{i,j,1},\xi'_{i,j,2}\}} & \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\mathbb{E}_F[h(a, A^2 z + \xi'_{i,j,1} + A\xi'_{i,j,2})] \\ \text{s.t. } & \|A - \hat{A}\|_F \le \epsilon_1 \\ & \frac{1}{m^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\|\xi'_{i,j,1} - \xi_i\|^p + \|\xi'_{i,j,2} - \xi_j\|^p \le 2\epsilon_2 \\ & \xi_i = z_{i+1} - Az_i \\ & \|z - z_{n-c}\| \le 4B_1.\end{aligned} \tag{3.5.48}$$

Here, the parameter m indicates the number of empirical residuals we can access, which after the screening, we expect it to be $O(n)$. By Theorem 3.4.5, we know that the solution of (3.5.48) has performance no worse than $2B/m^2$ compared to (3.5.47) if h is continuous and bounded by B.

Since this approach will give $O(m^2)$ variables. We can use a simpler DRO formulation inspired by a common approach in multi-step time series prediction [173]. We still use $c = 1$ as an example. Instead to estimating the distribution of ξ, *we estimate the distribution* of $A\xi_i + \xi_{i+1}$. This gives the mapping $G(A) = \sum_{(i,i+1,i+2)\in\mathcal{S}_{trust},i=0 \bmod 2} \delta_{z_i - A^2 z_{i-2}}$. Then we formulate for DRO as follows,

$$\begin{aligned}\min_{a} \max_{A,z,G} & \mathbb{E}_{u\sim G}[h(a, A^2 z + u)] \\ \text{s.t. } & \|A - \hat{A}\|_F \le \epsilon_1 \\ & W(G, G(A)) \le \epsilon_2 \\ & \|z - z_{n-c}\| \le 4B_1.\end{aligned} \tag{3.5.49}$$

This results in the following overall screening+DRO procedure.

1. Receive time series $(z_0, \cdots, z_n)$. If any $\|z_t\| \ge B_2$, we substitute z_t by arbitrary v with

$\|v\| \leq B_2$ and add t to $\mathcal{S}_{untrust}$. Also, we denote $(x_i, y_i) = (z_i, z_{i+1})$.

2. Perform robust regression with Huber's loss to get $\hat{A}$.

3. If $y_i - \hat{A}x_i$ maps to the linear part of Huber's loss, we add i and $i+1$ into $\mathcal{S}_{untrust}$.

4. Let $\mathcal{S}_{trust} = \{0, \cdots, n\}/\mathcal{S}_{untrust}$, and suppose $c = 1$, that is, the largest index in $\mathcal{S}_{trust}$ is $n-1$.

By Proposition 3.4.4, we have the following finite approximation of the problem,

$$\begin{aligned} \min_{a} \max_{A,z,\xi_i'} & \mathbb{E}_{u\sim G}[h(a, A^2 z + \xi_i')] \\ \text{s.t. } & \|A - \hat{A}\|_F \leq \epsilon_1 \\ & \frac{1}{m}\sum_{i=1}^{m} \|\xi_i' - \xi_i\|^p \leq \epsilon_2 \\ & \xi_i = z_i - A^2 z_{i-2}, \text{ for } i \text{ in } \mathcal{S}_{trust},\ m = |\mathcal{S}_{trust}| \\ & \|z - z_{n-1}\| \leq 4B_1. \end{aligned} \tag{3.5.50}$$

By Theorem 3.4.5, plugging the solution of (3.5.50) into the (3.5.49), the performance wouldn't be worse than $2B/m$ than the actual optimal solution given that h is continuous and bounded by B.

ϵ specification

Here we note that the selection of ϵ_1, ϵ_2 can be chosen in a same fashion as in §3.4.2. Still the selection of ϵ_1 is suggested in (3.5.37). Reader may have concern that the robust screening procedure may make the residuals dependent, we address this issue in Theorem 3.7.2 to show that a subsequence composite of $i.i.d$ residuals will guarantee to be kept.

We still use case $c = 1$ to demonstrate how to select ϵ_2 for (3.5.49), still we know $A^*\xi_i + \xi_{i+1} = z_{i+2} - A^{*2} z_i$ is included if $(i-1, i, i+1, i+2, i+3)$, $i = 0 \bmod 2$ are

not contaminated. Therefore, there are $n_{good} = n/2 - 2\alpha n$ number (if there are more, we keep the first n_{good} number of them) of the residual are uncontaminated and $i.i.d$ from the distribution of $\xi_{i+1} + A^*\xi_i$ (note that $i = 0 \bmod 2$ guarantees that they are independent), for a more detailed discussion, see §3.7. For the contaminated $z_{i+2} - A^{*2} z_i$, since $(i, i+1, i+2)$ are trusted, by Remark 3.5.11, we know,

$$\|z_{i+2} - A^{*2} z_i\| = \|z_{i+2} - A^* z_{i+1} + A^* z_{i+1} - A^{*2} z_i\| \tag{3.5.51}$$

$$\leq \|z_{i+2} - A^* z_{i+1}\| + \|A^*\|\|A^* z_{i+1} - A^{*2} z_i\| \leq 3B_1 + 3\|A^*\|B_1, \tag{3.5.52}$$

and $\|A^*\| \leq \|A^*\|_F \leq \|\hat{A}\|_2 + \epsilon_1 := e$. Then $\|z_{i+2} - A^{*2} z_i\| \leq 3(1+e)B_1$, also we know $\|\xi_{i+1} + A^*\xi_i\| \leq (1+e)B_1$. Therefore, we can write $G(A^*) = (1-\alpha')G_{good}(A^*) + \alpha' G_{bad}(A^*)$ and $1 - \alpha' = (n/2 - 2\alpha n)/m \geq (n/2 - 2\alpha n)/(n/2) = 1 - 4\alpha$. Then we can select $W_p^p(G(A^*), G) \leq (1-\alpha')W_p^p(G_{good}(A^*), G) + \alpha' W_p^p(G_{bad}(A^*), G)$
$\leq CB_1{}^p \left((n/2 - 2\alpha n)^{-p/d}\right) + 4\alpha(4(1+e)B_1)^p = \epsilon_2$, whereas the first term is by [61], and second is by the boundness results derived above. Then we have the following guarantee.

Theorem 3.5.15. *With ϵ_1 and ϵ_2 select above, with probability greater than $1 - 3\delta$, the true transition matrix A^* and distribution G will be feasible for DRO formulation* (3.5.49).

3.6 Experiment

3.6.1 Synthetic Data

For the synthetic data experiment. We take the target function to be linear

$$h(a, z) = a^T z, \tag{3.6.1}$$

to model invest in stock. Here our decision set $a \in \Delta^{d-1} = \{a | \sum_i a_i = 1, a_i \geq 0\}$ is the $(d-1)$-dimensional simplex, and the data is from a d dimensional VAR(1) model.

$$z_{t+1} = Az_t + \xi_t + o_t. \tag{3.6.2}$$

Here, we let ξ_i sampled from $i.i.d$ Normal distribution $N(0, RI_d)$ and truncation at radius R, i.e. $x = Rx/\|x\|$ if $\|x\| \geq R$, where A is sampled $i.i.d$ entry-wise from standard Gaussian and let $A = 0.8A/\|A\|_2$. We generate random sample $\{z_t\}_{i=1}^T$ with different sample size T, and let o_i be 0 with probability $1-\alpha$,or sampled $i.i.d$ entry-wise from Cauchy distribution with contamination probability $\alpha = 0.1$. We perform regression and DRO with 2-Wasserstein distance framework (3.5.50) on the data set to make a robust decision a given the last trust sample z_{T+1-c} to maximize $a^T z_{T+1}$.

$$\max_a \min_{A,z,\{\xi_i\}} \frac{1}{m} \sum_{i \in \text{trust}} \langle a, A^c z + \xi_i \rangle \tag{3.6.3}$$

$$\text{s.t. } \hat{\xi}_i = z_{i+c} - A^c z_i \tag{3.6.4}$$

$$\frac{1}{m} \sum \|\xi_i - \hat{\xi}_i\|^2 \leq \epsilon_2 \tag{3.6.5}$$

$$\|A - \hat{A}\|_F \leq \epsilon_1 \tag{3.6.6}$$

$$\|z - z_{T+1-c}\| \leq 4R. \tag{3.6.7}$$

The suggested ϵ_1, ϵ_2 can be too conservative, so we multiply by $\delta = 0.5, 0.1, 0.01$ and choosing the best using the experiments conducting on the first 200 trails. We report our results as 'regret' (as if we observe z_{T+1}), $\max[z_{T+1}] - a^T z_{T+1}$. We also compare the results to MLE, which is selecting the largest index using $\hat{A}^c z_{T+1-c}$, and sample average

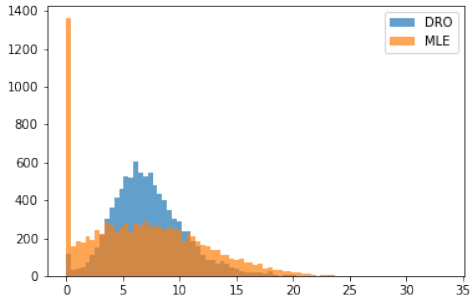


Figure 3.1: Comparison of DRO and MLE, synthetic data. All data are normalized by noise radius R. Noise radius is 20. Sample size is 15. Problem dimension is 10.

approximation, which is optimizing

$$\max_{a} \min_{A,z} \frac{1}{m} \sum_{i \in \text{trust}} \langle a, A^c z + \xi_i \rangle \tag{3.6.8}$$

$$\text{s.t. } \hat{\xi}_i = z_{i+c} - A^c z_i \tag{3.6.9}$$

$$\|A - \hat{A}\|_F \leq \epsilon_1 \tag{3.6.10}$$

$$\|z - z_{T+1-c}\| \leq 4R. \tag{3.6.11}$$

We solve the original finite formulation min-max programming and other min/max formulations by 1) extra gradient method [104] 2) Gradient descent ascent. The 'regret' reported are normalized by dividing R. The results with different settings are summary in Table 3.1. In the lower noise and lower dimension regime, SAA/MLE performs better on average. Our robust method outperforms other methods in the higher dimension/higher noise regime. Across all the settings, our DRO method has the lightest tail. A sample of histogram is displayed in Figure 3.1 and Figure 3.2.

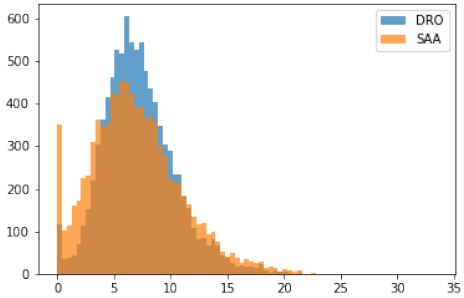


Figure 3.2: Comparison of DRO and SAA, synthetic data. All data are normalized by noise radius R. Noise radius is 20. Sample size is 15. Problem dimension is 10.

Setup	Mean	Median	75th Perc.	90th Perc.
10-20-50	0.3493/0.3138/**0.3005**	0.338/0.2976/**0.2775**	0.4239/**0.4319**/0.4868	**0.5114**/0.5416/0.6576
5-20-50	0.2517/0.2092/**0.1942**	0.2371/0.1846/**0.1141**	0.322/**0.2915**/0.3208	**0.4227**/0.4523/0.5277
10-20-15	0.369/0.3655/**0.3479**	0.3549/0.3419/**0.331**	**0.4633**/0.4832/0.5153	**0.5661**/0.6511/0.7117
5-20-15	0.2738/0.2543/**0.2537**	0.2557/0.2264/**0.2185**	**0.3563**/0.3575/0.426	**0.4463**/0.4966/0.5935
10-5-50	0.4763/**0.4532**/0.4564	0.4678/**0.444**/0.4488	**0.5728**/0.6218/0.7132	**0.6718**/0.8094/0.9157
5-5-50	0.3749/0.3232/**0.3095**	0.3673/0.2522/**0.1707**	**0.5068**/0.5194/0.5518	**0.6575**/0.7424/0.8518
10-5-15	0.4936/0.4723/**0.4629**	0.4744/0.4538/**0.4406**	**0.6255**/0.6699/0.6892	**0.766**/0.8561/0.9453
5-5-15	0.4348/**0.4019**/0.405	0.4222/0.3593/**0.3015**	**0.575**/0.6386/0.6887	**0.7542**/0.8709/1.0291

Table 3.1: Comparison of DRO, SAA and MLE for several standard percentiles, synthetic data. Setup is "dimension(d)-noise radius(R)-sample size(T)." Lowest regret among methods is boldfaced.

3.7 Appendix

We need the following lemma which states that a random indexed subsequence of *i.i.d* sequence is still a *i.i.d* sequence.

Lemma 3.7.1. *Suppose $X = (X_1, \cdots, X_n)$ is i.i.d sequence from P, and $S = (s_1, \cdots, s_m)$ with $|S| = m$ and $s_i \neq s_j$ is a random index sequence which is sampled independent of X, then $X_S = (X_{s_1}, \cdots, X_{s_m})$ is still a i.i.d sequence from P.*

Proof. We prove the case for $n = 3$ and $m = 2$ to demonstrate the idea. It suffices to show for any A_1, A_2 measurable,

$$P(X_{s_1} \in A_1 \text{ and } X_{s_2} \in A_2) = P(A_1)P(A_2) \tag{3.7.1}$$

$$LHS = \mathbb{E}\left[I_{X_{s_1} \in A_1} I_{X_{s_2} \in A_2}\right]. \tag{3.7.2}$$

We build the following function,

$$p(i, x_i) = \left(1 - \sum_{j=1}^{2} I_{i=s_j} I_{x_i \in A_j^c}\right), \tag{3.7.3}$$

$$I_{X_{s_1} \in A_1} I_{X_{s_2} \in A_2} = \prod_{i=1}^{3} p(i, X_i). \tag{3.7.4}$$

Let $\mathcal{F}_0 = \sigma\left(\{s_i\}_{i=1}^2\right)$, then we have

$$\mathbb{E}\left[I_{X_{s_1} \in A_1} I_{X_{s_2} \in A_2}\right] = \mathbb{E}\left[\prod_{i=1}^{3} p(i, X_i)\right] = \mathbb{E}\left(\mathbb{E}\left[\prod_{i=1}^{3} p(i, X_i) \mid \mathcal{F}_0\right]\right). \tag{3.7.5}$$

Using the fact that $X_i \perp \mathcal{F}_0$ and $I_{i=s_j} \in \mathcal{F}_0$, we have, e.g., $\mathbb{E}[I_{i=s_1} I_{X_i \in A_1^c} \mid \mathcal{F}_0] =$

$I_{i=s_1}\mathbb{E}[I_{X_i\in A_1^c}] = I_{i=s_1}P(A_1^c)$, using this property we have

$$\mathbb{E}\left[\prod_{i=1}^{3} p(i, X_i) \mid \mathcal{F}_0\right] \tag{3.7.6}$$

$$= 1 - \left(\sum_{i=1}^{3} I_{i=s_1}\right) P(A_1^c) - \left(\sum_{i=1}^{3} I_{i=s_2}\right) P(A_2^c) + \left(\sum_{i_1\neq i_2} I_{i_1=s_1} I_{i_2=s_2}\right) P(A_1^c)P(A_2^c) + \tag{3.7.7}$$

$$\sum_{k=1}^{2}\sum_{i\neq j} I_{i=s_k} I_{j=s_k} P(A_k^c)^2 + \text{higher order term} \tag{3.7.8}$$

$$= 1 - P(A_1^c) - P(A_2^c) + P(A_1^c)P(A_2^c) = P(A_1)P(A_2), \tag{3.7.9}$$

Where the last line follows from the fact the only one $I_{i=s_1}$ and one $I_{j=s_2}$ are not zero for different $i \neq j$. □

Theorem 3.7.2. *Out of the residuals defined in Remark 3.5.2, there are n_{good} number of residuals are i.i.d sampled from the desired distribution $G \overset{(d)}{=} A^*\xi_i + \xi_{i+1}$ with $A = A^*$.*

Proof. First let $\zeta_i = A^*\xi_i + \xi_{i+1}$ for $i = 0 \bmod 2$. To prove this, we only need to build a random index sequence $S = \{s_1, \cdots, s_{n_{good}}\}$ with value s_i taken within the set of kept sample index, and independent of ξ_i, then the above Lemma 3.7.1 applies. Note that $S_{con} = \{i_1, i_2, \cdots, i_k\}$ is sampled indepadent of ξ_i. Let $ALL = \{i \mid i = 0 \bmod 2\}, i \leq n-2$. Then eliminate i from ALL if any of $(i-1, i, i+1, i+2, i+3)$ is showing up in S_{con} to get ALL'. The construction of ALL' only depends on S_{con} and therefore is independent of ξ_i. Also the construction of ALL' guarantees that the indexed sample ζ_i will not be eliminated by our screening and robust detection procudure, and $|ALL'| \geq n_{good}$ by the fact that $|S_{con}| \leq n\alpha$. Finally we keep the n_{good} smallest index in ALL' as $ALL'_{n_{good}}$ to the desired index sequence we want to construct. □

CHAPTER 4

MAKING SGD ROBUST AGAINST OUTLIER ATTACK

4.1 Introduction

Stochastic Gradient Descent (SGD) [143] is arguably the most commonly used algorithm for modern data-driven problem [26]. With this simple form

$$\theta_{t+1} = \arg\min_{\theta \in \mathcal{D}} \left[\eta_t \langle g_t, \theta - \theta_t \rangle + \frac{1}{2} \|\theta - \theta_t\|^2 \right] := \prod_{\theta \in \mathcal{D}} \left[\theta_t - \eta_t g_t \right], \tag{4.1.1}$$

where $\mathcal{D}$ is some convex bounded feasible region of the parameter to be optimized, and g_t is some noisy version of the gradient of some target function $F(\theta)$ at state θ_t. SGD are often easy to implement in practice, and output θ with extraordinary performance both in-sample and out-sample. Also, it is very flexible to be adjusted for different specific problems. There are multiple successful SGD variants, to name a few, conditional gradient descent or Frank-Wolfe algorithm [94, 63] to solving the sparsity related problem, dual averaging [192, 54] and mirror descent [10, 124] for solving online decision making problem on with specific constraint set $\mathcal{D}$, adaptive gradient descent [53] as a approximated second order amend for handling problem that are ill-conditioned, accelerated stochastic gradient descent [127, 11] that achieves the theoretical optimality (matching problem lower bound) [126].

The study of SGD started from a more general framework; stochastic approximation [143]. From there, there are fruitful results regarding SGD in different settings. For example, 1. SGD with different step sizes and averaging schemes achieves optimal solutions with different convex settings. [28] 2. SGD behaves well and finds a good solution with several statistical estimation problems with a non-convex formulation. see e.g. [96, 68, 69] 3. The study of the SGD dynamics of extremely complicated model classes (e.g., neural network and many other overparametrized models) helps us fundamentally understand why several models general-

ized well [52, 163, 37], which is beyond the traditional understanding of statistical learning theory [180] and computational learning theory [101].

Robustness has become one of the primary concerns in the modern machine learning task [78]. Typical robust concerns include the following two topics. i) Our algorithm is relying too much on the data points seen. This problem is always called overfitting [84]. The measurement of the performance of the output of our algorithm on the entire data population (out-of-sample) and the performance on the training dataset (in-sample) is called generalization [119]. ii) The second robust concern is that our algorithms of decision making and inference rely too much on the model [92]. Still, the model can be miss-specified, which can cause our algorithms to 'overfit' the model forgetting that model is always an approximation/simplification of the real-world problem. In this paper, we mainly study the second issue.

Two significant amendments to the second issue mentioned above are i) making weak model assumptions. In the literature, statisticians achieve this by studying the problem with only finite (usually 2nd or 4th) moment assumption [141], rather than the specific distributional assumption, e.g., normal distribution. ii) The second line of works focuses on Huber's contamination model [91], in which model we assume the majority of data are coming from a distribution that is of interest, and a small portion of data is from another arbitrary distribution, i.e.,

$$X \sim (1-\epsilon)P + \epsilon Q. \tag{4.1.2}$$

A closely related model recently received much attention is the adversarial contamination model, in which we get data X_i with probability $1-\epsilon$ from P, with probability ϵ, the returned data X_i can be maliciously designed (e.g., adaptive to your algorithm based on the previous data given). This adversarial model strictly includes Huber's contamination model. We are going to study the first-order method within the adversarial contamination setting.

Related work In [136], the authors consider the optimization problem in 1) Huber ϵ-contamination model and 2) heavy tail distribution model. They also take the high-dimensional issue into account. Mean estimation in high-dimension is a fundamental technique in their proposed algorithm. While in the SGD framework, they address the issue of efficiently obtaining a gradient estimation in each step. Moreover, they don't assume the homogeneous bound on the noise term on different locations in the feasible region $\mathcal{D}$ as we do. Using recent statistics and probability results about robust mean estimation, leveraging the mini-batch sample, they estimate the gradient adaptively and accurately. However, their techniques only work for strongly convex and smoothed functions, which doesn't apply to all of the problems that are of interest. Also, we should mention that this plug-in estimator approach and their analysis cannot be applied to study streaming algorithms since the study essentially solves an independent estimation at each step using a batch of fresh samples and applied inexact gradient descent analysis neglect the concentration effect of uncontaminated sample.

It comes to our attention that another line of works, Byzantine tolerance optimization, which consider a problem closely related with Huber ϵ-contamination model [34] [136] [118] [193] [166] [194]. The typical setting is that a canter machine receives noisy gradient from distributed gradient machines at each step, but the ϵ fraction of the gradient machines are malicious (Byzantine workers). While in our work, in the adversarial ϵ-contamination model, we only have one streaming of data. The contamination can induce an intrinsic bias into our object [33], but in Byzantine problem, the general intuition (and technique) is that we have multiple streamings of data. All good machines will eventually behave relatively similarly, so we can use them as good references to rule out the bad machines, which is not the case in our setting.

The most related work would be [178]. In this work, the authors study the online optimization problem (which is closely related to stochastic Lipchitz convex optimization) with

k number of contaminated samples. While their algorithm needs to know the number of contaminated samples and only applies to Lipchitz convex function with our stochastic setting, we don't need to know the number of contaminated samples a priori. Also, our approach and analysis can be applied to other settings, e.g., smoothed convex optimization and several variants of SGD, e.g., proximal gradient descent and mirror descent.

4.1.1 Problem Setup and Notation

We start by formally stating the problem. We consider a stochastic optimization problem with potentially contaminated data. Our goal is to optimize the following object, for $f : \Theta \times \mathcal{Z} \to \mathbb{R}$,

$$\min_{\theta \in \mathcal{D}} \{F(\theta) = \mathbb{E}_{Z \sim P}[f(\theta, Z)]\}. \tag{4.1.3}$$

Here $\mathcal{Z} \in \mathbb{R}^p$, $\Theta \in \mathbb{R}^d$, and P are the data generating distribution that is of interest. Suppose that each time we can sample a Z_i from a **contaminated Oracle**, which is, each round i, the oracle sample independent data from $X_i \sim P$ and also design a sample Y_i, which can be adversarial (i.e., design Y_i based on historical X_i, Y_i sampled and designed before and the current status of our algorithm θ_i), the oracle then flip a biased coin $c_i \in \{0, 1\}$ independently $\sim \ Ber(1-\epsilon)$, with probability $1-\epsilon$, the Oracle returns the random sample $Z_i = X_i$, with probability $\epsilon \leq 1/8$, the Oracle returns $Z_i = Y_i$. We assume that under the target distribution P, we have the following property.

Assumption 1 (norm-subGaussian under distribution P)**.** For any $\theta \in \mathcal{D}$, and $\lambda \in \mathbb{R}$, we have the following norm-subGaussian property with parameter G assumption, let $X \sim P$,

$$\begin{aligned} &\mathbf{E}[\nabla f(\theta, X)] = \nabla F(\theta), \\ &P\left(\|\nabla f(\theta, X) - \nabla F(\theta)\| \geq t\right) \leq e^{-\frac{t^2}{2G^2}}. \end{aligned} \tag{4.1.4}$$

Here $\nabla f(\theta, Z)$ is the partial derivative taken with respect to θ.

Here we say a scaler random variable X and its probability distribution is subGaussian(G), if

$$P(|X - \mathbb{E}X| \geq t) \leq e^{-\frac{t^2}{2G^2}}, \tag{4.1.5}$$

and we say a random vector X and its probability distribution is norm-subGaussian(G), if

$$P(\|X - \mathbb{E}X\| \geq t) \leq e^{-\frac{t^2}{2G^2}}. \tag{4.1.6}$$

We note that here are other equivalent (up to a constant) definitions of subGaussian and norm-subGaussian random variable. see e.g. [182, Lemma 5.5] and [97, Lemma 2]. Specifically, we are going to use the moment characterization to justify certain truncated random vector still preserves the norm-subGaussianity. Also, by definitions above and the Cauchy-Schwarz inequality, given $\theta \leq R$, $x \sim$ norm-subGaussian(G), we have

$$\langle x, \theta \rangle \sim subGaussian(GR). \tag{4.1.7}$$

Remark 4.1.1. Note that a stronger bounded gradient deviation assumption is made in other outlier robust first-order streaming optimization works, see e.g. [3], [178]. For which, we can effectively take an additional logarithmic factor $G = G\sqrt{\log(T/\delta)}$ to make a bounded contaminated Oracle in T calls as in their work with chance greater than $1 - \delta$. However, it seems that in their work, a bounded gradient Oracle is necessary for the streaming algorithm to get rid of a non-vanishing logarithmic factor $\log(T)$. We will discuss this issue in detail in the later section §16. Also, we will propose a fix of the non-vanishing $\log T$ by introducing a truncation technique.

Remark 4.1.2. One may ask whether we can use first-order gradient Oracle to replace our

sample Oracle, the answer is no if we want the algorithm to be in a streaming fashion. The intuition is that we need to use historical data $g(\theta_{new}, Z_{old})$ to estimate $\nabla F(\theta_{new})$, a Lipchitz continuity and radius dependent additional error $\|\nabla F(\theta_{old}) - \nabla F(\theta_{new})\| \leq MR$ will show up.

Definition 1. A function f is M-smooth if the function f is differentiable and has Lipchitz gradient

$$\|\nabla f(x) - \nabla f(y)\| \leq M\|x - y\|. \tag{4.1.8}$$

4.2 Main Result

The main contribution of the paper is that we prove that, for Problem 4.1.3 with Assumption 1 and some further assumptions on the sample Oracle, with $M-$smooth target function F, there is a streaming SGD algorithm in $O(T)$ sample oracle calls, outputs $\bar{\theta}$ achieves the following convergence rate,

$$F(\bar{\theta}) - F^* \lesssim \frac{MR^2}{T\eta} + GR\left(\sqrt{\frac{\log(T/\delta)}{T}} + \epsilon\sqrt{\log(1/\epsilon)}\right). \tag{4.2.1}$$

The first term, $MR^2/T\eta$, is known very well for the optimization rate of the problem [28], i.e., rate noiseless and contamination-free case. The term $GR/\sqrt{T}$ is known matching the statistical lower bound of the problem, see, e.g., [125] and [122, Proposition 14.1.1], i.e., the contamination-free case. Finally, the term ϵGR is known to be unavoidable under Huber's contamination model; for detail, see § 4.4. Therefore, for SGD based algorithm, we know this cannot be fundamentally improved.

4.3 Main Theory

4.3.1 Useful Lemmas and Tools

The next lemma is going to be useful to prove the norm-subGaussianity of random variable.

Lemma 4.3.1. *(Conditional independence)* Let $\mathcal{F}$ be the σ-field of random variable Z, i.e. $\sigma(Z)$, random variable $X \in \mathcal{F}$, then we know $X = h(Z)$ for some measure function $h \in \mathcal{F}$. If random variable $Y \perp \mathcal{F}$, and function ϕ of X and Y is integrable $\mathbb{E}[|\phi(X, Y)|] < \infty$, then we have,

$$\mathbb{E}[\phi(X, Y) \mid \mathcal{F}] = g(X), \tag{4.3.1}$$

where $g(x) = \mathbb{E}[\phi(x, Y)]$.

Proof. Denote $\psi(z, y) = \phi(h(z), y)$. Then by [55, Example 4.1.7], we know

$$\mathbb{E}[\psi(Z, Y) \mid \mathcal{F}] = j(Z) = g(h(Z)) = g(X). \tag{4.3.2}$$

Here $j(z) = \mathbb{E}[\psi(z, Y)] = \mathbb{E}[\phi(h(z), Y)] = g(h(z))$. □

We first present two lemmas about the deviation bound on the good sample. We also provide a useful, robust median and screening algorithm based on known noise level G, which effectively detects outliers beyond $\Omega(G)$ and estimates the mean within $O(G)$.

Lemma 4.3.2 (Sample radius)**.** Under Assumption 1, for any fixed $\theta_1, \theta_2, \cdots, \theta_k$ selected, we sample $Z_1.Z_2, \ldots, Z_m \sim P$. Then with probability at least $1 - \delta$, we have

$$\max_{i \in [m], j \in [k]} \left\| \nabla f(\theta_j, Z_i) - \nabla F(\theta_j) \right\| \leq c_1 G \sqrt{\log(m) + \log(k) + \log(1/\delta)}. \tag{4.3.3}$$

This is the norm-subGaussian tail bound with a union bound (take $\delta = \delta/(mk)$ on each event) result.

Lemma 4.3.3 (Sample sum)**.** Under Assumption 1, for any fixed $\theta_1, \theta_2, \cdots, \theta_k$ selected, we sample $Z_1.Z_2, \ldots, Z_m \sim P$. Then with probability at least $1 - \delta$, we have

$$\max_{j \in [k]} \left\| \sum_{i=1}^{m} \left[\nabla f(\theta_j, Z_i) - \nabla F(\theta_j) \right] \right\| \leq c_2 G \sqrt{m \left[\log(k) + \log(1/\delta) + \log(2d) \right]}. \tag{4.3.4}$$

The proof is by [97, Corollary 7].

Remark 4.3.1. The condition of the above two lemmas can be relaxed to selecting $\theta_1, \cdots, \theta_k$ independent of $Z_1, \cdots, Z_m$.

This is a simple union bound over k event with concentration results on the norm-subGaussian random variable. For the proof, see [97, Corollary 7]. Then apply the same argument with Lemma 4.3.1. The following two algorithms give a robust screening procedure Algorithm 1 and a robust median as a rough estimation of the mean Algorithm 2. These techniques appear in [3].

Algorithm 4.1: Robust screening algorithm

Robust-Screen;
Input : Data sample $S = \{Z_1, Z_2, \cdots, Z_m\}$, Radius G
Output: Trust set S'
shuffle sample set
i = 1
repeat
  $i \leftarrow i + 1$
until $\left|\left\{j \mid \|Z_i - Z_j\| \leq 2G\right\}\right| \geq m/2$;
$S' = \left\{Z_i \in S \mid \|Z_i - Z_j\| \leq 4G\right\}$

Algorithm 4.2: Robust median as robust mean estimation algorithm

Robust-Median;
Input : Data sample $S = \{Z_1, Z_2, \cdots, Z_m\}$, Radius G
Output: Robust Median Z
$S' =$ Robust-Screen(S, G)
$Z =$ select any sample in S'

Lemma 4.3.4 (Robust trick)**.** When the input dataset S, has more than half of the data satisfies $\|Z_i - \mu\| \leq G$, Algorithm 1 returns S' satisfies,

$$\begin{aligned} &\forall Z' \in S', \quad \|Z' - \mu\| \leq 7G, \\ &\forall Z \in S, \text{ such that } \|Z - \mu\| \leq G, \text{ will be included } Z \in S'. \end{aligned} \tag{4.3.5}$$

Also Algorithm 2 returns Z' satisfies,

$$\|Z' - \mu\| \leq 7G. \tag{4.3.6}$$

Proof. For data point $Z_1, Z_2, \ldots, Z_n \in S$, if we know that at least half of the point are in the ball $B(\mu, G)$. Given any Z_i such that $|\{j \mid \|Z_i - Z_j\| \leq 2G\}| \geq n/2$, we have

$$\|Z_i - \mu\| \leq 3G. \tag{4.3.7}$$

Otherwise, there won't be many Z's satisfies the condition. Let $\mu_2 = Z_i$, we have $B(\mu, G) \subseteq B(\mu_2, 4G)$. By the triangle inequality, we have for any point Z within $B(\mu_2, 4G)$, we have $\|Z - \mu\| \leq 7G$. □

4.3.2 Basic Online Algorithm

We start by studying a simple streaming SGD with a robust screening algorithm. Each time we get a sample Z from the contaminated Oracle, we build a confidence interval based on the historical data and drop the incoming sample if we are confident that the current sample is contaminated. Otherwise, we perform gradient descent based on the new sample. The algorithm can be formally stated as follows.

Hereafter, we denote $k(t)$ be the value of K in the algorithm at t-th round of the algorithm, i.e. $\theta_t = \theta'_{k(t)}$. As we will show, with appropriate V, with high probability, we can guarantee

Algorithm 4.3: Basic version of RSGD algorithm

Basic-RSGD;
Input : Initial point $\theta_1 = \theta'_1$, maximum number of iteration T, pre-sample size m, confidence region radius V, step-size η_t
Output: $\bar{\theta}$
$K \leftarrow 1$
Sample $S = \{Z_1, Z_2, \ldots, Z_m\}$;
for $t \leftarrow 1$ **to** T **do**
  $\theta_t = \theta'_K$
  Compute $\mu(\theta_t) = \text{Robust-Median}\left(\left\{\nabla f(\theta_t, Z_i)_{Z_i \in S}\right\}\right)$;
  Query a sample $\tilde{Z}_t$ fron Oracle;
  if $\|\nabla f(\theta'_K, \tilde{Z}_t) - \mu(\theta'_K))\| \leq V$ **then**
    $\theta'_{K+1} \leftarrow \prod_{\mathcal{D}}[\theta'_K - \eta_K \nabla f(\theta'_K, \tilde{Z}_t)]$;
    $S \leftarrow S \bigcup \{Z_{m+K} = \tilde{Z}_t\}$
    $K \leftarrow K + 1$
$\bar{\theta} = \text{Mean}\left(\left\{\theta'_1, \cdots, \theta'_K\right\}\right)$

that all of the good samples will be accepted. For the rest of this manuscript, we call a step i good if we get a sample from P, i.e., $\tilde{Z}_i = X_i \sim P$, and bad otherwise, and we call a sample Z good if it is from $X \sim P$, and bad otherwise. We have the following bound for the radius of gradient spawn by good samples.

Lemma 4.3.5 (Radius of good sample)**.** Under Assumption 1, if function f is M-smooth in θ, suppose $G/M \leq R$, for any $\theta \in \mathcal{D}$, for $Z_1.Z_2, \ldots, Z_m \sim P$, with probability at least $1 - \delta$, we have

$$\max_{i \in [m]} \|\nabla f(\theta, Z_i) - \nabla F(\theta)\| \leq \min_{R \geq \tau > 0} \left(2M\tau + c_1 G\sqrt{d\log(2R/\tau) + \log(m/\delta)}\right) \tag{4.3.8}$$

$$= \min_{R \geq \tau > 0} \left(2M\tau + c_1 G\sqrt{\log(Query_1) + \log(m/\delta)}\right) \tag{4.3.9}$$

$$\leq G\left(2 + c_1\sqrt{d\log(2RM/G) + \log(m/\delta)}\right) \doteq D_1(m, \delta). \tag{4.3.10}$$

$$\text{(taking } \tau = G/M.\text{)} \tag{4.3.11}$$

Proof. The argument is based on a covering results and a union bound. First, we construct a τ-net $N(\mathcal{D}, \tau, \|\cdot\|)$ of the set $\mathcal{D}$. By and fact that $\mathcal{D}$ is bounded by R and [182, Lemma 5.2], we know for $\tau \leq R$,

$$|N(\mathcal{D}, \tau, \|\cdot\|)| \leq \left(\frac{2R}{\tau}\right)^d := Query_1. \tag{4.3.12}$$

Then, by a union bound on m data point and the $|N(\mathcal{D}, \tau, \|\cdot\|)|$ along with tail Assumption 1, using Lemma 4.3.2 we have, with probability greater than $1-\delta$, for any $\theta \in N(\mathcal{D}, \tau, \|\cdot\|)$

$$\max_{i\in[m]} \|\nabla f(\theta, Z_i) - \nabla F(\theta)\| \leq c_1 G\sqrt{d\log(R/\tau) + \log m + \log(1/\delta)}. \tag{4.3.13}$$

Suppose the above event is true. Then for any $\theta \in \mathcal{D}$, we can find $\theta' \in N(\mathcal{D}, \tau, \|\cdot\|)$ such that $\|\theta - \theta'\| \leq \tau$. By the Lipchitz continuity of f and F, we have

$$\max_{i\in[m]} \|\nabla f(\theta, Z_i) - \nabla F(\theta)\| \leq 2M\tau + c_1 G\sqrt{d\log(R/\tau) + \log m + \log(1/\delta)}. \tag{4.3.14}$$

□

Next, we get statistical accuracy of $\mu(\theta_t)$ based on empirical process theory to determine the deviation $\|\mu(\theta_t) - \nabla F(\theta_t)\|$ which in turn will decide the confidence radius V in Algorithm 3 such that our algorithm will accept all of the good samples with a decent chance. The bad sample wouldn't mislead us too much.

A simple tail bound of Bernoulli random variable with a uniform bound along T step. We can take the pre-sample size m greater than $16\log(T/\delta)$ to guarantee that for each step t, the good sample portion along the steps of the entire algorithm is always greater than $3/4$, which will make the Robust-Screen 1 and Robust-Median 2 viable.

Corollary 4.3.1. With input pre sample size $m = 16\log(T/\delta)$, $V = V_1 := 7D_1(m + T, \delta/T) + G\sqrt{\log(T/\delta)}$ in Algorithm 3, with probability at least $1 - 3\delta$ the algorithm will

accept all of the sample from P. Also, with probability greater than $1-\delta$ all of the good sample satisfies $\left\|\nabla f(\theta'_{k(t)}, \tilde{Z}_t) - \nabla F(\theta'_{k(t)})\right\| \leq G\sqrt{\log(T/\delta)}$.

To see this, fist with probability greater than $1-\delta$ we know S always has more than half of the data that are good, and with probability greater than $1-\delta$ we known $\|\mu(\theta_t) - \theta_t\| \leq 7D_1(m+T, \delta/T)$ using Lemma 4.3.4 and the good sample radius bound Lemma 4.3.5. The second term $G\sqrt{\log(T/\delta)}$ is from the norm-subGaussian deviation together with a union bound with at most T good samples to guarantee $\left(I_{\tilde{Z}_t \text{ is } good} \cdot \|\nabla f(\theta'_{k(t)}, \tilde{Z}_t) - \nabla F(\theta'_{k(t)})\|\right) \leq G\sqrt{\log(T/\delta)}$, which holds with probability greater than $1-\delta/T$. Moreover, we have the deviation guarantee for the bad sample as follows.

Remark 4.3.2. Here, using Robust-Mean Algorithm 4 wouldn't help too much, since we will still have a $O(G\sqrt{\log Query_1/m})$ in V.

Remark 4.3.3. Here the first argument $m+T$ in D_1 term eventually becomes a $\log(m+T)$ term in analyzing the algorithms (an upper bound of good sample we get from pre-sample and SGD process sample) in our final bound of study the SGD algorithm. This seems unreasonable since a better sample size will give us a worse result. Later in section § 16, we will show we can eliminate these $\log(m+T)$ terms by a simple truncation rule and only induce an $\epsilon\sqrt{\log(1/\epsilon)}G$ bias. i.e. substitute the $\sqrt{\log(m+T)}$ by $\epsilon\sqrt{\log(1/\epsilon)}$. For the latter part of presenting the result in this section, we will write these logarithmic terms regarding the deviation of bad samples in a Big-O term.

Corollary 4.3.2. With same setting as in the above corollary, in Algorithm 3, with probability at least $1-3\delta$, the following holds for all accepted gradient $\nabla f(\theta'_{k(t)}, Z_{m+k(t)})$,

$$\|\nabla f(\theta'_{k(t)}, Z_{m+k(t)}) - \nabla F(\theta'_{k(t)})\| \leq 14D_1(m+T, \delta/T) + G\sqrt{\log(T/\delta)} := \tilde{V}_1. \quad (4.3.15)$$

This is a simple result of triangle inequality (insert $\mu(\theta_t)$ in).

Remark 4.3.4. This bound is suboptimal in the sense that we need to take a union bound over effectively $Query_1$ number of points. It's tempting to reduce the $Query_1$ to just T point along the SGD training trajectory. However, this approach is incorrect, because the point $\theta_{k(t)}$ is actually a function of θ_1 and all of the history data $Z_1, \cdots, Z_{t-1}$, which makes the **independent** or **fixed** condition as in Lemma 4.3.2 or Lemma 4.3.3 no longer hold. The fact that the trajectory $\theta_{k(t)}$ is **dependent** on the historical data makes we need to take a large union bound, i.e., $Query_1$.

Optimization Convergence

In this section, we give the framework of analyzing general stochastic gradient descent based on inexact gradient analysis [42, 28], and analyze Algorithm 3 using the framework. By studying this algorithm, we get the important steps for bounding deviation which lead to improvements for the algorithm. For notational simplicity, let $g_k = \nabla f(\theta_k, Z_{m+k})$ be the descent step, and $e'_k = \nabla F(\theta'_k) - g_k$.

Proposition 4.3.1. For the projection descent
$\theta_{k+1} = \arg\min_{y \in \mathcal{D}} \left[\eta_k \langle g_k, y\rangle + \frac{1}{2}\|y - \theta_k\|^2\right]$, for any $z \in \mathcal{D}$, the following holds,

$$\langle g_k, \theta_{k+1} - z\rangle \leq \frac{1}{2\eta_k}(\|\theta_k - z\|^2 - \|\theta_{k+1} - z\|^2 - \|\theta_k - \theta_{k+1}\|^2).$$

This is a standard results from the first-order optimality condition of gradient step, for the proof see [28].

Lemma 4.3.6 (Inexact Gradient)**.** For Problem 4.1.3 with M-smooth function $F(\theta)$, under Assumption 1, with input step size $\eta_t \leq 1/M$, for each θ_t in the trajectory of the gradient

descent

$$F(\theta'_k) - F^* \leq \frac{1}{2\eta_k}\left(\|\theta'_k - \theta^*\|^2 - \|\theta'_{k+1} - \theta^*\|^2\right) + \langle e'_k, \theta'_k - \theta^*\rangle + \frac{\eta_k}{2(1 - M\eta_k)}\|e'_k\|^2. \tag{4.3.16}$$

Proof. Suppose $\theta^* \in \arg\min_{\theta\in\mathcal{D}} F(\theta)$. From the smoothness of function F, we have

$$F(\theta'_{k+1}) \leq F(\theta'_k) + \langle \nabla F(\theta'_k), \theta'_{k+1} - \theta'_k\rangle + \frac{M}{2}\|\theta'_{k+1} - \theta'_k\|^2 \quad \text{(smoothness of F)} \tag{4.3.17}$$

$$\leq F(\theta^*) + \langle \nabla F(\theta'_k), \theta'_{k+1} - \theta^*\rangle + \frac{M}{2}\|\theta'_{k+1} - \theta'_k\|^2 \quad \text{(convexity)} \tag{4.3.18}$$

$$= F^* + \langle g_k, \theta'_{k+1} - \theta^*\rangle + \langle e'_k, \theta'_{k+1} - \theta^*\rangle + \frac{M}{2}\|\theta'_{k+1} - \theta'_k\|^2 \quad \text{(definition of } e'_k\text{)} \tag{4.3.19}$$

$$\leq F^* + \frac{1}{2\eta_k}\left(\|\theta'_k - \theta^*\|^2 - \|\theta'_{k+1} - \theta^*\|^2 - \|\theta'_k - \theta'_{k+1}\|^2\right) + \langle e'_k, \theta'_{k+1} - \theta^*\rangle \tag{4.3.20}$$

$$+ \frac{M}{2}\|\theta'_{k+1} - \theta'_k\|^2 \tag{4.3.21}$$

$$\leq F^* + \frac{1}{2\eta_k}\left(\|\theta'_k - \theta^*\|^2 - \|\theta'_{k+1} - \theta^*\|^2 - \|\theta'_k - \theta'_{k+1}\|^2\right) + \langle e'_k, \theta'_k - \theta^*\rangle + \tag{4.3.22}$$

$$\|e'_k\| \cdot \|\theta'_{k+1} - \theta'_k\| + \frac{M}{2}\|\theta'_{k+1} - \theta'_k\|^2 \tag{4.3.23}$$

$$= F^* + \frac{1}{2\eta_k}\left(\|\theta'_k - \theta^*\|^2 - \|\theta'_{k+1} - \theta^*\|^2\right) + \langle e'_k, \theta'_k - \theta^*\rangle + \|e'_k\| \cdot \|\theta'_{k+1} - \theta'_k\| \tag{4.3.24}$$

$$+ \frac{M\eta_k - 1}{2\eta_k}\|\theta'_{k+1} - \theta'_k\|^2 \tag{4.3.25}$$

$$\leq F^* + \frac{1}{2\eta_k}\left(\|\theta'_k - \theta^*\|^2 - \|\theta'_{k+1} - \theta^*\|^2\right) + \langle e'_k, \theta'_k - \theta^*\rangle \tag{4.3.26}$$

$$+ \frac{\eta_k}{2(1 - M\eta_k)}\|e'_k\|^2. \tag{4.3.27}$$

The inequality on the fourth line is from Proposition 4.3.1, and the last inequality follows

from maximizing the last two terms as a quadratic function in $\|\theta'_{k+1} - \theta'_k\|^2$.

□

Theorem 4.3.1. For Problem 4.1.3 with M-smooth function $f(\cdot, Z)$ for any Z, under Assumption 1, running Algorithm 3 with setting step size $\eta_t = \eta \leq 1/M$, pre-sample size $m = 16\log(T/\delta)$, confidence radius $\tilde{V} = V_1$ specified in Corollary 4.3.1, with probability greater than $1 - 4\delta$, the output $\bar{\theta}$ satisfies

$$F(\bar{\theta}) - F^* \lesssim \frac{1}{K\eta}\|\theta_1 - \theta^*\|^2 + GR\frac{\sqrt{T\log(1/\delta)}}{K} + \frac{|\mathcal{S}_{bad}|}{K}\tilde{V}_1 R + \eta G^2\log(T/\delta) + \eta\frac{|\mathcal{S}_{bad}|}{K}\tilde{V}_1^2. \tag{4.3.28}$$

Here we let $[K] = \mathcal{S}_{good} \bigcup \mathcal{S}_{bad}$ corresponding to the accepted sample Z_{m+k} is an independent sample from P or not, and $\tilde{V}_1$ is defined in Corollary 4.3.2.

Proof. Let $k(t)$ denotes the k value at t round of Algorithm 3, i.e. $\theta_t = \theta'_{k(t)}$. Let $\mathcal{F}_1 = \sigma(\theta_1)$ and $\mathcal{F}_t$ be the filtration σ field up to t steps, i.e. $\mathcal{F}_t$ be the σ-field including $\mathcal{F}_{t-1}$ and $\sigma(c_{t-1}, X_{t-1}, Y_{t-1})$. Also, we let $d_t = \nabla f(\theta_{k(t)}, \tilde{X}_t) - \nabla F(\theta_{k(t)})$, then $d_t = e_{k(t)}$ if we proceed at t round and t round is good, i.e., $k(t+1) \neq k(t)$. For $k \in \mathcal{S}_{bad}$, by Lemma 4.3.6 and Corollary 4.3.2, we have with probability greater than $1 - 3\delta$

$$\begin{aligned}
(\eta - M\eta^2)\left[F(\theta'_{k+1}) - F^*\right] &\leq \frac{1 - M\eta}{2}\left(\|\theta'_k - \theta^*\|^2 - \|\theta'_{k+1} - \theta^*\|^2\right) && (4.3.29)\\
&\quad + (\eta - M\eta^2)\langle e'_k, \theta'_k - \theta^*\rangle + \frac{\eta^2}{2}\|e'_k\|^2 && (4.3.30)\\
&\leq \frac{1 - M\eta}{2}\left(\|\theta'_k - \theta^*\|^2 - \|\theta'_{k+1} - \theta^*\|^2\right) && (4.3.31)\\
&\quad + (\eta - M\eta^2)\tilde{V}_1 R + \frac{\eta^2}{2}\|e'_k\|^2. && (4.3.32)
\end{aligned}$$

Now, we have the telescope sum, with probability greater than $1-3\delta$

$$\sum_{k=1}^{K}(\eta - M\eta^2)\left[F(\theta'_k) - F^*\right] \leq \frac{1-M\eta}{2}\underbrace{\sum_{k=1}^{K}\left(\|\theta'_k - \theta^*\|^2 - \|\theta'_{k+1} - \theta^*\|^2\right)}_{(I)} \tag{4.3.33}$$

$$+ (\eta - M\eta^2)\underbrace{\sum_{k\in\mathcal{S}_{good}}\langle e'_k, \theta'_k - \theta^*\rangle}_{(II)} \tag{4.3.34}$$

$$+ (\eta - M\eta^2)\underbrace{\sum_{k\in\mathcal{S}_{bad}}\tilde{V}_1 R}_{(III)} \tag{4.3.35}$$

$$+ \frac{\eta^2}{2}\underbrace{\sum_{k\in\mathcal{S}_{good}}\|e'_k\|^2}_{(IV)} + \frac{\eta^2}{2}\underbrace{\sum_{k\in\mathcal{S}_{bad}}\|e'_k\|^2}_{(V)}. \tag{4.3.36}$$

Obviously, we have

$$(I) \leq R^2, \quad (III) \leq |\mathcal{S}_{bad}|\tilde{V}_1 R, \quad (IV) \leq KG^2\log(T/\delta), \quad (V) \leq |\mathcal{S}_{bad}|\tilde{V}_1^2 \tag{4.3.37}$$

We study the following martingale sum to bound (II). Note that we call round t is good if the Oracle returns a good sample $\tilde{Z}_t = \tilde{X}_t \sim P$,

$$(II') = \sum_{t=1}^{T}\langle d_t \cdot I_{t \text{ is } good}, \theta'_{k(t)} - \theta^*\rangle \tag{4.3.38}$$

$$= \sum_{t=1}^{T}\langle[\nabla f(\theta'_{k(t)}, \tilde{X}_t) - \nabla F(\theta'_{k(t)})] \cdot I_{t \text{ is } good}, \theta'_{k(t)} - \theta^*\rangle. \tag{4.3.39}$$

Note that $\theta'_{k(t)} - \theta^* \in \mathcal{F}_t$, and the Oracle's decision of returning good sample or not is independent of $\mathcal{F}_t$, also the good sample if it returns is independent of $\mathcal{F}_t$, the term will be

zero if t is not good. Therefore, $d_t \cdot I_{t \text{ is } good} \perp \mathcal{F}_t$. Then we have

$$\mathbb{E}\left[\langle d_t I_{t \text{ is } good}, \theta'_{k(t)} - \theta^* \rangle \mid \mathcal{F}_t\right] \tag{4.3.40}$$

$$= \langle \mathbb{E}\left[d_t I_{t \text{ is } good}\right], \theta'_{k(t)} - \theta^* \rangle = 0. \tag{4.3.41}$$

Note that $\mathbb{E}[\|d_t I_{t \text{ is } good}\|^k] \leq \mathbb{E}[\|d_t\|^k]$, then $d_t I_{t \text{ is } good}$ is also a norm-subGaussian(G) by Assumption 1, then by (4.1.7) the term in the martingale sum above satisfies the Azuma-Hoeffding concentration inequality [152, Theorem 2]. Then we have the bound, with probability greater than $1 - \delta$,

$$(II') \lesssim GR\sqrt{T \cdot \log(1/\delta)}. \tag{4.3.42}$$

Note that we accept all good samples with a probability greater than $1 - 3\delta$. At this event, we have,

$$(II) = \sum_{k \in \mathcal{S}_{good}} \langle e'_k, \theta_k - \theta^* \rangle = \sum_{t=1}^{T} \langle d_t \cdot I_{t \text{ is } good}, \theta'_{k(t)} - \theta^* \rangle = (II') \tag{4.3.43}$$

Together with a union bound, with probability greater than $1 - (3\delta + \delta) = 1 - 4\delta$, divide both sides by $K(\eta - M\eta^2)$, with $\eta \leq 1/M$, we get the desired result. □

Remark 4.3.5. Now, for T large enough, we know $|\mathcal{S}_{bad}|/T \leq 2\epsilon$ (since we always accept

good sample). Picking $\eta_t = \eta \leq 1/M$, we have

$$F(\bar{\theta}) - F^* \lesssim \frac{1}{T\eta}R^2 + GR\sqrt{\frac{\log(T/\delta)}{T}} + \eta G^2 \log(T/\delta) + \epsilon\eta\tilde{V}_1^2 + \epsilon\tilde{V}_1 R \tag{4.3.44}$$

$$\lesssim \frac{1}{T\eta}R^2 + GR\left(\sqrt{\frac{\log(T/\delta)}{T}} + \epsilon\sqrt{d\log(RM/G)}\right) \tag{4.3.45}$$

$$+ \eta G^2 \left(\log(T/\delta) + \epsilon d \log(RM/G)\right) \tag{4.3.46}$$

$$+ \underbrace{O\left(\epsilon GR \log(T/\delta)\right)}_{\text{can be eliminate by truncation § 16}} \tag{4.3.47}$$

Optimize over $\eta \leq min(1/M, R/G\sqrt{T})$, for T sufficient large, we have

$$F(\bar{\theta}) - F^* \lesssim \frac{MR^2}{T\eta} + GR\left(\sqrt{\frac{\log(T/\delta)}{T}} + \sqrt{\frac{\epsilon d\log(RM/G)}{T}} + \epsilon\sqrt{d\log(RM/G)}\right) \tag{4.3.48}$$

$$+ \underbrace{O\left(\epsilon GR \log(T/\delta)\right)}_{\text{can be eliminate by truncation § 16}} \tag{4.3.49}$$

4.3.3 Double Sequence Algorithm

As mentioned above, the main obstacle is that the screening rule makes data reuse, so we need a union bound on all possible points being visited (potentially all points in $\mathcal{D}$ as $Query_1$ suggests). Therefore, reducing the 'query complexity, i.e., the concentration behavior of good on the **points that are potentially being visited**, is the primary goal of our study.

Assumption 2 (Double Oracle)**.** Suppose we have two Oracles to query from. They cannot communicate, i.e., when they design malicious bad samples, one cannot develop based on the data given by the other Oracle.

It is worth noting that our analysis of the query complexity should be of more general interest. Our analysis technique can be applied to a more general streaming-based algorithm with a robust screening step. To name a few, filtering based algorithm [100], model predictive control [30], stochastic control [5], etc.

Algorithm

Algorithm 4.4: Robust mean estimation

Robust-Mean;
Input : Data sample $Z_1, Z_2, \cdots, Z_m$, Radius G
Output: Robust Mean Z
$S = \text{Robust-Screen}(Z_1, Z_2, \cdots, Z_m, G)$
Replicate arbitrary sample in S multiple times such that $|S| = m$
$Z = \text{Mean}(S)$

Algorithm 4.5: RSGD with double sequences from oracles

Double-RSGD;
Input : Initial point θ_1, maximum number of iteration T, pre-sample size m, confidence region radius V, step-size η_t
Output: $\bar{\theta}$
Step $K \leftarrow 1$
Sample $S_1 = \{Z'_1, Z'_2, \ldots, Z'_m\}$ fron Oracle 1;
for $t \leftarrow 1$ **to** T **do**
    $\theta_t = \theta'_K$
    Compute $\mu(\theta'_K) = \text{Robust-Mean}\left(\left\{\nabla f(\theta'_K, Z'_i)\right\}_{i=1}^m\right)$;
    Query sample $\tilde{Z}_t$ from Oracle 2;
    if $\|\nabla f(\theta'_K, \tilde{Z}_t) - \mu(\theta'_K)\| \leq V$ **then**
        $\theta'_{K+1} \leftarrow \prod_{\mathcal{D}}[\theta'_K - \eta_K \nabla f(\theta'_K, \tilde{Z}_t)]$;
        $Z_K = \tilde{Z}_t$
        $K \leftarrow K + 1$
$\bar{\theta} = \text{Mean}\left(\left\{\theta'_1, \cdots, \theta'_K\right\}\right)$

Statistical Accuracy

Lemma 4.3.7 (Deviation of Robust-Mean)**.** Under Assumption 1, for any fixed $\theta_1, \theta_2, \cdots, \theta_k$ selected, sample $Z_1.Z_2, \ldots, Z_m \sim$ Contaminated Oracle, let $[m] = \mathcal{S}_{good} \bigcup \mathcal{S}_{bad}$, where $\mathcal{S}_{bad}$ include the bad samples and the later replicated samples (the original good one is still in

$\mathcal{S}_{good}$ but the replicated ones are in $\mathcal{S}_{bad}$), if the sample has $m \geq 4\log(1/\delta) := s_1(k)$, denote

$$RM_i = \text{Robust-Mean}\left(\{\nabla f(\theta_i, Z_1), \cdots, \nabla f(\theta_i, Z_m)\}, c_1 G\sqrt{\log(mk/\delta)}\right). \tag{4.3.50}$$

with probability at least $1 - 3\delta$, we have

$$\begin{aligned}
\max_{j\in[k]} \left\|RM_j - \nabla F(\theta_j)\right\| &\leq c_2 G\sqrt{\frac{(\log k + \log(1/\delta) + \log 2d)}{m}} && (4.3.51)\\
&+ \frac{7c_1 G|\mathcal{S}_{bad}|\sqrt{\log(1/\delta) + \log k + \log m}}{m/2} && (4.3.52)\\
&\leq c_2 G\sqrt{\frac{(\log k + \log(1/\delta) + \log 2d)}{m}} && (4.3.53)\\
&+ 7c_1 G\sqrt{\log(1/\delta) + \log k + \log m} && (4.3.54)\\
&:= D_{1.5}(m, k, \delta). && (4.3.55)
\end{aligned}$$

Proof. With m specified as in the Lemma, with probability greater than $1 - \delta$, we have that over $m/2$ number of Z_i', $i \leq m$ are good. Then, by Lemma 4.3.2, we know for all $i \in \mathcal{S}_{good}$, for any θ_j, with probability greater than $1 - \delta$

$$\left\|\nabla f(\theta_j, Z_i) - \nabla F(\theta_j)\right\| \leq c_1 G\sqrt{\log k + \log m + \log(1/\delta)} := R_g. \tag{4.3.56}$$

By Lemma 4.3.3, we have, for ant θ_j, with probability greater than $1 - \delta$

$$\left\|\sum_{i\in\mathcal{S}_{good}} \left[\nabla f(\theta_j, Z_i) - \nabla F(\theta_j)\right]\right\| \leq c_2 G\sqrt{m\left[\log k + \log(1/\delta) + \log 2d\right]}. \tag{4.3.57}$$

Also, with Lemma 4.3.4, we know that Robust-Screen step will include all of the good sample, and the bad sample Z_i kept has the property,

$$\left\|\nabla f(\theta_j, Z_i) - \nabla F(\theta_j)\right\| \leq 7R_g, \tag{4.3.58}$$

for all θ_j. Then using the sum concentration bound on good and deviation bound on bad along with the fact that all $|\mathcal{S}_{good}| > m/2$ number of good are kept, we have the desired result. □

Remark 4.3.6. Still, as mentioned in Remark 4.3.3, the $\log m$ term can be eliminated with a technique in § 16.

Now let's consider a complete tree to characterize the trajectory complexity. Let θ_1 be the root node $\theta_{1,1}$. We spawn new nodes by depth inductively. Each time suppose we have nodes up to depth t, for each node $\theta_{t,i}$ we sample $Z_{t,i}$ from Oracle 2, we spawn two children $\theta_{t+1,2i-1}, \theta_{t+1,2i}$, corresponding to staying at $\theta_{t,i} = \theta_{t+1,2i-1}$ and performing a projected gradient step $\theta_{t+1,2i}$. The reason we construct this complete tree is that our algorithm's trajectory θ_t will for sure follow a root to leaf path (actually, we need to construct a tedious probability coupling of the Oracle 2 spawning this tree and the one we are sampling in the algorithm, we omit the details here). Note that the construction of the tree is completely independent of Oracle 1. Because of the independence, when we control the sample deviation of Z' from Oracle 1 acting on the nodes in the tree, we can put a uniform bound by counting the number of nodes in the tree.

Lemma 4.3.8 (Query capacity)**.** Given a sequence of calls by Oracles 2, the number of nodes in the tree at depth t has the following upper bound,

$$N(\theta_t) = 2^t := \text{Query}_2(t).$$

Also the $\theta'_{k(t)} = \theta_t$ in our Algorithm 5 will be one of the node at depth t of the tree.

Corollary 4.3.3 (Accuracy of estimation)**.** Under Assumption 1 and 2, for $\mu(\theta'_{k(t)})$ in Algorithm 5 with pre-sample set size $m \geq 16 \log(1/\delta)$, we have, with probability greater than

$1 - 3\delta$,

$$\|\mu(\theta'_k) - \nabla F(\theta'_k)\| \leq D_{1.5}(m, \text{Query}_2(t), \delta/T) \leq D_{1.5}(m, \text{Query}_2(T), \delta/T) := V_{1.5}. \tag{4.3.59}$$

With m specified as in the above corollary, apply $N(\theta'_{k(t)})$ as k in Lemma 4.3.7, which holds with probability greater than $1 - 3\delta$.

Corollary 4.3.4. Under Assumption 1 and 2, if we select $V = V_{1.5} + G\sqrt{\log(T/\delta)}$ in Algorithm 5, then the algorithm will accept and proceed on all good sample given by Oracle 2. Also, for the accepted sample, we have

$$\|\nabla f(\theta'_k, Z_k) - \nabla F(\theta'_k)\| \leq 2V_{1.5} + G\sqrt{\log(T/\delta)} := \tilde{V}_{1.5}. \tag{4.3.60}$$

Also, all of the good sample satisfies $\left\|\nabla f(\theta'_{k(t)}, \tilde{Z}_t) - \nabla F(\theta'_{k(t)})\right\| \leq G\sqrt{\log(T/\delta)}$.

Remark 4.3.7. From the discussion and lemma above, we can see using the T number of pre-samples from Oracle 1. The deviation of Robust-Mean is well controlled on the tree constructed up to T depth (neglecting a $\log T$ factor). Suppose at $T + 1$ round of the algorithm we reached θ_{new}. If we gather T more samples from Oracle 1, the deviation of the next T steps starting from θ_{new} are well controlled (Think of θ_{new} replacing θ_0 and restart the algorithm to run T more rounds). Notice that this collection of T new samples from Oracle 1 doesn't have to be done in batch when we reach θ_{new}; it can be gathered in a steaming fashion. Then we have the following Algorithm 6. Note that at the beginning of new stage (new L rounds), the start point $\theta_{k(Lt+1)}$ is constructed totally independent of the good samples in new S_{test} (we left S_{next} in the previous stage totally intact).

We have the following corresponding deviation guarantee for Algorithm 6.

Corollary 4.3.5. Under Assumption 1 and 2, for $\mu(\theta_t)$ in Algorithm 6 with stage period

$L \geq 16\log(T/\delta)$, we have, with probability greater than $1 - 3\delta$,

$$\|\mu(\theta'_k) - \nabla F(\theta'_k)\| \leq D_{1.5}(E \cdot L, \text{Query}_2(L), \delta/T) := V_2. \tag{4.3.61}$$

Also, if we select $V = V_2 + G\sqrt{\log(T/\delta)}$ in, then the algorithm will accept and proceed on all good sample given by Oracle 2. Also, for the accepted sample, we have

$$\|\nabla f(\theta'_k, Z_k) - \nabla F(\theta'_k)\| \leq 2V_2 + G\sqrt{\log(T/\delta)} := \tilde{V}_2. \tag{4.3.62}$$

Algorithm 4.6: RSGD with linear sample query complexity

Linear-RSGD;
Input : Initial point θ_1, maximum number of iteration T, period size L, estimate size E, confidence region radius V, step-size η_t
Output: $\bar{\theta}$
Step $K \leftarrow 1$
Sample $S_{test} = \{Z'_1, Z'_2, \ldots, Z'_{EL}\}$ from Oracle 1;
for $t \leftarrow 1$ **to** T **do**
    $\theta_t = \theta'_K$
    Sample E samples from Oracle 1 and add in the next set:
     $S_{next} \leftarrow S_{next} \bigcup \{Z'_{t,1}, \cdots, Z'_{t,E}\}$
    Compute $\mu(\theta'_K) = \text{Robust} - \text{Mean}\left(\{\nabla f(\theta'_K, Z') \mid Z' \in S_{test}\}\right)$;
    Sample $\tilde{Z}_t$ from Oracle 2;
    if $\|\nabla f(\theta'_K, \tilde{Z}_t) - \mu(\theta'_K)\| \leq V$ **then**
        $\theta'_{K+1} \leftarrow \prod_{\mathcal{D}}[\theta'_K - \eta_K \nabla f(\theta'_K, \tilde{Z}_t)]$;
        $Z_K = \tilde{Z}_t$
        $K \leftarrow K + 1$
    if $t = 0 \, mod \, L$ **then**
        $S_{test} \leftarrow S_{next}$
        $S_{next} \leftarrow \emptyset$
$\bar{\theta} = \text{Mean}\left(\{\theta'_1, \cdots, \theta'_K\}\right)$

Improved Analysis by Truncation

Here we briefly discuss how to eliminate the $\log T$ term, i.e., the $\log(m)$ or $\log(m+T)$ in the previous Lemma 4.3.2, Lemma 4.3.5 and Lemma 4.3.7 as in the bound of good gradient deviation radius. Since this is a pure technical trick, which is not the main focus of this paper, we will briefly go through how to adapt the proof from previous sections.

Proposition 4.3.2. For $Z \sim$ norm-subGaussian(G), $0 < \epsilon < 1/8$, there exists $C_s G$, such that the following arguments hold:

$$\mathbb{E}\left[\|Z - \mathbb{E}Z\| \cdot I_{\{\|Z-\mathbb{E}Z\| \geq C_s G\}}\right] \leq \epsilon\left(2 + \sqrt{2\log(2/\epsilon)}\right) G \leq c_3 \epsilon \sqrt{\log(1/\epsilon)} G, \text{ and} \tag{4.3.63}$$

$$P(\|Z - \mathbb{E}Z\| \geq C_s G) \leq \epsilon. \tag{4.3.64}$$

Here $C_s = \sqrt{2\log(2/\epsilon)}$.

Proof. By a scaling argument, WLOG, we assume $G = 1$. Denote

$$X = \|Z - \mathbb{E}Z\| I_{\|Z-\mathbb{E}Z\| \geq C_s}. \tag{4.3.65}$$

We have, for any $t > 0$,

$$P(X \neq 0) \leq P(\|Z - \mathbb{E}Z\| \geq C_s) \leq 2e^{-\frac{C_s^2}{2}} \leq \epsilon, \tag{4.3.66}$$

$$P(X > t) = \min\left(\epsilon, 2e^{-\frac{t^2}{2}}\right). \tag{4.3.67}$$

Then we have the following tail 0th and 1st moment estimation,

$$\mathbb{E}\left[\|Z - \mathbb{E}[Z]\| I_{\|Z-\mathbb{E}Z\| \geq C_s}\right] \leq \int_0^\infty P(X \geq t)dt \leq \int_0^\infty \min(\epsilon, e^{-t^2/2})dx \tag{4.3.68}$$

$$\leq \int_0^{C_s} \epsilon dt + \int_{C_s}^\infty t 2e^{-\frac{t^2}{2}} dt = \epsilon(2 + C_s). \tag{4.3.69}$$

Here we use the fact that $C_s \geq 1$ for the second integration term. By the norm-subGaussian(G) tail bound, we also have

$$P(\|Z - \mathbb{E}Z\| \geq C_s) \leq 2e^{-C_s^2/2} \leq \epsilon. \tag{4.3.70}$$

□

Corollary 4.3.6. Suppose X is norm-subGaussian(G), WLOG, we assume that $\mathbb{E}X = 0$, and denote the truncated random variable $Y = XI_{\|X\|<C_sG}$, with C_s specified in Proposition 4.3.2. Then we have

$$\mathbb{E}[|Y\|^k] \leq \mathbb{E}[\|X\|^k], \tag{4.3.71}$$

$$\|\mathbb{E}[Y]\| \leq c_3\epsilon\sqrt{\log(1/\epsilon)}G. \tag{4.3.72}$$

Also $Y - \mathbb{E}Y$ is an also norm-subGaussian(C_tG), with some fixed C_t independent of ϵ, G.

This is by equivalent definitions of norm-subGaussian random variable (up to a constant) [97, Lemma 2] and a proof exactly the same (substitute $|X|$ by $\|X\|$ in their proof) as the centering argument in [182, Remark 5.18] i.e. if Y is norm-subGaussian(G) then then centering $Y - \mathbb{E}Y$ is also a norm-subGaussian(C_tG) with a absolute constant C_t. Next, we applied our truncation technique to our Oracle query gradient descent step.

Lemma 4.3.9 (Conditional subGaussian)**.** Given a random variable θ, denote the corresponding $\sigma-$field as $\mathcal{F}_\theta$. Let Z be a sample from Oracle, and denote I_{good} the indicator of the sample is good, i.e., the coin $c = 1$, let $g = \nabla f(\theta, Z)$, $d = \nabla F(\theta) - g$, then applied the above truncation results to random variable $\nabla f(\theta, Z)$, we have the following. Also when the

Oracle returns a good sample, the sample is independent to θ, with $h := d \cdot I_{\|d\|<C_sG} \cdot I_{good}$,

$$h - \mathbb{E}\left[h \mid \mathcal{F}_\theta\right] \text{ is norm-subGaussian}(C_tG), \tag{4.3.73}$$

$$\mathbb{E}\left[h \mid \mathcal{F}_\theta\right] := b(\theta), \tag{4.3.74}$$

$$\|b(\theta)\| \leq c_3\epsilon\sqrt{\log(1/\epsilon)}G. \tag{4.3.75}$$

Proof. To see this we take $X = d$, and $Y = dI_{\|d\|\leq C_sG}$. For notational simplicity, let $Z' \sim P$ independent of other random variable, and d', h', X', Y' defined correspondingly, we have

$$\|b(\theta)\| = \left\|(1-\epsilon)\mathbb{E}_{Z'}\left[d' I_{\|d'\|<C_sG} \mid \mathcal{F}_\theta\right] + \epsilon \cdot 0\right\| \leq \epsilon G, \tag{4.3.76}$$

$$\text{(condition on } I_{good} = 1 \text{ or } 0) \tag{4.3.77}$$

$$\mathbb{E}\left[\|h\|^k \mid \mathcal{F}_\theta\right] = (1-\epsilon)\mathbb{E}\left[\|Y'\|^k \mid \mathcal{F}_\theta\right] \leq \mathbb{E}\left[\|X'\|^k \mid \mathcal{F}_\theta\right]. \tag{4.3.78}$$

Note that $\|X'\|$ given θ is subGaussian(G). Therefore, by the same moment and centering argument as in Corollary 4.3.6, we have that p is norm-subGaussin(C_tG). □

Remark 4.3.8. Given θ, a query from Oracle can be understood as follows. With $1-\epsilon$ chance, Oracle decides to be good, otherwise bad, given the Oracle decides to sample from P, then with probability $1 - \beta(\theta) \geq 1 - \epsilon$, the Oracle sampling from P_θ, otherwise sample from P'_θ.

We denote the truncation induced decomposition of P given θ as $P = (1-\beta(\theta))P_\theta + \beta P'_\theta$, with $\beta = P(\|\nabla f(\theta, Z) - \nabla F(\theta)\| > C_s \cdot G) \leq \epsilon \leq 1/8$. We note that

$$\mathbb{E}_{X\sim P_\theta}[\|\nabla f(\theta,X)-\nabla F(\theta)\|^k]=\frac{1}{1-\beta}\mathbb{E}_Z[\|Z\|^k I_{\|Z\|<C_sG}]\le 2\mathbb{E}_Z[\|Z\|^k I_{\|Z\|<G_gG}], \tag{4.3.79}$$

$$\|\mathbb{E}_{X\sim P_\theta}[\nabla f(\theta,X)-\nabla F(\theta)]\|=\|\frac{1}{1-\beta}\mathbb{E}_Z[ZI_{\|Z\|>C_sG}]\|\le 2c_3\epsilon\sqrt{\log(1/\epsilon)}G. \tag{4.3.80}$$

Here Z denotes the random variable $\nabla f(\theta,Y)-\nabla F(\theta)$, where $Y\sim P$. By the same argument, with a slight abuse of C_t (times 2), we know that P_θ is also subGaussian(C_tG). With the truncation technique, we can improve Lemma 4.3.7 to get the following deviation bound. Here we say a sample Z is nice to θ if the sample is good and $\|\nabla f(\theta,Z)-\nabla F(\theta)\|\le C_sG$.

Lemma 4.3.10 (Deviation of Truncate Mean)**.** Under Assumption 1, for any fixed $\theta_1,\theta_2,\cdots,\theta_k$ selected, sample $Z_1.Z_2,\ldots,Z_m\sim$ Contaminated Oracle, if the sample size $m\ge 48\max(\log k,\ \log(1/\delta)):=s_3(k,\delta)$, denote

$$RM_i=\text{Robust-Mean}\left(\{\nabla f(\theta_i,Z_1),\cdots,\nabla f(\theta_i,Z_m)\},C_sG\right). \tag{4.3.81}$$

with probability at least $1-2\delta$, we have

$$\max_{i\in[k]}\|RM_i-\nabla F(\theta_i)\|\le 2c_3\epsilon\sqrt{\log(1/\epsilon)}G+c_2C_tG\sqrt{\frac{(\log k+\log(1/\delta)+\log 2d)}{m}} \tag{4.3.82}$$

$$+\frac{7C_sG|\mathcal{S}_{bad}|}{m/2} \tag{4.3.83}$$

$$\le 2c_3\epsilon\sqrt{\log(1/\epsilon)}G+c_2C_tG\sqrt{\frac{(\log k+\log(1/\delta)+\log 2d)}{m}} \tag{4.3.84}$$

$$+7C_sG \tag{4.3.85}$$

$$:=D_3(m,k,\delta). \tag{4.3.86}$$

Proof. The proof essentially follows Lemma 4.3.7, but this time we consider nice other than good. We define $nice_i$ as the sample set that is nice to θ_i. Note that this time, the set $nice_i$ varies for different i. Note that each sample from Oracle has greater than $1 - 2\epsilon \geq 3/4$ to be nice. Therefore, by a Binomial tail bound, we have

$$P(|nice_i| \leq m/2) \geq P(Bin(m, 1/4)/m \leq 1/2) \geq 2\exp(-2m(1/2)^2) \leq \exp(\log(\delta/k)) \leq \delta/k. \tag{4.3.87}$$

Then with probability greater than $1 - \delta/k$, we have

$$|nice_i| \geq m/2. \tag{4.3.88}$$

Then by a union bound, we know, with probability greater than $1 - \delta$, the nice sample size $|nice_i| > m/2$ for all i. Then following the exact proof, with an additional bias $2c_3\epsilon\sqrt{\log(1/\epsilon)}$ term as in P_{θ_i}. With the statistical accuracy, we can present our deviation and confidence radius select.

Corollary 4.3.7. Under Assumption 1 and 2, for $\mu(\theta'_{k(t)})$ in Algorithm 6 with stage period $L \geq 4max(\log(T/\delta), \log 2d)$, and estimate size $E \geq 48$, we have, with probability greater than $1 - 3\delta$,

$$\|\mu(\theta'_k) - \nabla F(\theta'_k)\| \leq D_3(E \cdot L, \text{Query}_2(L), \delta/T) := V_3. \tag{4.3.89}$$

Also, if we select $V = V_3 + C_s G$ in, then the algorithm will accept and proceed on all nice sample given by Oracle 2. Also, for the accepted sample, we have

$$\|\nabla f(\theta'_k, Z_k) - \nabla F(\theta'_k)\| \leq 2V_3 + C_s G := \tilde{V}_3. \tag{4.3.90}$$

Also, further with probability greater than $1-4\delta$, all of the nice sample satisfies

$$\left\|\nabla f(\theta_t, \tilde{Z}_t) - \nabla F(\theta_t)\right\| \leq c_1 C_t G\sqrt{\log(T/\delta)} + 2c_3\epsilon\sqrt{\log(1/\epsilon)}G := \bar{V}_3. \tag{4.3.91}$$

The very last argument follows from a union bound control the deviation of norm-subGaussian(C_tG) random variable, like the argument of Corollary 4.3.1, along with the fact that the truncation induces a $\epsilon\sqrt{\log(1/\epsilon)}G$ bias Lemma 4.3.9.

□

Optimization Convergence

Theorem 4.3.2. For Problem 4.1.3 with M-smooth function $F(\cdot)$, under Assumption 1 and 2, with stage period $L \geq 4max(\log(T/\delta), \log 2d)$, and estimate size $E = 48$, step size $\eta_t = \eta \leq 1/M$, confidence radius $\tilde{V} = V_3 + C_sG$ specified in Corollary 4.3.7, running Algorithm 6, then with probability greater than $1-5\delta$, the output $\bar{\theta}$ satisfies

$$F(\hat{\theta}) - F^* \lesssim \frac{1}{K\eta}\|\theta_1 - \theta^*\|^2 + C_tGR\frac{\sqrt{T\log(1/\delta)}}{K} + \epsilon\sqrt{\log(1/\epsilon)}GR + \frac{|\mathcal{S}_{rough}|}{K}\tilde{V}_3 R + \eta\bar{V}_3^2 \tag{4.3.92}$$

$$+ \eta\frac{|\mathcal{S}_{rough}|}{K}\tilde{V}_3^2. \tag{4.3.93}$$

Here we let $[K] = \mathcal{S}_{nice}\bigcup\mathcal{S}_{rough}$ corresponding to the accepted sample Z_k is nice to θ_k' or not.

Proof. Here the filtration $\mathcal{F}_t$ is σ-field of θ_1 together with all of the samples seen without the sample in the S_{next} (we hold it out, and Oracle 2 cannot peek at it). Using the same notation $e_k' = \nabla f(\theta_k, Z_k) - \nabla F(\theta_k)$, $d_t = \nabla f(\theta_{k(t)}, \tilde{Z}_t) - \nabla F(\theta_{k(t)})$. We call round t is good if the Oracle 2 returns a good sample, we call t is nice if Oracle 2 returns a good sample and $\|d_t\| \leq C_s$. We denote $h_t = d_t \cdot I_{\|d_t\|\leq C_sG} \cdot I_{t \text{ is } good}$. Following the same step in the proof

of Theorem 4.3.1, we have the telescope sum, with probability greater than $1-4\delta$

$$\sum_{k=1}^{K}(\eta - M\eta^2)\left[F(\theta_k) - F^*\right] \leq \frac{1-M\eta}{2}\underbrace{\sum_{k=1}^{K}\left(\|\theta_k - \theta^*\|^2 - \|\theta_{k+1}-\theta^*\|^2\right)}_{(I)} \tag{4.3.94}$$

$$+ (\eta - M\eta^2)\underbrace{\sum_{k\in\mathcal{S}_{nice}}\langle e_k', \theta_k - \theta^*\rangle}_{(II)} \tag{4.3.95}$$

$$+ (\eta - M\eta^2)\underbrace{\sum_{k\in\mathcal{S}_{rough}}\tilde{V}_3 R}_{(III)} \tag{4.3.96}$$

$$+ \frac{\eta^2}{2}\underbrace{\sum_{k\in\mathcal{S}_{nice}}\|e_k'\|^2}_{(IV)} + \frac{\eta^2}{2}\underbrace{\sum_{k\in\mathcal{S}_{rough}}\|e_k'\|^2}_{(V)}. \tag{4.3.97}$$

Obviously, we have

$$(I) \leq R^2, \quad (III) \leq |\mathcal{S}_{rough}|\tilde{V}_3 R, \quad (IV) \leq K\bar{V}_3^2, \quad (V) \leq |\mathcal{S}_{rough}|\tilde{V}_3^2 \tag{4.3.98}$$

To bound (II), we study the following martingale sum.

$$(II') = \sum_{i=1}^{T}\langle h_t - \mathbb{E}\left[h_t \mid \mathcal{F}_t\right], \theta_t - \theta^*\rangle. \tag{4.3.99}$$

By Lemma 4.3.9, we have the following martingale concentration. With probability greater than $1-\delta$, we have

$$(II') \lesssim C_t G R\sqrt{T\log(1/\delta)}. \tag{4.3.100}$$

Note that with probability greater than $1-3\delta$, we accept all nice samples. On this event,

we have,

$$(II) = \sum_{k \in \mathcal{S}_{nice}} \langle e'_k, \theta'_k - \theta^* \rangle = \sum_{t=1}^{T} \langle h_t, \theta_{k(t)} - \theta^* \rangle = (II') + \sum_{i=1}^{T} \langle \mathbb{E}\left[h_t \mid \mathcal{F}_t\right], \theta_{k(t)} - \theta^* \rangle. \tag{4.3.101}$$

Whereas, for the second term, we have the bias bound Lemma 4.3.9.

$$\sum_{i=1}^{T} \langle \mathbb{E}\left[h_t \mid \mathcal{F}_t\right], \theta_{k(t)} - \theta^* \rangle \leq c_3 T \epsilon \sqrt{\log(1/\epsilon)} GR. \tag{4.3.102}$$

□

Remark 4.3.9. With settings specifies in Theorem 4.3.2, we have

$$D_3(48L, 2^L, \delta/T) \lesssim \sqrt{\log(1/\epsilon)} G. \tag{4.3.103}$$

Note that C_t is a absolute constant, i.e. $O(1)$, then we know,

$$\tilde{V}_3 \lesssim \left(1 + \sqrt{\log(1/\epsilon)}\right) G, \tag{4.3.104}$$

$$\bar{V}_3 \lesssim \left(1 + \epsilon\sqrt{\log(1/\epsilon)}\right) G. \tag{4.3.105}$$

Plug above bound into (4.3.92), for T sufficient large, we know $K \geq |\mathcal{S}_{nice}| \geq (1 - 3\epsilon)T$, we have

$$F(\bar{\theta}) - F^* \lesssim \frac{1}{T\eta} R^2 + GR\sqrt{\frac{\log(T/\delta)}{T}} + \epsilon\sqrt{\log(1/\epsilon)} GR + \epsilon \tilde{V}_3 R + \eta \bar{V}_3^2 + \epsilon\eta \tilde{V}_3^2 \tag{4.3.106}$$

$$\lesssim \frac{1}{T\eta} R^2 + GR\left(\sqrt{\frac{\log(T/\delta)}{T}} + \epsilon\sqrt{\log(1/\epsilon)} + \epsilon\right) + \eta G^2 \left(1 + \epsilon \log(1/\epsilon)\right) \tag{4.3.107}$$

Optimize over $\eta \leq min(1/M, R/G\sqrt{T})$, for T sufficient large, we have

$$F(\bar{\theta}) - F^* \lesssim \frac{MR^2}{T\eta} + GR\left(\sqrt{\frac{\log(T/\delta)}{T}} + \epsilon\sqrt{\log(1/\epsilon)}\right). \tag{4.3.108}$$

$$\tag{4.3.109}$$

4.3.4 Online Mini-batch Algorithm

Algorithm

Algorithm 4.7: RSGD with mini batch query at each iteration

Mini-Batch-RSGD;
Input : Initial point θ_1, maximum number of iteration T, mini-batch size m, confidence region radius V, step-size η_t
Output: $\bar{\theta}$
for $t \leftarrow 1$ **to** T **do**
  Sample $\{Z_{t,i}\}_{i=1}^m$;
  $g_t = \text{Robust-Mean}(\{\nabla f(\theta_t, Z_{t,i})\}_{i=1}^m, V)$
  $\theta_{t+1} \leftarrow \prod_{\mathcal{D}}[\theta_t - \eta_t g_t]$;
$\bar{\theta} = \text{Mean}(\{\theta_1, \cdots, \theta_T\})$

In this section, we study a mini-batch SGD algorithm. Each time, we sample a mini-batch, and based on the fresh sample, we throw away the outliers to guarantee the bad sample is within an appropriate region around the actual gradient. Because each time we have a new sample, we don't have to make a decision based on the historical data, which free us from analyzing the query complexity/capacity of the current estimation $\mu(\theta_t)$.

Statistical Accuracy

We denote S_t as in the Robust-Mean set at each step as $S = \{g_{t,1}, \cdots, g_{t,m}\}$ be the gradient after Robust-Screening Algorithm 1, and $nice_t$ the index that are sample from P and nice to θ_t, other indexes are denote as $rough_t$.

Lemma 4.3.11 (*nice* sample size)**.** Under Assumption 1, with probability at least $1-\delta$, the following holds for the nice set size $|nice_t|$ in Algorithm 7 with mini-batch size $m > 16 \cdot \log(T/\delta)$:

$$\frac{|nice_t|}{m} > 1/2.$$

Still, this is a bound of Binomial tail with chance $1-\delta/T$ at each step t together with a union bound of total T step. With the above results, we can now select V such that our algorithm with descent chance will always accept all nice samples, and the rough sample wouldn't mislead us too much. With almost the same argument as Corollary 4.3.7, we have the following two guarantees.

Corollary 4.3.8. With input $V = C_s G$ in Algorithm 7, with probability at least $1-\delta$, the algorithm will not reject any nice sample. With probability greater than $1-\delta$, all of the nice sample satisfies

$$\left\|\nabla f(\theta_t, Z_{t,i}) - \nabla F(\theta_t)\right\| \lesssim \epsilon\sqrt{\log(1/\epsilon)}G + C_t G\sqrt{\log(mT/\delta)} := \bar{V}_4. \tag{4.3.110}$$

Corollary 4.3.9 (Deviation bound)**.** Under Assumption 1, selecting $V = C_s G$ in Algorithm 7, with probability at least $1-\delta$, the following holds for any $Z_{t,i}$ after Robust-Screen (as part of Robust-Mean),

$$\|\nabla f(\theta_t, Z_{t,i}) - \nabla F(\theta_t)\| \leq 7C_s G := \tilde{V}_4. \tag{4.3.111}$$

This the from the fact that $nice_t$ is always greater than one half with the robust screen guarantee Lemma 4.3.4.

Optimization Convergence

We define the following terms to simplify notation. Let $e_{t,i} = g_{t,i} - \nabla F(\theta_t)$, $e_t = \nabla F(\theta_t) - g_t$, and g_t is the mean of $g_{t,i}$.

Theorem 4.3.3 (Mini Batch Convergence)**.** For Problem 4.1.3, if the objective function F is M-smooth, under Assumption 1, with sub-sample size $m \geq 16 \log(T/\delta)$, step size $\eta_t = \eta \leq 1/M$, confidence radius $\tilde{V} = C_s G$ in Algorithm 7, then with probability greater than $1 - 3\delta$, output $\bar{\theta}$ satisfies

$$F(\bar{\theta}) - F^* \lesssim \frac{1}{K\eta}\|\theta_1 - \theta^*\|^2 + C_t G R\sqrt{\frac{\log(1/\delta)}{mT}} + \epsilon\sqrt{\log(1/\epsilon)} GR \tag{4.3.112}$$

$$+ \frac{|\mathcal{S}_{\text{all rough}}|}{mT} C_s G R + \eta\left[\frac{(C_t G)^2 \log(T/\delta)}{m} + (\epsilon\sqrt{\log(1/\epsilon)} G)^2\right] \tag{4.3.113}$$

$$+ \eta \frac{|\mathcal{S}_{\text{all rough}}|}{mT} \tilde{V}_4^2. \tag{4.3.114}$$

Here $\mathcal{S}_{\text{all rough}} = \cup_{i=1}^{T} \mathcal{S}_{\text{rough}_i}$.

Proof. Following the same proof for Theorem 4.3.2, we have

$$\sum_{t=1}^{T} (\eta - M\eta^2) \left[F(\theta_{t+1}) - F^*\right] \leq \frac{1 - M\eta}{2}\left(\|\theta_0 - \theta^*\|^2 - \|\theta_{t+1} - \theta^*\|^2\right) + \tag{4.3.115}$$

$$(\eta - M\eta^2) \underbrace{\sum_{t=1}^{T} \langle e_t, \theta_t - \theta^* \rangle}_{I} + \frac{\eta^2}{2} \underbrace{\sum_{t=1}^{T} \|e_t\|^2}_{II} \tag{4.3.116}$$

Let's now consider the two error terms,

$$I = \sum_{t=1}^{T} \langle e_t, \theta_t - \theta^* \rangle, \text{ and} \tag{4.3.117}$$

$$II = \sum_{t=1}^{T} \|e_t\|^2. \tag{4.3.118}$$

First, with probability greater than $1-\delta$, we have the following decomposition I,

$$I = \sum_{t=1}^{T} \sum_{i \in nice_t} \frac{1}{m} \langle e_{t,i}, \theta_t - \theta^* \rangle + \sum_{t=1}^{T} \sum_{i \in rough_t} \frac{1}{m} \langle e_{t,i}, \theta_t - \theta^* \rangle \tag{4.3.119}$$

$$\lesssim C_t GR \sqrt{\frac{T \log(1/\delta)}{m}} + \epsilon \sqrt{\log(1/\epsilon)} TGR + \frac{|\mathcal{S}_{\text{all rough}}|}{m} C_S GR. \tag{4.3.120}$$

The first term is by Azuma-Hoeffding inequality [152], which holds with probability greater than $1-\delta$, and Lemma 4.3.9. The second term is by Lemma 4.3.11 and the error bound (4.3.111). For the second error II, we have

$$II \lesssim \frac{1}{m^2} \left(\sum_{t=1}^{T} \left\| \sum_{i \in nice_t} e_{t,i} \right\|^2 + \sum_{t=1}^{T} \left\| \sum_{i \in rough_t} e_{t,i} \right\|^2 \right) \tag{4.3.121}$$

$$\lesssim T \left[\frac{(C_t G)^2 \log(T/\delta)}{m} + (\epsilon \sqrt{\log(1/\epsilon)} G)^2 \right] + \sum_{t=1}^{T} \frac{|\mathcal{S}_{rough_t}|^2}{m^2} \tilde{V}_4^2 \tag{4.3.122}$$

$$\lesssim T \left[\frac{(C_t G)^2 \log(T/\delta)}{m} + (\epsilon \sqrt{\log(1/\epsilon)} G)^2 \right] + \frac{|\mathcal{S}_{\text{all rough}}|}{m} \tilde{V}_4^2, \tag{4.3.123}$$

where the first term is by concentration property of truncated subGaussian Corollary 4.3.6 and the last inequality using the fact that $|\mathcal{S}_{rought}| \leq m/2$. □

Remark 4.3.10. Now, for T large enough, we know $|\mathcal{S}_{\text{all rough}}|/mT \leq 2\epsilon$ (since we always accept *nice* sample). Picking $\eta_t = \eta \leq 1/M$, we have

$$F(\bar{\theta}) - F^* \lesssim \frac{1}{T\eta} R^2 + GR \sqrt{\frac{\log(T/\delta)}{mT}} + \epsilon \sqrt{\log(1/\epsilon)} GR + \eta G^2 \frac{\log(T/\delta)}{m} \tag{4.3.124}$$

$$+ \eta \epsilon^2 \log(1/\epsilon) G^2 + \eta \epsilon \log(1/\epsilon) G^2 \tag{4.3.125}$$

$$\lesssim \frac{1}{T\eta} R^2 + GR \left(\sqrt{\frac{\log(T/\delta)}{mT}} + \epsilon \sqrt{\log(1/\epsilon)} \right) + \eta G^2 \left(\frac{\log(T/\delta)}{m} + \epsilon \log(1/\epsilon) \right) \tag{4.3.126}$$

Optimize over $\eta \leq min(1/M, R/G\sqrt{T})$, for T sufficient large, we have

$$F(\bar{\theta}) - F^* \lesssim \frac{MR^2}{T\eta} + GR\left(\sqrt{\frac{\log(T/\delta)}{mT}} + \epsilon\sqrt{\log(1/\epsilon)}\right). \tag{4.3.127}$$

4.4 Lower Bound

Here, we note that the i) MR^2/T and ii) $GR/\sqrt{T}$ terms corresponding to the i) (noiseless) pure optimization rate and ii) stochasticity rate have already been studied in-depth in the literature. Our main goal is to give a lower bound of the terms involving ϵ, i.e., the intrinsic bias induced by the contamination. We reduce the optimization problem to a hypothesis testing problem and utilize the corresponding information-theoretic lower bound for the lower bound of the problem. First, we present a well-known lower bound result of hypothesis testing with contaminated distributions.

Lemma 4.4.1. For any distributions P_1, P_2, if the their total variation distance $\mathrm{TV}(P_1, P_2) \leq \frac{\epsilon}{1-\epsilon}$, then their exist ϵ-contaminated distributions $\mathcal{Q}_1$, $\mathcal{Q}_2$, such that the contaminated distributions are same:

$$P_1' = (1-\epsilon)P_1 + \epsilon\mathcal{Q}_1 = (1-\epsilon)P_2 + \epsilon\mathcal{Q}_2 = P_2'.$$

Proof. Let $C = (1-\epsilon)TV(P_1, P_2) \leq \epsilon$. Take $\mathcal{Q}_1 = \frac{1}{\epsilon}(P_2 - P_1)_+ + \frac{\epsilon - C}{\epsilon}\delta_0$ and $\mathcal{Q}_2 = \frac{1}{\epsilon}(P_1 - P_2)_+ + \frac{\epsilon - C}{\epsilon}\delta_0$, with C a normalizing factor, which makes the contaminated distributions exactly the same. □

Therefore, even given infinite number of contaminated samples, we cannot tell whether P_1 or P_2 is the true model. By the information-theoretic lower bound of hypothesis testing, we have the following impossibility result.

Corollary 4.4.1. For distributions P_1, P_2, with $\mathrm{TV}(P_1, P_2) \leq \frac{\epsilon}{1-\epsilon}$, there is no testing

function ϕ that can distinguish them (better than random guess) from arbitrary amount of samples from the ϵ-contaminated distribution.

Now we reduce the optimization with smooth function to a hypothesis test function. Taking $f(\theta, z) = G\langle\theta, z\rangle$ with domain $\theta \times z \in [0, R] \times \mathbb{R}$. Consider two distributions $P_1 = N(C\epsilon, 1)$, and $P_2 = N(-C\epsilon, 1)$, with $C = \sqrt{2}$. Then we have the following result: there is no algorithm with output $\hat{\theta}$ such that with probability greater than $2/3$, we have

$$|F(\hat{\theta}) - F^*| \leq \frac{CGR\epsilon}{2}.$$

Proof. First we know

$$\mathrm{TV}(P_1, P_2) \leq \sqrt{\frac{KL(P_1, P_2)}{2}} \leq \sqrt{2}C\epsilon \leq \frac{\epsilon}{1-\epsilon}. \tag{4.4.1}$$

We construct a function ϕ taking 1 if $F(\theta_T) \geq (CGR\epsilon)/2$ and taking 2 otherwise. If we have an algorithm that with probability at least $2/3$, outputs an $(CGR\epsilon)/2$-optimal solution, then the testing function ϕ with can distinguish P_1 and P_2 with probability greater than $1/2$, which violates the impossibility result Corollary 4.4.1. □

CHAPTER 5

TRAINING NEURAL NETWORKS AS LEARNING DATA-ADAPTIVE KERNELS: PROVABLE REPRESENTATION AND APPROXIMATION BENEFITS

5.1 Introduction

Consider i.i.d. data pairs drawn from a joint distribution $(\mathbf{x}, \mathbf{y}) \sim P = P_x \times P_{y|x}$ on the space $\mathcal{X} \times \mathcal{Y}$. At the intersection of statistical learning theory [179] and approximation theory [39], the following *approximation* problem requires to be first understood before any further statistical results to be established. For a model class $\mathcal{F}$, one is interested in whether there exists $f \in \mathcal{F} : \mathcal{X} \to \mathcal{Y}$ such that the population squared loss is small,

$$L(f) = \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim P}\frac{1}{2}\left(\mathbf{y} - f(\mathbf{x})\right)^2 = \mathbf{E}_{\mathbf{x}\sim P_x}\frac{1}{2}\left(f_*(\mathbf{x}) - f(\mathbf{x})\right)^2 + \mathbf{E}_{(\mathbf{x},\mathbf{y})\sim P}\frac{1}{2}\left(\mathbf{y} - f_*(\mathbf{x})\right)^2, \tag{5.1.1}$$

with the conditional expectation (or Bayes estimator) defined as $f_*(x) := \mathbf{E}[\mathbf{y}|\mathbf{x} = x]$. Eqn. (5.1.1) generally reads as approximating f_* in the mean squared error sense.

Statistically, researchers approach the above question mainly in two ways. The first is by assuming that the conditional expectation f_* lies in the correct model class $\mathcal{F}$. For example, say $\mathcal{F}$ consists of linear models or splines with a particular order of smoothness or more broadly, functions lying in a reproducing kernel Hilbert space (RKHS). Conceptually, this "well-specification" assumption requires substantial knowledge about what model class $\mathcal{F}$ might be suitable for the regression task at hand, which is often unavailable in practice. Within each framework, minimax optimal rates and extensive study have been established in [165, 185]. The second way, which extends the first approach further, considers all f_* under some mild conditions. Building upon certain *universal approximation theorem*, one studies a

sequence of model classes $\mathcal{F}_\epsilon$ called sieves with ϵ changing [70], such that the class $\mathcal{F}_\epsilon$ contains an ϵ-approximation to any f_* under some metric. A final result usually requires a careful balancing of the approximation and stochastic error by tuning ϵ. Particular cases for the latter approach include polynomials (Stone-Weierstrass, Bernstein), radial-basis [131, 128], and two-layer and multi-layer neural networks [39, 87, 4, 137, 40, 6, 60, 103, 134].

However, the following significant drawbacks of the above current theory make it inadequate to present an *adaptive* and realistic explanation of the practical success of *neural networks*. Firstly, the function computed in practice could be very different from that claimed in the approximation theory, either by the existence or by constructions. To see this, consider the multi-layer neural networks. It is hard to conceive that the function, computed in practice via the now-standard stochastic gradient descent (SGD) training procedure, is close to the one asserted by the universal approximation results. Secondly, in practice, researchers usually explore different model classes $\mathcal{F}$ to learn which representation best suits the data. For example, using different kernels machines, random forests, or specify certain architectures then run SGD on neural networks. In this case, strictly speaking, the choice of the model class depends on the data in an *adaptive* way, without prior knowledge about the basis. There have been substantial advances made to address the above two concerns — for instance, [98] on the first and [88, 7] on the second — for $\mathcal{F}$ being a linear span of a library of candidate functions (union of various set of basis that can be correlated), with greedy selection rules. Nevertheless, the current theory still falls short of describing the approximation and adaptivity for the non-convex and possibly non-smooth gradient descent training on all-layer weights of the neural networks, as done in practice.

We take a step to bridge the above mismatch in the current theory and practice for neural networks and to establish a theoretical framework where the model classes adapt to the data. In particular, we answer the following *algorithmic approximation* question:

Given data pair $(\mathbf{x}, \mathbf{y}) \sim P$, denote $f_*(x) = \mathbf{E}[\mathbf{y}|\mathbf{x} = x]$. Specify a neural

networks model, and run gradient flow until any stationarity ($t \to \infty$). Denote the computed function to be $f_t(x)$. How does $f_t(x)$ relate to $f_*(x)$, in terms of approximation and representation?

Also, we aim to formalize and shed light on the *representation benefits* of neural networks:

What are the provable benefits of the adaptive representation learned by training neural networks compared to the classical nonparametric pre-specified fixed basis representation?

The intimate connection between two-layer neural networks and reproducing kernel Hilbert spaces (RKHS) has been studied in the literature, see [137, 38, 40, 6, 93]. However, to the best of our knowledge, known results are mostly based on a *fixed* RKHS (in our notation K_0 in Section 5.4.1). In that sense, random features for kernel learning [137, 138, 148] can be viewed as neural networks with fixed random sampled first layer weights and tunable second layer weights. From the neural networks side, [144, 117, 162] study the mean-field theory for two-layer neural networks, and [93, 52, 36, 74] study the linearization of neural networks around the initialization and draw connections to RKHS $\mathcal{K}_0$ in various over-parametrized settings. In contrast, we will establish a general theory with the *dynamic* and *data-adaptive* RKHS $\mathcal{K}_t$ obtained via training neural networks, with standard gradient flow on weights of *both* layers. Connections and distinctions to the literature that motivates our study are further discussed with details in Section 5.4. As a distinctive feature of the adaptive theory, we emphasize that all $f_* \in L^2(P_x)$ is considered without pre-specified structural assumptions.

5.1.1 Problem Formulation

In this paper, we consider the time-varying function f_t to approximate f_*, parametrized by a two-layer rectified linear unit (ReLU) neural network (NN).

$$f_t(x) := \sum_{j=1}^{m} w_j(t)\sigma(x^T u_j(t)). \tag{5.1.2}$$

The time index t corresponds to the evolution of parameters driven by the gradient flow/descent (GD) training dynamics. Here each individual pair $(w_j \in \mathbb{R}, u_j \in \mathbb{R}^d)$ in the summation is associated with a *neuron*. Consider the gradient flow as the training dynamics for the weights of the neurons: for the loss function $\ell(y, f) = (y - f)^2/2$ and the random variable $\mathbf{z} := (\mathbf{x}, \mathbf{y})$, the parameters (w_j, u_j) evolve with time as follows

$$\frac{dw_j(t)}{dt} = -\mathbf{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, f_t)}{\partial f}\sigma(\mathbf{x}^T u_j(t))\right], \quad \frac{du_j(t)}{dt} = -\mathbf{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, f_t)}{\partial f} w_j(t)\mathbf{1}_{\mathbf{x}^T u_j(t)\geq 0}\mathbf{x}\right]. \tag{5.1.3}$$

Equivalently, we can rewrite the function computed by NN at time t as

$$f_t(x) := \int \sigma(x^T u)\tau_t(du), \tag{5.1.4}$$

where $\tau_t = \sum_{j=1}^{m} w_j(t)\delta_{u_j(t)}$ is a signed combination of delta measures. We will define a careful rescaling of τ_t denoted as ρ_t (Eqn. (5.4.8)), then derive the corresponding distribution dynamic for ρ_t driven by the gradient flow later in Section 5.4.2. The rescaled formulation naturally extends to the infinite neurons case with $m \to \infty$.

In this paper, by considering various distributions of $\mathbf{z}$, we study two following problems: approximation and empirical risk minimization (ERM).

Function Approximation: The data pair $\mathbf{z} \sim P$ is sampled from the population joint distribution. We are going to answer how f_t *approximates* $f_*(x) = \mathbf{E}[\mathbf{y}|\mathbf{x} = x]$ in function

spaces, induced by the gradient flow on neuron weights

$$\mathbf{E}_{\mathbf{z}\sim P}(\mathbf{y}-f_t(\mathbf{x}))^2=\|f_t-f_*\|_{L^2_\mu}^2+\mathbf{E}_{\mathbf{z}\sim P}(\mathbf{y}-f_*(\mathbf{x}))^2\ . \tag{5.1.5}$$

Here we denote $\mu:=P_x$ and remark that all $f_*\in L^2_\mu$ are considered without additional assumptions.

ERM and Interpolation: The data pair $\mathbf{z}\sim\frac{1}{n}\sum_{i=1}^n\delta_{\mathbf{x}=x_i,\mathbf{y}=y_i}$ follows the empirical distribution. We will study gradient flow for the ERM

$$\frac{1}{2n}\sum_{i=1}^n(y_i-f_t(x_i))^2\ . \tag{5.1.6}$$

In this case, the target reduces to $\widehat{\mathbf{E}}[\mathbf{y}|\mathbf{x}=x_i]=y_i$ with $\widehat{\mathbf{E}}$ as the empirical expectation. When the minimizer of Eqn. (5.1.6) achieves the zero loss, we call it the *interpolation* problem [199, 13, 112, 110, 139, 12]. Here we are interested in when and how $f_t(x_i)$ *interpolates* y_i, for $1\le i\le n$.

Finally, we remark that in practice, extending the gradient flow results to the (1) positive step size GD and (2) mini-batch stochastic GD are standalone interesting research topics. The reasons are that the optimization is non-smooth for the ReLU activation and that the interplay between the batch size and step size is less transparent in non-convex problems.

5.1.2 Notations

We use the boldface lower case $\mathbf{x}$ to denote a random variable or vector. The normal letter x can either be a scaler or a vector when there is no confusion. The transpose of a matrix $\mathbf{A}$, resp. vector u is denoted by $\mathbf{A}^T$, resp. u^T. $\mathbf{A}^+$ denotes the Moore–Penrose inverse. For $n\in\mathbb{N}$, let $[n]:=\{1,\dots,n\}$. We use $\mathbf{A}[i,j]$ to denote the i,j-th entry of a matrix. We denote $\mathbf{1}_{\mathcal{D}}$ as the indicator function of set $\mathcal{D}$. We call symmetric positive semidefinite functions $K(\cdot,\cdot),H(\cdot,\cdot):\mathcal{X}\times\mathcal{X}\to\mathbb{R}$ kernels, and use calligraphy letter

$\mathcal{K}, \mathcal{H}$ to denote Hilbert spaces. We use $\langle f, g\rangle_\mu = \int f(x)g(x)\mu(dx)$ to denote the inner product in L^2_μ (or $L^2(P_x)$). $\hat{\mu}$ denotes the empirical distribution for μ. Notation $\mathbf{E}_{\mathbf{x}}$ is the expectation w.r.t random variable $\mathbf{x}$, and $\mathbf{E}_{\mathbf{x},\tilde{\mathbf{x}}} h(\mathbf{x}, \tilde{\mathbf{x}}) = \int\int h(x, \tilde{x})\mu(dx)\mu(d\tilde{x})$. For a signed measure $\rho = \rho_+ - \rho_-$ with the positive and negative parts, define $|\rho| = \rho_+ + \rho_-$.

5.1.3 Preliminaries

We use the signed measure ρ_t, defined by the neuron weights at training time t collectively, to construct a *dynamic RKHS*. The mathematical definition of ρ_t is deferred to Section 5.4.1 and 5.4.2 (specifically, Eqn. (5.4.8)). The stationary signed measure at $t \to \infty$ is denoted as ρ_∞. For completeness we walk through the construction of the dynamic kernel and RKHS with ρ_t. Define the linear operator $\mathcal{T} : L^2_\mu(x) \to L^2_{|\rho_t|}(\Theta)$, such that for any $f(x) \in L^2_\mu(x)$

$$(\mathcal{T}f)(\Theta) := \int f(x)\|\Theta\|\sigma(x^T\Theta)\mu(dx), \ \forall \Theta \in \mathrm{supp}(\rho_t).$$

One can define the adjoint operator $\mathcal{T}^\star : L^2_{|\rho_t|}(\Theta) \to L^2_\mu(x)$, such that for $p(\Theta) \in L^2_{|\rho_t|}(\Theta)$,

$$(\mathcal{T}^\star p)(x) := \int p(\Theta)\|\Theta\|\sigma(x^T\Theta)|\rho_t|(d\Theta).$$

Note that both $\mathcal{T}$ and $\mathcal{T}^\star$ are compact operators under the finite total variation and compact support assumptions. For the finite neurons case (5.1.2), the operator is of finite rank. We define the compact integral operator $\mathcal{T}^\star\mathcal{T}$ with the corresponding kernel

$$H_t(x, \tilde{x}) = \int \|\Theta\|^2\sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta)|\rho_t|(d\Theta), \text{ and } (\mathcal{T}^\star\mathcal{T}f)(x) := \int H_t(x, \tilde{x})f(\tilde{x})\mu(d\tilde{x}). \tag{5.1.7}$$

The dynamic RKHS $\mathcal{H}_t$ can be readily constructed via H_t. Let the eigen decomposition of $\mathcal{T}^\star\mathcal{T}$ be the countable sum $\mathcal{T}^\star\mathcal{T} = \sum_{i=1}^E \lambda_i e_i e_i^*$. Here E can be a nonnegative integer

or ∞, and $\lambda_i > 0$. e_i without confusion can represent either an eigen function or a linear functional. Similarly, we have the singular value decomposition for $\mathcal{T} = \sum_{i=1}^{E} \sqrt{\lambda_i} t_i e_i^*$. and $\mathcal{T}^\star$ as well. For a detailed discussion, see e.g. [32]. Again, t_i is a function in $L^2_{|\rho_t|}(\Theta)$ or a linear functional. The RKHS can be specified as follows.

$$\mathcal{H}_t = \left\{ h \mid h(x) = \sum_i h_i e_i(x),\ \sum_i \frac{h_i^2}{\lambda_i} < \infty \right\}.$$

We refer to H_∞ as the stationary RKHS kernel and $\mathcal{H}_\infty$ as the stationary RKHS. One can view that the gradient flow training dynamics — on the parameters of NN — induces a sequence of functions $\{f_t : t \geq 0\}$ and dynamic RKHS $\{\mathcal{H}_t : t \geq 0\}$, indexed by the time t.

5.1.4 Organization and Summary

	finite neurons m	infinite neurons $m \to \infty$
finite samples n	Interpolation (finite rank kernel, Thms. 5.2.1, 5.2.1 & Prop. 5.3.1)	Interpolation (finite rank kernel, Thms. 5.2.1, 5.2.1 & Prop. 5.3.1)
infinite samples $n \to \infty$	Approximation (finite rank kernel, Thms. 5.2.1 & 5.2.1)	Approximation (possibly universal kernel[2], Thms. 5.2.1 & 5.2.1)

Table 5.1: Nature of the results studied in this paper.

We will prove three results, which are summarized informally in this section (see also Table 5.1). We remark that Theorems 5.2.1 and 5.2.1 are stated for the approximation problem. However, as done in Corollary 5.2.1, by substituting $\mathcal{P}, \mu$ by the empirical counterparts, one can easily state the analog for the ERM problem. Recall $f_*(x) = \mathbf{E}[\mathbf{y}|\mathbf{x} = x]$.

Gradient flow on NN converges to projection onto data-adaptive RKHS. Theorem 5.2.1 shows that as done in practice training NN with simple gradient flow, in the limit of any *local* stationarity, learns the adaptive representation, and performs the *global* least squares projection simultaneously. Define $f_\infty = \lim_{t\to\infty} f_t$ as the function computed

2. Whether the kernel is universal in the $m, n \to \infty$ case still depends on f_* and the data distribution P. See the simulations of [113].

by ReLU networks (defined in (5.1.2), or more generally in (5.4.9)) until any stationarity of the gradient flow dynamics (defined in (5.1.3), with the squared loss) for the population distribution $(\mathbf{x}, \mathbf{y}) \sim P$. Define the corresponding stationary RKHS
$\mathcal{H}_\infty = \lim_{t\to\infty} \mathcal{H}_t$ (defined in (5.1.7)).

> [Informal version of Thm. 5.2.1] Consider $f_* \in L^2_\mu$, for any local stationarity of the gradient flow dynamics (5.1.3) on the weights of neural networks (5.1.2), the function computed by NN at stationarity f_∞ satisfies
>
> $$f_\infty \in \arg\min_{g\in\mathcal{H}_\infty} \|f_* - g\|^2_{L^2_\mu}.$$

Representation benefits of data-adaptive RKHS. Theorem 5.2.1 illustrates the provable benefits of the learned data-adaptive representation/basis $\mathcal{H}_\infty$. We emphasize that $\mathcal{H}_\infty$, as obtained by training neural networks on the data $(\mathbf{x}, \mathbf{y}) \sim P$, depends on the data in an implicit way such that there are advantages of representing and approximating f_*.

> [Informal version of Thm. 5.2.1] Consider $f_* \in L^2_\mu$ and the same setup as Theorem 5.2.1. Decompose f_* into the function f_∞ computed by the neural network and the residual Δ_∞
>
> $$f_* = f_\infty + \Delta_\infty.$$
>
> Then there is another RKHS (defined in (5.2.4)) $\mathcal{K}_\infty \supset \mathcal{H}_\infty$, such that
>
> $$f_\infty \in \mathcal{H}_\infty, \quad \Delta_\infty \in \mathrm{Ker}(\mathcal{K}_\infty) \subset \mathrm{Ker}(\mathcal{H}_\infty),$$
>
> with a gap in the spaces $\mathcal{H}_\infty \oplus \mathrm{Ker}(\mathcal{K}_\infty) \neq L^2_\mu$.

Convergence to Ridgeless regression with adaptive kernels. Proposition 5.3.1 establishes that in the vanishing regularization $\lambda \to 0$ limit, the neural network function computed

by gradient flow converges to the kernel ridgeless regression with an adaptive kernel (denoted as $\widehat{f}_{\infty}^{\text{rkhs}}(x)$). Consider using the gradient flow on the weights of the neural network function $f_t(x) = \sum_{j=1}^{m} w_j(t)\sigma(x^T u_j(t))$, to solve the ℓ_2-regularized ERM

$$\frac{1}{2n}\sum_{i=1}^{n}(y_i - f_t(x_i))^2 + \frac{\lambda}{2m}\sum_{j=1}^{m}\left[w_j(t)^2 + \|u_j(t)\|^2\right] \quad .$$

Denote the function computed by NN at any local stationarity of ERM as $\widehat{f}^{\text{nn},\lambda}(x)$, we answer the extrapolation question at a new point x, with the generalization error discussed in Prop. 5.3.2. The result is extendable to the infinite neurons case.

> [Informal version of Prop. 5.3.1] Consider only the bounded assumption on initialization that $|w_j^2(0) - \|u_j\|^2(0)| < \infty$ for all $1 \le j \le m$. At stationarity, denote the corresponding adaptive kernel as $\widehat{H}_{\infty}^{\lambda}$. The neural network function $\widehat{f}_{\infty}^{\text{nn},\lambda}(x)$ has the following expression,
>
> $$\lim_{\lambda \to 0} \widehat{f}_{\infty}^{\text{nn},\lambda}(x) = \widehat{H}_{\infty}(x, X)\widehat{H}_{\infty}(X, X)^{+}Y =: \widehat{f}_{\infty}^{\text{rkhs}}(x)$$
>
> (ridgeless regression with kernel $\widehat{H}_{\infty}$).

5.2 Main Results: Benefits of Adaptive Representation

We formally state two main results of the paper, Theorem 5.2.1 and Theorem 5.2.1 below.

5.2.1 Gradient Flow, Projection and Adaptive RKHS

We study how the function f_t computed from gradient flow on NN represents f_* when reaching any stationarity under the squared loss. Consider the gradient flow dynamics (5.4.12) reaching *any stationarity*. Assume that the corresponding signed measure in (5.4.8) satisfies $\text{TV}(\rho_{\infty}) < \infty$ with a compact support. The mathematical details about ρ_{∞} are postponed

to Section 5.4.2. We employ the notation ρ_∞ since reaching stationarity can be viewed as $t \to \infty$.

We would like to emphasize that this stationary signed measure ρ_∞ is *task adaptive*: it implicitly depends on the regression task f_* and the data distribution P, rather than being pre-specified by the researcher as in [6, 40, 38]. With the RKHS established in Section 5.1.3, we are ready to state the following theorem.

Theorem 5.2.1 (Approximation)**.** *For any conditional mean $f_*(x) = \mathbb{E}[\mathbf{y}|\mathbf{x} = x] \in L^2_\mu$, consider solving the approximation problem* (5.1.5)*, with the ReLU NN function f_t defined in* (5.1.2) *where $w_j(t)$ and $\theta_j(t)$ are the weights for $t \geq 0, 1 \leq j \leq m$. For any signed measure ρ_0 with $\mathrm{TV}(\rho_0) < \infty$, consider the infinitesimal initialization weights $u_j(0) = \Theta_j/\sqrt{m}$, and $w_j(0) = \mathrm{sgn}(\rho_0(\Theta_j))\|\Theta_j\|/\sqrt{m}$, with $\Theta_j \sim \rho_0$ sampled independently. When the training dynamics* (5.1.3) *reaches any stationarity, it defines a stationary signed measure $\rho_\infty^{(m)}$ (on the collective weights) with $\mathrm{TV}(\rho_\infty^{(m)}) < \infty$, and a corresponding stationary RKHS $\mathcal{H}_\infty$ with the kernel defined in Eqn.* (5.1.7)*, such that:*

1. *the function computed by neural networks at stationarity has the form*

$$f_\infty(x) = \int \|\Theta\|\sigma(x^T\Theta)\rho_\infty^{(m)}(d\Theta) \ ; \tag{5.2.1}$$

2. *f_∞ is a global minimizer of approximating f_* within the RKHS $\mathcal{H}_\infty$*

$$f_\infty \in \arg\min_{g \in \mathcal{H}_\infty} \|f_* - g\|^2_{L^2_\mu} \ . \tag{5.2.2}$$

In addition, the same results extend to the infinite neurons case with $m \to \infty$ where the limit for $\rho_\infty^{(m)}$ can be defined in the weak sense.

Remark 5.2.1. The above theorem shows that $\lim_{t\to\infty} f_t$ obtained by training on two-layer weights over time until any stationarity, is the same as projecting f_* onto the stationary

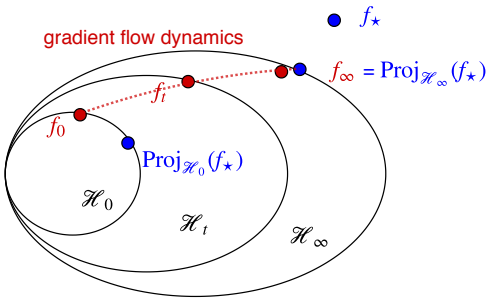


Figure 5.1: Illustration of Theorem 5.2.1. Red dotted line denotes the function f_t computed along the gradient flow dynamics on the weights of NN. Along training, one learns a sequence of dynamic RKHS representation $\mathcal{H}_t$'s. Over time, f_t converges to the projection of f_* onto $\mathcal{H}_\infty$. We emphasize that the initial function f_0 computed by NN is very different from the projection of f_* onto the initial RKHS $\mathcal{H}_0$.

RKHS $\mathcal{H}_\infty$. The projection is the solution to the classic nonparametric least squares, had one known the adaptive representation $\mathcal{H}_\infty$ beforehand. Conceptually, this is *distinct* from the theoretical framework in the current statistics and learning theory literature: we do not require the structural knowledge about f_* (say, smoothness, sparsity, reflected in $\mathcal{F}$). Instead, we run gradient descent on neural networks to learn an adaptive representation for f_*, and show how the computed function represents f_* in this adaptive RKHS $\mathcal{H}_\infty$.

In other words, as done in practice training NN with simple gradient flow, in the limit of any *local* stationarity, learns the adaptive representation, and performs the *global* least-squares projection simultaneously. Training NN is learning a dynamic representation (quantified by $\mathcal{H}_t$), at the same time updating the predicted function f_t, as shown in Fig. 5.1.

A final note on the infinite neuron case: for any fixed time t, with the proper random initialization, setting $m \to \infty$ defines a proper distribution dynamics on the weak limit ρ_t shown in Lemma 5.4.3. Then set $t \to \infty$ to obtain the stationarity RKHS $\mathcal{H}_\infty$.

From the above, we have the following natural decomposition,

$$\Delta_\infty(x) = f_*(x) - f_\infty(x) \in \mathrm{Ker}(\mathcal{H}_\infty). \tag{5.2.3}$$

Surprisingly, as we show in the next section, Δ_∞ actually lies in a smaller subspace of $\mathrm{Ker}(\mathcal{H}_\infty)$, characterized by $\mathrm{Ker}(\mathcal{K}_\infty)$. We call this the *representation and approximation benefits* of the data-adaptive RKHS learned by training neural networks.

Before moving next, we briefly discuss the above theorem when applied to the empirical measure, to solve the ERM problem. First, as a direct corollary, the following holds.

Corollary 5.2.1 (ERM)**.** Consider the ERM problem (5.1.6), with the other settings the same as in Theorem 5.2.1. One can define the finite dimensional RKHS $\widehat{\mathcal{H}}_\infty$ (at most rank n) as in (5.1.7) with $\widehat{\mu} = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$ substituting μ. When reaches any stationarity, the solution satisfies

$$\widehat{f}_\infty \in \arg\min_{g \in \widehat{\mathcal{H}}_\infty} \frac{1}{n}\sum_{i=1}^n (y_i - g(x_i))^2 \ .$$

More importantly, we will show in Proposition 5.3.1 that the function computed by training neural networks with gradient descent on the empirical risk objective $\widehat{f}_\infty(x)$ until any stationarity (with vanishing ℓ_2 regularization), can be shown to be the **kernel ridgeless regression** with the **data-adaptive RKHS** $\widehat{\mathcal{H}}_\infty$. Hence, studying the out of sample performance for GD on NN reduces to the generalization of kernel ridgeless regression with adaptive kernels.

5.2.2 Representation Benefits of Adaptive RKHS

We now define another adaptive RKHS $\mathcal{K}_\infty$ named as the GD kernel, which turns out to be different from $\mathcal{H}_\infty$ in (5.1.7). Interestingly, the difference in these two kernels sheds light on the representation benefits of the adaptive RKHS. The new RKHS $\mathcal{K}_\infty$ is motivated by the gradient training dynamics. Recall the associated signed measure ρ_∞ at the stationarity,

The GD kernel is defined as

$$K_\infty(x,\tilde{x}) = \int \left(\|\Theta\|^2 \mathbf{1}_{x^T\Theta\geq 0}\mathbf{1}_{\tilde{x}^T\Theta\geq 0} x^T\tilde{x} + \sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta) \right) |\rho_\infty|(d\Theta) \neq H_\infty(x,\tilde{x}) \tag{5.2.4}$$

which is different than the stationary RKHS kernel H_∞ in (5.1.7). We use $\mathcal{K}_t : L^2_\mu(x) \to L^2_\mu(x)$ to denote the integral operator associated with K_t,

$$(\mathcal{K}_t f)(x) := \int K_t(x,\tilde{x}) f(\tilde{x}) \mu(d\tilde{x}).$$

With a slight abuse of notation, we denote the corresponding RKHS to be $\mathcal{K}_t$ as well. Now we are ready to state the main theorem on the representation benefits.

Theorem 5.2.1 (Representation Benefits)**.** Consider $f_* \in L^2_\mu$ and the same setting as in Theorem 5.2.1. Consider the approximation problem (5.1.5) with either finite or infinite neurons, and the gradient flow dynamics (5.4.12) (equivalently (5.1.3)) with data pair $(\mathbf{x},\mathbf{y}) \sim P$ drawn from the population distribution. When reaching any stationary signed measure ρ_∞, f_* is decomposed into the function f_∞ computed by the neural network and the residual Δ_∞

$$f_* = f_\infty + \Delta_\infty.$$

Recall the RKHS $\mathcal{H}_\infty$ in (5.1.7) and the GD RKHS $\mathcal{K}_\infty$ in (5.2.4), all learned from the data $(\mathbf{x},\mathbf{y}) \sim P$ and f_* adaptively. The following holds,

$$f_\infty \in \mathcal{H}_\infty, \quad \Delta_\infty \in \mathrm{Ker}(\mathcal{K}_\infty) \subset \mathrm{Ker}(\mathcal{H}_\infty),$$

with $\mathcal{H}_\infty \oplus \mathrm{Ker}(\mathcal{K}_\infty) \neq L^2_\mu$. In other words, GD on NN decomposes f_* into two parts, and each lies in a space that is NOT the orthogonal complement to the other.

Remark 5.2.2. As we can see $\mathrm{Ker}(K_\infty)$ and $\mathrm{Ker}(H_\infty)$ are not the same. Therefore, the decomposition $f_\infty + \Delta_\infty$ is not a trivial orthogonal decomposition to the RKHS $\mathcal{H}_\infty$ and its complement.

Recall Theorem 5.2.1, projecting f_* to the RKHS $\mathcal{H}_\infty$ with the data-adaptive kernel

$$H_\infty(x, \tilde{x}) = \int \sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta)|\rho_\infty|(d\Theta)$$

associated with $|\rho_\infty|$ is the same as the function constructed by neural networks (GD limit as $t \to \infty$). However, the residual lies in a possibly much smaller space due to Theorem 5.2.1, which is the null space of the RKHS $\mathcal{K}_\infty$

$$K_\infty(x, \tilde{x}) = \int \left(\|\Theta\|^2 \mathbf{1}_{x^T\Theta \geq 0} \mathbf{1}_{\tilde{x}^T\Theta \geq 0} x^T\tilde{x} + \sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta) \right) |\rho_\infty|(d\Theta).$$

In other words, as the learned adaptive basis $\mathcal{H}_\infty$ (from GD) depends on the data distribution and the task f_* implicitly, it has the advantage of representing f_* by squeezing the residual into a smaller subspace in the null space of $\mathcal{H}_\infty$. A pictural illustration can be found in Fig. 5.2. This representation and approximation benefit helps with explaining the better interpolation results obtained by neural networks [199, 13, 110, 12]: (1) the adaptive basis is tailored for the task f_*, thus the residual/interpolation error lies in a smaller space; (2) in view of the ODE in Corollary 5.4.2, the second layer of NN adds *implicit regularization* to the smallest eigenvalues of K_t, thus improving the converging speed of Δ_t to zero.

Before concluding this section, we remark that a similar result holds for the ERM problem (5.1.6). As we shall discuss in the next section, the gap between $\mathcal{H}_\infty$ and $\mathcal{K}_\infty$ can be large, even for the ERM problem.

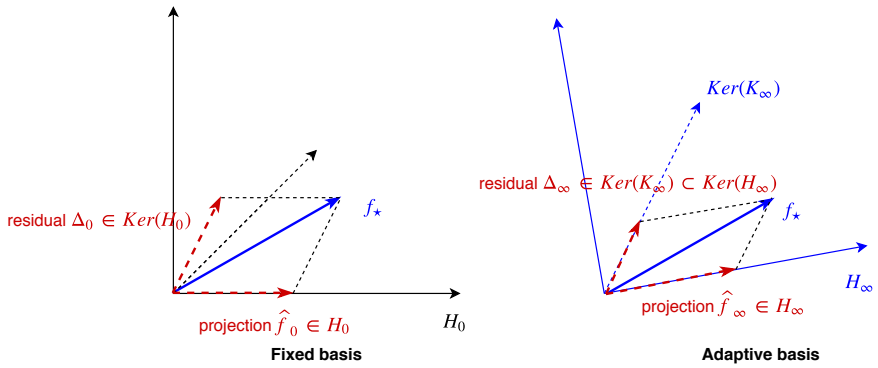


Figure 5.2: Illustration of 5.2.1: fixed basis vs. adaptive learned basis. In classic statistics, one specifies the fixed function space/basis H_0 then decompose f_* into the projection $\hat{f}_0$ and residual $\Delta_0 \in \text{Ker}(H_0)$. However, for GD on NN, one learns the adaptive basis H_∞ that depends on f_*. Therefore, the residual Δ_∞ lies in a subspace of $\text{Ker}(H_\infty)$.

5.3 Implications of the Adaptive Theory

In this section, we will discuss some direct implications of the adaptive kernel theory for neural networks established in this paper.

Example: Gap in Spaces $\mathcal{H}_\infty$ and $\mathcal{K}_\infty$. In Theorem 5.2.1, it is established that $\text{Ker}(\mathcal{K}_\infty) \subset \text{Ker}(\mathcal{H}_\infty)$. We now construct a concrete case to illustrate the potentially significant gap in these two spaces as follows. Consider only one neuron with $m = 1$, solving ERM problem (5.1.6) with n samples, and $\mathbf{x}$ with dimension d. In this case, ρ_∞ is supported on only one point, noted as $\Theta_\infty \in \mathbb{R}^d$. Denote $X \in \mathbb{R}^{n\times d}$ as the data matrix, one can show that

$$H_\infty(X, X) = \underbrace{\sigma(X\Theta_\infty^T)}_{n\times 1}\underbrace{\sigma(X\Theta_\infty^T)^T}_{1\times n}$$

has rank 1. In contrast,

$$K_\infty(X, X) \succeq \underbrace{\text{diag}(\mathbf{1}_{X\Theta_\infty^T\geq 0})X}_{n\times d}\underbrace{X^T\text{diag}(\mathbf{1}_{X\Theta_\infty^T\geq 0})}_{d\times n}$$

can be of rank $d \wedge |\{i : x_i^T \Theta_\infty \geq 0\}|$. Hence the null space of K_∞ is much smaller than that of H_∞. The gap can be large for many other settings of (n, m, d).

Connections to Min-norm Interpolation. The following result establishes the connections between the solution of gradient descent on neural networks (at local stationarity), and the kernel ridgeless regression [13, 110, 83] with an adaptive kernel $\widehat{H}_\infty^\lambda$. Empirical evidence on the similarity between the interpolation with kernels and neural networks was discovered in [13]. The following proposition provides a novel way of studying the generalization property of neural networks via adaptive kernels.

Proposition 5.3.1 (Interpolation: Connection to Kernel Ridgeless Regression)**.** Consider the gradient flow dynamics on all the weights of the neural network function $f_t(x) = \sum_{j=1}^m w_j(t)\sigma(x^T u_j(t))$, to solve the ℓ_2-regularized ERM

$$\frac{1}{2n}\sum_{i=1}^n (y_i - f_t(x_i))^2 + \frac{\lambda}{2m}\sum_{j=1}^m \left[w_j(t)^2 + \|u_j(t)\|^2\right] \ .$$

Consider only the bounded assumption on initialization that $|w_j^2(0) - \|u_j\|^2(0)| < \infty$ for all $1 \leq j \leq m$. At stationarity, denote the signed measure as $\widehat{\rho}_\infty^\lambda$ and the corresponding adaptive kernel as $\widehat{H}_\infty^\lambda$. Then the neural network function at stationarity $\widehat{f}_\infty^{\mathrm{nn},\lambda}(x)$ satisfies,

$$\widehat{f}_\infty^{\mathrm{nn},\lambda}(x) = \widehat{H}_\infty^\lambda(x, X)\left[\frac{n}{m}\lambda \cdot I_n + \widehat{H}_\infty^\lambda(X, X)\right]^{-1} Y \ .$$

In the vanishing regularization $\lambda \to 0$ limit, the neural network function converges to the kernel ridgeless regression with the adaptive kernel, when $\widehat{H}_\infty(X, X) := \lim_{\lambda\to 0} \widehat{H}_\infty^\lambda$ exists,

$$\lim_{\lambda\to 0} \widehat{f}_\infty^{\mathrm{nn},\lambda}(x) = \widehat{H}_\infty(x, X)\widehat{H}_\infty(X, X)^+ Y = \widehat{f}_\infty^{\mathrm{rkhs}}(x).$$

Note that the generalization theory for the kernel ridgeless regression has been established

[110, 83]. Here the kernel $\widehat{H}_\infty(X, X)$ is data-adaptive (that adapts to f_*) learned along training, instead of being fixed and pre-specified.

Connections to Random Kitchen Sinks. Let us introduce two function spaces, with the base measure ρ_0 (fixed representation)

$$\Gamma_2(\rho_0) := \left\{ f(x) \mid f(x) = \int \sigma(x^T\Theta) w(\Theta) \rho_0(d\Theta), w \in L^2_{\rho_0} \right\}$$

$$\Gamma_1(\rho_0) := \left\{ f(x) \mid f(x) = \int \sigma(x^T\Theta) w(\Theta) \rho_0(d\Theta), w \in L^1_{\rho_0} \right\}$$

In random kitchen sinks studied in [137, 138], by assuming $f_* \in \Gamma_2(\rho_0)$ that lies in the RKHS, the approximation error can be controlled by the existence of the following function with $\theta_j, j \in [m]$ i.i.d. sampled from ρ_0

$$\widehat{f}(x) = \frac{1}{m} \sum_{j=1}^{m} \sigma(x^T\Theta_j) w(\Theta_j) \in \Gamma_1(\rho_0), \text{but } \widehat{f}(x) \notin \Gamma_2(\rho_0) \ .$$

Note that $\widehat{f}$ lies in a possibly much larger space $\Gamma_1(\rho_0)$ though the target only lies in $f_* \in \Gamma_2(\rho_0)$. Similarly for two-layer neural networks function $f_t(x)$ considered in [6, Section 2.3], the RKHS space $\Gamma_2(\rho_0)$ can be more restrictive compared to $f_t \in \Gamma_1(\rho_0)$.

In contrast, with the adaptive RKHS representation $\mathcal{H}_\infty$, we have shown that

$$f_\infty(x) \in \Gamma_1(|\rho_\infty|), \text{and } f_\infty(x) \in \Gamma_2(|\rho_\infty|) \ .$$

The extreme case of fully adaptive function space $\Gamma_2(|\rho_*|)$ is defined with ρ_* tailored for f_*, $f_* = \int \sigma(x^T\Theta)\rho_*(d\Theta)$. The adaptive representation learned by neural networks can be viewed as in between the fixed and the fully adaptive representation.

Adaptive Generalization Theory. Now we attempt to provide a new decomposition to study the generalization of NN via adaptive kernels. Recall we have shown that $\widehat{f}_\infty^{\text{rkhs}}(x) = \lim_{\lambda\to 0}\widehat{f}_\infty^{\text{nn},\lambda}(x) = \widehat{H}_\infty(x,X)\widehat{H}_\infty(X,X)^+Y$, where

$$\widehat{H}_\infty(x,\tilde{x}) := \int \sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta)\widehat{\rho}_\infty^{(n,m)}(d\Theta).$$

Define the population limit $\rho_\infty^{(m)}(d\Theta) := \lim_{n\to\infty}\widehat{\rho}_\infty^{(n,m)}$ and
$H_\infty(x,\tilde{x}) := \int \sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta)\rho_\infty^{(m)}(d\Theta)$. Denote the ridgeless regression with the population adaptive kernel H_∞,

$$f_\infty^{\text{rkhs}}(x) = H_\infty(x,X)H_\infty(X,X)^+Y.$$

Assume $(\mathbf{y}-f_*(\mathbf{x}))^2 \le \sigma^2$ a.s. (can be relaxed). One can derive the following decomposition for generalization.

Proposition 5.3.2 (Adaptive Generalization)**.**

$$\begin{aligned}\|\lim_{\lambda\to 0}\widehat{f}_\infty^{\text{nn},\lambda} - f_*\|_\mu^2 \precsim\ & \underbrace{\|\widehat{f}_\infty^{\text{rkhs}} - f_\infty^{\text{rkhs}}\|_\mu^2}_{\text{adaptive representation error}} + \underbrace{\|f_\infty - f_*\|_\mu^2}_{\text{adaptive approximation error}} \\ & + (n\|f_\infty - f_*\|_{\hat{\mu}}^2 + \sigma^2)\underbrace{\mathbb{E}_{\mathbf{x}\sim\mu}\|H_\infty(X,X)^{-1}H_\infty(X,\mathbf{x})\|^2}_{\text{adaptive variance}} \\ & + \underbrace{\|H_\infty(x,X)H_\infty(X,X)^{-1}f_\infty(X) - f_\infty(x)\|_\mu^2}_{\text{adaptive bias}}\end{aligned}$$

Note this result holds without requiring global optimization guarantees. The first term is the representation error, which corresponds to the closeness of the adaptive RKHS $\widehat{\mathcal{H}}_\infty$ (using empirical distribution) and $\mathcal{H}_\infty$ (using population distribution). The second term is the adaptive approximation error studied in the current paper. The third and fourth terms are the variance and bias expressions studied in [110, 83, 139], as if assuming the actual

function lies in $\mathcal{H}_\infty$. This decomposition suggests the possibility of studying generalization without explicit global understanding of the optimization, and providing rates that adapts to f_* without structural assumptions.

5.4 Time-varying Kernels and Evolution

In this section, we lay out the mathematical details on the time-varying kernels and the evolution of the signed measure ρ_t supporting the main results. In the meantime, we will discuss in depth the relevant literature motivating our proof ideas.

First, we describe the motivation behind the dynamic RKHS $\mathcal{K}_t$, and the GD kernel induced by the gradient descent dynamics. Extensions to multi-layer perceptrons is in Sec. 5.7.2.

Lemma 5.4.1 (Dynamic kernel of finite neurons GD)**.** Consider the approximation problem (5.1.1) with a neural network function (5.1.2), and the training process (5.1.3) with population distribution. Let $\Delta_t(x) = f_*(x) - f_t(x)$ be the residual. Define the time-varying kernel $K_t(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$,

$$K_t(x,\tilde{x}) = \sum_{j=1}^{m} \left[\sigma(x^T u_j(t))\sigma(\tilde{x}^T u_j(t)) + w_j(t)^2 \mathbf{1}_{x^T u_j(t)\geq 0} \mathbf{1}_{\tilde{x}^T u_j(t)\geq 0} x^T \tilde{x} \right]. \tag{5.4.1}$$

Then the residual Δ_t driven by the GD dynamics satisfies,

$$\frac{d\mathbf{E}_{\mathbf{x}}\left[\frac{1}{2}\Delta_t(\mathbf{x})^2\right]}{dt} = -\mathbf{E}_{\mathbf{x},\tilde{\mathbf{x}}}\left[\Delta_t(\mathbf{x}) K_t(\mathbf{x},\tilde{\mathbf{x}}) \Delta_t(\tilde{\mathbf{x}})\right]. \tag{5.4.2}$$

When running GD to solve the empirical risk minimization (ERM), the dynamics of the finite-dimensional sample residual $\|\Delta_t\|_{\hat{\mu}}^2$ has been established in [93, 52]. Here we generalize the result to optimize the weights of both layers, and to solve the infinite-dimensional population approximation problem rather than the empirical risk minimization problem. For

a general loss function $\ell(y, f)$ with curvature (say, logistic loss), similar results hold under slightly stronger conditions.

Corollary 5.4.1. Consider a general loss function $\ell(y, f)$ that is α-strongly convex in the second argument f, with K_t defined in (5.4.1). Assume in addition $\frac{1}{n}K_t(X, X) \in \mathbb{R}^{n\times n}$ has smallest eigenvalue $\lambda_t > 0$. Define $\Delta_t(x_i) := \frac{\partial \ell(y_i, f_t(x_i))}{\partial f}$, then we have for all $f_* : \mathbb{R}^d \to \mathbb{R}$,

$$\frac{d\widehat{\mathbb{E}}\left[\ell(\mathbf{y}, f_t(\mathbf{x}))\right]}{dt} = -\widehat{\mathbb{E}}_{\mathbf{x},\tilde{\mathbf{x}}}\left[\Delta_t(\mathbf{x})K_t(\mathbf{x}, \tilde{\mathbf{x}})\Delta_t(\tilde{\mathbf{x}})\right] \leq -2\alpha\lambda_t \cdot \widehat{\mathbb{E}}\left[\ell(\mathbf{y}, f_t(\mathbf{x})) - \ell(\mathbf{y}, f_*(\mathbf{x}))\right] \ .$$

5.4.1 Initialization, Rescaling and K_0

Now we describe the initialization and rescaling schemes used in the main theorems. Rewrite (5.1.1) according to the signs of the second layer weights

$$f_t(x) := \sum_{j=1}^{m_+} w_{+,j}(t)\sigma(x^T u_{+,j}(t)) + \sum_{j=1}^{m_-} w_{-,j}(t)\sigma(x^T u_{-,j}(t)).$$

Initialization. We consider the "infinitesimal" initialization drawn *i.i.d.* from two probability measures $\rho_{+,0}$ and $\rho_{-,0}$ that do not depend on m:

$$u_{+,j}(0) = \frac{1}{\sqrt{m}}\Theta_{+,j} \text{ where } \Theta_{+,j} \sim \rho_{+,0} \ , \ u_{-,j}(0) = \frac{1}{\sqrt{m}}\Theta_{-,j} \text{ where } \Theta_{-,j} \sim \rho_{-,0} \ . \quad (5.4.3)$$

Here $m = m_+ + m_-$ with $m_+ \asymp m_-$. The $1/\sqrt{m}$ rescaling factor turns out to be crucial when defining the infinite neurons limit for the evolution of signed measures. Remark that such initialization is w.l.o.g., and accounts for the infinitesimal nature used in practice when the number of neurons grows. For the second layer weights, we impose the "balanced condition" motivated by [113],

$$w_{+,j}(0) = \|u_{+,j}(0)\| \geq 0 \ , \ w_{-,j}(0) = -\|u_{-,j}(0)\| \leq 0. \quad (5.4.4)$$

It turns out that with such initialization, the balanced condition holds throughout the training process induced by gradient flow, which is useful for the main theorems. Interestingly, in the proof of Proposition 5.3.1, we show that such balanced condition always holds at stationarity when training neural networks with ℓ_2 regularization, even for unbalanced initialization.

Proposition 5.4.1 (Balanced condition)**.** For $u_{+,j}(t)$, $u_{-,j}(t)$, $w_{+,j}(t)$ and $w_{-,j}(t)$, and the initialization specified above, at any time t, we have

$$w_{+,j}(t) = \|u_{+,j}(t)\|, \quad w_{-,j}(t) = -\|u_{-,j}(t)\|.$$

Rescaling. To prepare for the distribution dynamic theory in the next section, we introduce a parameter rescaling with the $\sqrt{m}$ factor. Let $\theta_{+,j}(t) = \sqrt{m}w_{+,j}(t)$ and $\theta_{-,j}(t) = \sqrt{m}w_{-,j}(t)$, also define $\Theta_{+,j}(t) = \sqrt{m}u_{+,j}(t)$ and $\Theta_{-,j}(t) = \sqrt{m}u_{-,j}(t)$ sampled from $\rho_{+,0}$ and $\rho_{-,0}$ at $t = 0$. Under this representation,

$$f_t(x) = \frac{1}{m}\sum_{j=1}^{m_+} \theta_{+,j}(t)\sigma(x^T\Theta_{+,j}(t)) + \frac{1}{m}\sum_{j=1}^{m} \theta_{-,j}(t)\sigma(x^T\Theta_{-,j}(t)). \tag{5.4.5}$$

By the positive homogeneity of ReLU, we have the corresponding dynamics on the rescaled parameters,

$$\frac{d\theta_{\cdot,j}}{dt} = \sqrt{m}\frac{dw_{\cdot,j}}{dt} = -\sqrt{m}\mathbf{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, f(\mathbf{x}))}{\partial f}\sigma(\mathbf{x}^T u_{\cdot,j})\right] = -\mathbf{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, f(\mathbf{x}))}{\partial f}\sigma(\mathbf{x}^T \Theta_{\cdot,j})\right], \tag{5.4.6}$$

$$\begin{aligned}\frac{d\Theta_{\cdot,j}}{dt} &= \sqrt{m}\frac{du_{\cdot,j}}{dt} = -\sqrt{m}\mathbf{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, f(\mathbf{x}))}{\partial f} w_{\cdot,j}\mathbf{1}_{\mathbf{x}^T u_{\cdot,j}\geq 0}\mathbf{x}\right] \\ &= -\mathbf{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, f(\mathbf{x}))}{\partial f}\theta_{\cdot,j}\mathbf{1}_{\mathbf{x}^T\Theta_{\cdot,j}\geq 0}\mathbf{x}\right].\end{aligned} \tag{5.4.7}$$

Define at time t

$$\rho_{+,t} := \frac{1}{m}\sum_{j=1}^{m_+} \delta_{\Theta_{+,j}(t)}, \quad \rho_{-,t} := \frac{1}{m}\sum_{j=1}^{m_-} \delta_{\Theta_{-,j}(t)} \tag{5.4.8}$$

as the empirical distribution over neurons on the parameter space Θ. The $\rho_{+,t}$ and $\rho_{-,t}$ converge weakly to proper distributions in the infinite neurons limit $m \to \infty$, see e.g. [6, 117]. Through the balanced condition in Proposition 5.4.1 and Proposition 5.7.1, we know (by substituting θ_j by $\|\Theta_j\|$)

$$f_t(x) = \int \|\Theta\|\sigma(x^T\Theta)\rho_t(d\Theta), \text{ where the signed measure } \rho_t := \rho_{+,t} - \rho_{-,t}. \tag{5.4.9}$$

The above motivates the study of the RKHS $\mathcal{H}_t$ as in Theorem 5.2.1, with the kernel

$$H_t(x, \tilde{x}) = \int \|\Theta\|^2\sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta)|\rho_t|(d\Theta). \tag{5.4.10}$$

To conclude this section, we provide the explicit formula for the initial kernel matrix K_0 under such infinitesimal random initialization. Specifically, consider the initialization with w_j being $\pm 1/\sqrt{m}$ with equal chance and $u_i \sim N(\mathbf{0}, 1/m \cdot \mathbf{I}_d)$ *i.i.d.* sampled. The initial kernel K_0 has the following expression, in the infinite neurons limit.

Lemma 5.4.2 (Fixed Kernel)**.** With initialization specified above, consider w.l.o.g. $\|x\| = \|\tilde{x}\| = 1$, and denote $\Theta \sim \pi$ as the isotropic Gaussian $N(\mathbf{0}, \mathbf{I}_d)$. By the strong law of large number, we have almost surely,

$$\begin{aligned}\lim_{m\to\infty} K_0(x, \tilde{x}) &= \mathbf{E}_{\Theta\sim\pi}\left[\sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta) + \mathbf{1}_{x^T\Theta>0}\mathbf{1}_{\tilde{x}^T\Theta>0}x^T\tilde{x}\right] \\ &= \left[\frac{\pi - \arccos(t)}{\pi}t + \frac{\sqrt{1-t^2}}{2\pi}\right], \quad \text{where } t = x^T\tilde{x}.\end{aligned}$$

Much known results [17, 137, 6, 38, 40] on the connection between RKHS and two-layer

NN focus on some fixed kernel, such as K_0. To instantiate useful statistical rates, one requires f_* to lie in the corresponding pre-specified RKHS $\mathcal{K}_0$, which is non-verifiable in practice. In contrast, the dynamic kernel is less studied. We will establish a dynamic and adaptive kernel theory defined by GD, without making any structural assumptions on f_* other than $f_* \in L^2_\mu$.

5.4.2 Evolution of ρ_t

In this section, we derive the evolution of the signed measure ρ_t defined by the neurons at the training t, which in turn determines the dynamic kernel K_t defined in (5.4.1). To generalize the result to the case of infinite neurons, we follow and borrow tools from the mean-field characterization [117, 144, 99]. The rescaling described in the previous section proves handy when defining such infinite neurons limit. We define the velocity field driven by the regression task and the interaction among neurons,

$$V(\Theta) = \mathbf{E}[\mathbf{y}\sigma(\mathbf{x}^T\Theta)], \quad U(\Theta, \tilde{\Theta}) = -\mathbf{E}[\sigma(\mathbf{x}^T\Theta)\sigma(\mathbf{x}^T\tilde{\Theta})]. \tag{5.4.11}$$

The following theorem casts the training process as distribution dynamics on $\rho_{+,t}, \rho_{-,t}$.

Lemma 5.4.3 (Dynamic Kernel and Evolution)**.** Consider the approximation problem (5.1.1), and the gradient flow as the training dynamic (5.1.3). For $\rho_{+,t}$, $\rho_{-,t}$ and ρ_t defined in (5.4.8) with possibly infinite neurons, we have the following PDE characterization on distribution dynamics of $\rho_{+,t}, \rho_{-,t}$

$$\begin{aligned}
\partial_t \rho_{+,t}(\Theta) &= -\nabla_\Theta \cdot \left[\rho_{+,t}(\Theta) \cdot \|\Theta\| \left(\nabla_\Theta V(\Theta) + \nabla_\Theta \int U(\Theta, \tilde{\Theta}) \|\tilde{\Theta}\| \rho_t(d\tilde{\Theta})\right)\right], \\
\partial_t \rho_{-,t}(\Theta) &= \nabla_\Theta \cdot \left[\rho_{-,t}(\Theta) \cdot \|\Theta\| \left(\nabla_\Theta V(\Theta) + \nabla_\Theta \int U(\Theta, \tilde{\Theta}) \|\tilde{\Theta}\| \rho_t(d\tilde{\Theta})\right)\right].
\end{aligned} \tag{5.4.12}$$

Moreover, the GD kernel K_t is defined as

$$K_t(x,\tilde{x}) = \int \left(\|\Theta\|^2 \mathbf{1}_{x^T\Theta\geq 0}\mathbf{1}_{\tilde{x}^T\Theta\geq 0} x^T\tilde{x} + \sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta) \right) |\rho_t|(d\Theta). \tag{5.4.13}$$

Remark 5.4.1. As in [117, 144], let's first show that in the infinite neurons limit $m \to \infty$, $\rho_{+,t}, \rho_{-,t}$ are properly defined, with Eqn. (5.4.12) characterizing the distribution dynamics. For simplicity, we assume the initialization $\rho_{+,0}, \rho_{-,0}$ is with bounded support. Add the superscript m, $\rho_{+,t}^{(m)}, \rho_{-,t}^{(m)}, \rho_t^{(m)}$ to (5.4.8) to indicate their dependence on m. Consider that $\nabla_\Theta V(\Theta)$, $\nabla_\Theta U(\Theta, \tilde{\Theta})$ in (5.4.11) are bounded and uniform Lipchitz continuous as in [117, A3]. With the same proof as in [117, Theorem 3], one can show that with $m \to \infty$, the initial distribution $\rho_0^{(m)} \xrightarrow{d} \rho_0 = \rho_{+,0} - \rho_{-,0}$ by law of large number. And by the solution's continuity w.r.t. the initial value, we have $\rho_t^{(m)} \xrightarrow{d} \rho_t$ as $m \to \infty$ well defined, for any fixed t.

Note that our problem setting is slightly different from that in [117], where the authors consider the NN with fixed second layer weights to be $1/m$. We reiterate that the re-parameterization via θ and Θ is crucial: (1) weights on both layers are optimized following the gradient flow; (2) infinitesimal random initialization is employed in practice. In the setting of [117, Eqn. (3)], the training process is slightly different from the vanilla GD on weights, with an additional m factor in the velocity term. This subtlety is also mentioned in [144]. In short, the rescaling looks at the dynamics where Θ's are on the invariant scale as $m \to \infty$ for any fixed effective time t (that does not depend on m). Here we analyze the exact gradient flow on the two-layer weights, with infinitesimal random initialization as in practice, resulting in a different velocity field (5.4.11) compared to that in [117].

The proof of Theorem 5.2.1 makes use of (5.4.9)-(5.4.10) and the stationary condition implied by Lemma 5.4.3. The balanced condition is crucial in both Theorem 5.2.1 and Proposition 5.3.1. The details of the proof are deferred to Section 5.6.

5.4.3 Two RKHS: $\mathcal{K}_\infty$ and $\mathcal{H}_\infty$

In this section we compare the two adaptive RKHS appeared $\mathcal{K}_\infty$ in (5.4.13), and $\mathcal{H}_\infty$ in (5.4.10). The comparison will lead to the proof of Theorem 5.2.1. We start with generalizing Lemma 5.4.1 with the possibly infinite neurons case via the distribution dynamics in (5.4.12).

Corollary 5.4.2. Consider the same setting as in Lemma 5.4.1 with possibly infinite neurons NN (5.4.9), and the training process (5.4.12). Define the time-varying kernel matrix $K_t(\cdot,\cdot) : \mathcal{X}\times\mathcal{X}\to\mathbb{R}$, with the signed measure ρ_t follows (5.4.12)

$$K_t(x,\tilde{x}) = \int \left(\|\Theta\|^2 \mathbf{1}_{x^T\Theta\geq 0}\mathbf{1}_{\tilde{x}^T\Theta\geq 0} x^T\tilde{x} + \sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta)\right)|\rho_t|(d\Theta) \tag{5.4.14}$$

$$=: K_t^{(0)}(x,\tilde{x}) + K_t^{(1)}(x,\tilde{x}). \tag{5.4.15}$$

Then we still have $d\mathbf{E}_{\mathbf{x}}\left[\frac{1}{2}\Delta_t(\mathbf{x})^2\right]/dt = -\mathbf{E}_{\mathbf{x},\tilde{\mathbf{x}}}\left[\Delta_t(\mathbf{x})K_t(\mathbf{x},\tilde{\mathbf{x}})\Delta_t(\tilde{\mathbf{x}})\right]$.

It turns out that the kernels K_∞ and H_∞, defined in (5.2.4) and (5.1.7) respectively, satisfy the following inclusion property.

Proposition 5.4.2. Consider the training process reaches any stationarity $\rho_\infty = \rho_{+,\infty} - \rho_{-,\infty}$ with compact support within radius D and finite total variation. We have

$$K_\infty \succeq K_\infty^{(0)} \succeq K_\infty^{(1)} \succeq \frac{1}{D^2}H_\infty, \tag{5.4.16}$$

with $K_\infty^{(0)}, K_\infty^{(1)}$ defined in (5.4.14). Combining with the fact that $H_\infty \neq K_\infty$ implies

$$\mathrm{Ker}(\mathcal{K}_\infty) \subset \mathrm{Ker}(\mathcal{H}_\infty).$$

The proof of Theorem 5.2.1 uses the following fact: when reaching stationarity, due to

the ODE defined by GD in Lemma 5.4.1, the residual must satisfy

$$\Delta_\infty(x) = f_*(x) - f_\infty(x) \in \mathrm{Ker}(\mathcal{K}_\infty). \tag{5.4.17}$$

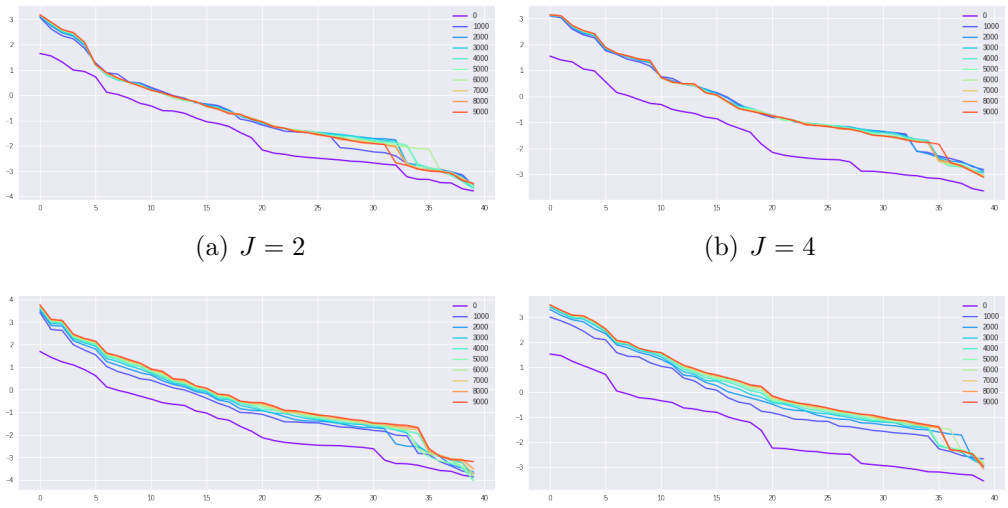
The proof of Proposition 5.4.2 and Theorem 5.2.1 are deferred to Section 5.6.

5.5 Experiments

We run experiments to illustrate the spectral decay of the dynamic kernels defined in K_t over time t. The exercise is to quantitatively showcase that during neural network training, one does learn the data-adaptive representation, which is task-specific depending on the true complexity of f_*. The training process is the same as the one we theoretically analyze: vanilla gradient descent on a two-layer NN of m neurons, with infinitesimal random initialization scales as $1/\sqrt{m}$.

The first experiment is a synthetic exercise with well-specified models. We generate $\{x_i\}_{i=1}^{50}$ from isotropic Gaussian in $\mathbb{R}^5$, and $y_i = f_*(x_i) = \sum_{j=1}^{J} w_j^* \sigma(x_i^T u_j^*)$ with different J. In other words, we choose different target f_* (task complexity) by varying J. We select $m = 500$ in our experiment. The top 80% of the sorted eigenvalues of the kernel matrix K_t along the GD training process are shown in Fig. 5.3. The x-axis is the index of eigenvalues in descending order, and the y-axis is the logarithmic values of the corresponding eigenvalues. Different color indicates the spectral decay of the K_t at different training time t. The eigenvalue-decays stabilize over time t means that the training process approaches stationarity. As we can see with f_* belongs to the NN family, the eigenvalues of the kernel matrix, in general, become larger during the training process. For a more complicated target function, it takes longer to reach stationarity.

The second experiment is another synthetic test on fitting random labels. We generate $\{x_i\}_{i=1}^{50}$ from isotropic Gaussian in $\mathbb{R}^5$, as y_i takes ± 1 with equal chance. We select $m =$



(a) $J = 2$ (b) $J = 4$

(c) $J = 8$ (d) $J = 16$

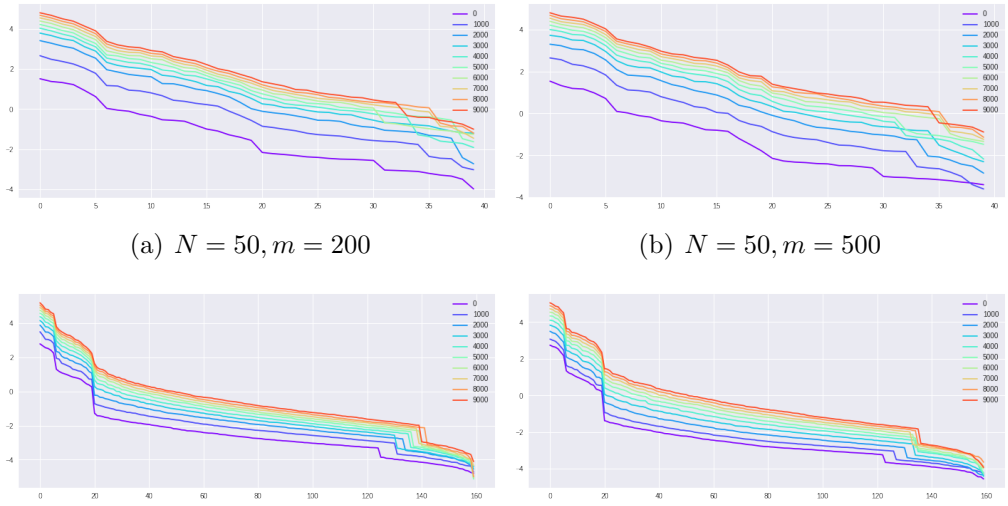
Figure 5.3: Log of the sorted top 80% eigenvalues of kernel matrix along training with different f_*

$200, 500$, and $n = 50, 200$ to investigate those parameters' influence on the kernel K_t. We want to point out two observations. First, fixed n, we investigate over-parametrized models ($m = 200, 500$ large). Shown from Fig. 5.4 along the row, the kernels for different m's behave much alike. In other words, in the infinite neurons limit, the kernel will stabilize. Second, fixed m, we vary the number of samples n, to simulate different interpolation hardness. As seen from Fig. 5.4 along the column, the kernels and the convergence over time are distinct, reflecting the different difficulty of the interpolation.

The third experiment (Fig. 5.5) is regression using the MNIST dataset with different sample size $n = 50, 200$. We hope to investigate the influence of sample size on the kernel matrix along the training process. For a larger sample size N, it takes longer to reach stationarity.

5.6 Main Proofs

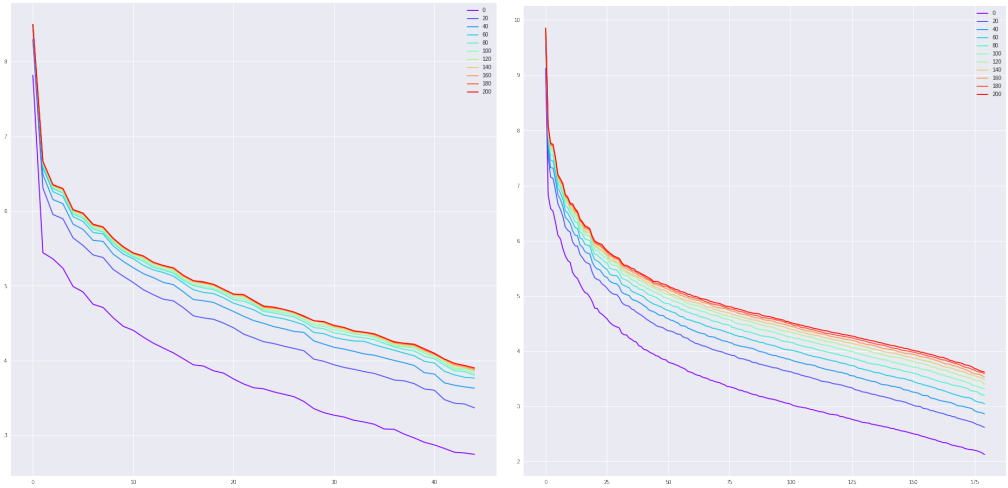
Proof of Theorem 5.2.1. From the definition, we have $\mathcal{T}^* p \in \mathcal{H}_\infty$ for any $p \in L^2_{|\rho_\infty|}$, and $\mathcal{T}^*$ is a surjective mapping. Suppose that $\widehat{g} \in \mathcal{H}_\infty$ is a minimizer of (5.2.2), then we claim that



(a) $N = 50, m = 200$ (b) $N = 50, m = 500$

(c) $N = 200, m = 200$ (d) $N = 200, m = 500$

Figure 5.4: Log of the sorted top 80% eigenvalues of kernel matrix along training with random labels.



(a) $N = 50$ (b) $N = 200$

Figure 5.5: Log of sorted top 90% eigenvalues of kernel matrix along training process for mnist

for any $p \in L^2_{|\rho_\infty|}$, one must have

$$\langle f_* - \widehat{g}, \mathcal{T}^* p \rangle_\mu = 0, \quad \forall p \in L^2_{|\rho_\infty|}. \tag{5.6.1}$$

This claim can be seen from the following argument. Suppose not, then for p that violates the above, construct

$$\widehat{g}_\epsilon = \widehat{g} + \epsilon \mathcal{T}^* p \in \mathcal{H}_\infty,$$

we know

$$\|f_* - \widehat{g}_\epsilon\|_\mu^2 = \|f_* - \widehat{g}\|_\mu^2 - 2\epsilon \langle f_* - \widehat{g}, \mathcal{T}^* p \rangle_\mu + \epsilon^2 \|\mathcal{T}^* p\|_\mu^2. \tag{5.6.2}$$

For ϵ with the same sign as $\langle f_* - \widehat{g}, \mathcal{T}^* p \rangle_\mu \neq 0$ and small enough, one can see that $\|f_* - \widehat{g}_\epsilon\|_\mu^2 < \|f_* - \widehat{g}\|_\mu^2$ which validates that $\widehat{g}$ is a minimizer. From the same argument, one can see that $\widehat{g}$ is a minimizer if and only if (5.6.1) holds, in other words,

$$\langle \mathcal{T}(f_* - \widehat{g}), p \rangle_{|\rho_\infty|} = \langle f_* - \widehat{g}, \mathcal{T}^* p \rangle_\mu = 0 \tag{5.6.3}$$

From PDE characterization (5.4.12) with ReLU activation, one knows that

$$\begin{aligned} V(\Theta) &= \mathbf{E}[\mathbf{y}\sigma(\mathbf{x}^T\Theta)] = \mathbb{E}[f_*(\mathbf{x})\sigma(\mathbf{x}^T\Theta)] \\ U(\Theta, \tilde{\Theta}) &= -\mathbf{E}[\sigma(\mathbf{x}^T\Theta)\sigma(\mathbf{x}^T\tilde{\Theta})], \end{aligned}$$

and the expression for the velocity field

$$\begin{aligned} &\|\Theta\| \left(\nabla_\Theta V(\Theta) + \nabla_\Theta \int U(\Theta, \tilde{\Theta}) \|\tilde{\Theta}\| \rho_t(d\tilde{\Theta}) \right) \\ &= \|\Theta\| \left(\int f_*(x) x \mathbf{1}_{x^T\Theta > 0} \mu(dx) - \int \int x \mathbf{1}_{x^T\Theta > 0} \sigma(x^T\tilde{\Theta}) \|\tilde{\Theta}\| \rho_\infty(d\tilde{\Theta}) \mu(dx) \right). \end{aligned}$$

We know that any stationary point $\left(\rho_{+,\infty}, \rho_{-,\infty}\right)$ has the following property [117]:

$$\operatorname{supp}(\rho_\infty) \subseteq \left\{\Theta : \int f_*(x) x \mathbf{1}_{x^T\Theta>0}\mu(dx) = \int\int x\mathbf{1}_{x^T\Theta>0}\sigma(x^T\tilde{\Theta})\|\tilde{\Theta}\|\rho_\infty(d\tilde{\Theta})\mu(dx)\right\}. \tag{5.6.4}$$

Multiplying both sides by $\|\Theta\|\Theta^T$ and recall the property of ReLU, the above condition implies that for all $\Theta \in \operatorname{supp}(\rho_\infty)$, we have

$$\int f_*(x)\|\Theta\|\sigma(x^T\Theta)\mu(dx) = \int\int \|\Theta\|\sigma(x^T\Theta)\sigma(x^T\tilde{\Theta})\|\tilde{\Theta}\|\rho_\infty(d\tilde{\Theta})\mu(dx). \tag{5.6.5}$$

One can see the stationary condition on ρ_∞ (fixed points of the dynamics) (5.6.5) translates to

$$\mathcal{T} f_*(\Theta) = \left(\mathcal{T}\mathcal{T}^\star \frac{d\rho_\infty}{d|\rho_\infty|}\right)(\Theta), \quad \forall \Theta \in \operatorname{supp}(\rho_\infty). \tag{5.6.6}$$

Here the function $\frac{d\rho_\infty}{d|\rho_\infty|}$ is the Radon-Nikodym derivative. In addition, one can easily verify that, as ρ_∞ has bounded total variation

$$\frac{d\rho_\infty}{d|\rho_\infty|} \in L^2_{|\rho_\infty|}.$$

Therefore, combining all the above, one knows that

$$f_\infty(x) = \int \|\Theta\|\sigma(x^T\Theta)\rho_\infty(d\Theta) = \mathcal{T}^\star \frac{d\rho_\infty}{d|\rho_\infty|} \in \mathcal{H}_\infty$$

and that for any $p \in L^2_{|\rho_\infty|}$

$$\langle f_* - f_\infty, \mathcal{T}^* p\rangle_\mu = \langle \mathcal{T}(f_* - f_\infty), p\rangle_{|\rho_\infty|} \tag{5.6.7}$$

$$= \left\langle \mathcal{T} f_* - \mathcal{T}\mathcal{T}^\star \frac{d\rho_\infty}{d|\rho_\infty|}, p \right\rangle_{|\rho_\infty|} \tag{5.6.8}$$

$$= \int \left(\mathcal{T} f_* - \mathcal{T}\mathcal{T}^\star \frac{d\rho_\infty}{d|\rho_\infty|} \right)(\Theta) |\rho_\infty|(d\Theta) = 0 \qquad \text{due to (5.6.6)} \tag{5.6.9}$$

We have proved that $f_\infty = \mathcal{T}^\star \frac{d\rho_\infty}{d|\rho_\infty|}$ satisfies normal condition for being a minimizer to (5.2.2). □

Proof of Proposition 5.4.2. The first inequality in (5.4.16) is trivial. For the second inequality, it suffices to show for any $c = (c_1, \ldots, c_p)^T$, $x_1, \ldots, x_p$, Θ, we have

$$\sum_{i,j} c_i c_j \|\Theta\|^2 x_i^T x_j \mathbf{1}_{x_i^T\Theta>0} \mathbf{1}_{x_j^T\Theta>0} \geq \sum_{i,j} c_i c_j \sigma(x_i^T\Theta)\sigma(x_j^T\Theta) \tag{5.6.10}$$

The RHS equals

$$\sum_{i,j} c_i c_j x_i^T \Theta x_j^T \Theta \mathbf{1}_{x_i^T\Theta>0} \mathbf{1}_{x_j^T\Theta>0} = \left(\sum_i c_i x_i^T \Theta \mathbf{1}_{x_i^T\Theta>0} \right)^2 \tag{5.6.11}$$

$$= \langle \Theta, \sum_i c_i x_i \mathbf{1}_{x_i^T\Theta>0} \rangle^2 \leq \|\Theta\|^2 \left\| \sum_i c_i x_i \mathbf{1}_{x_i^T\Theta>0} \right\|^2 = \text{LHS}. \tag{5.6.12}$$

For the last inequality, with compactness condition on ρ_∞, we have

$$\sum_{i,j} c_i c_j \int \|\Theta\|^2 \sigma(x_i^T\Theta)\sigma(x_j^T\Theta)|\rho_\infty|(\Theta) \leq D^2 \sum_{i,j} c_i c_j \int \sigma(x_i^T\Theta)\sigma(x_j^T\Theta)|\rho_\infty|(\Theta). \tag{5.6.13}$$

Therefore, $D^2 K_\infty^{(1)} \succeq H_\infty$.

□

Proof of Theorem 5.2.1. Let us rewrite Corollary 5.4.2 into

$$\frac{d}{dt}\|\Delta_t\|_\mu^2 = -2\langle \Delta_t, \mathcal{K}_t \Delta_t\rangle_\mu = -2\|\mathcal{K}_t^{1/2}\Delta_t\|_\mu^2, \tag{5.6.14}$$

here $\mathcal{K}_t : L_\mu^2(x) \to L_\mu^2(x)$ denotes the integral operator associated with K_t,

$$(\mathcal{K}_t f)(x) := \int K_t(x,\tilde{x}) f(\tilde{x}) \mu(d\tilde{x}). \tag{5.6.15}$$

From (5.6.14)

$$\frac{d}{dt}\|\Delta_\infty\|_\mu^2 = -2\|\mathcal{K}_\infty^{1/2}\Delta_\infty\|_\mu^2, \tag{5.6.16}$$

we know that the RHS equals zero implies

$$\begin{aligned} \|\mathcal{K}_\infty^{1/2}\Delta_\infty\|_\mu^2 &= 0 \\ \langle \mathcal{K}_\infty^{1/2} g, \Delta_\infty\rangle_\mu = \langle g, \mathcal{K}_\infty^{1/2}\Delta_\infty\rangle_\mu &= 0, \quad \forall g \in L_\mu^2. \end{aligned}$$

This further implies Δ_∞ lies in the kernel of RKHS $\mathcal{K}_\infty$ as $\mathcal{K}_\infty = \{\mathcal{K}_\infty^{1/2} g : g \in L_\mu^2\}$. □

Proof of Proposition 5.3.1. The gradients on the original parameters are,

$$\begin{aligned} \frac{dw_j(t)}{dt} &= -\widehat{\mathbf{E}}\left[\frac{\partial \ell(\mathbf{y}, f_t)}{\partial f}\sigma(\mathbf{x}^T u_j(t))\right] - \frac{1}{m}\lambda w_j(t), \\ \frac{du_j(t)}{dt} &= -\widehat{\mathbf{E}}\left[\frac{\partial \ell(\mathbf{y}, f_t)}{\partial f} w_j(t) \mathbf{1}_{\mathbf{x}^T u_j(t)\geq 0}\mathbf{x}\right] - \frac{1}{m}\lambda u_j(t). \end{aligned}$$

Clearly, on the rescaled parameter, the following holds

$$\begin{aligned} \frac{d\theta_j}{dt} = \sqrt{m}\frac{dw_j}{dt} &= -\widehat{\mathbf{E}}\left[(f_t(\mathbf{x}) - \mathbf{y})\sigma(\mathbf{x}^T\Theta_j(t))\right] - \frac{1}{m}\lambda\theta_j, \\ \frac{d\Theta_j}{dt} = \sqrt{m}\frac{du_j}{dt} &= -\widehat{\mathbf{E}}\left[(f_t(\mathbf{x}) - \mathbf{y})\theta_j \mathbf{1}_{\mathbf{x}^T\Theta_j\geq 0}\mathbf{x}\right] - \frac{1}{m}\lambda\Theta_j. \end{aligned}$$

Multiply the first equation by θ_j, and the second equation by θ_j^T, take the difference, we can verify that

$$\frac{d(\theta_j^2 - \|\Theta_j\|^2)}{dt} = -\lambda/m(\theta_j^2 - \|\Theta_j\|^2) \tag{5.6.17}$$

$$\theta_j(t)^2 - \|\Theta_j(t)\|^2 = \Big(\theta_j(0)^2 - \|\Theta_j(0)\|^2\Big)\exp(-\lambda t/m) \ . \tag{5.6.18}$$

Therefore the balanced condition still holds at stationarity for arbitrary bounded initialization,

$$\theta_j(\infty)^2 - \|\Theta_j(\infty)\|^2 = 0, \forall j.$$

Now the optimality condition for the velocity field reads the following, for any $\Theta_j(\infty) \in \mathrm{supp}(\widehat{\rho}_\infty^\lambda)$ (we abbreviate the ∞ in the following display, note $\tilde{\theta}(\infty)$ corresponds to the second layer weights w.r.t. to $\tilde{\Theta}(\infty)$)

$$\theta_j \widehat{\mathbb{E}}[\mathbf{y}\mathbf{1}_{\mathbf{x}^T\Theta_j \geq 0}\mathbf{x}] = \theta_j \int \tilde{\theta}\widehat{\mathbb{E}}[\mathbf{1}_{\mathbf{x}^T\Theta_j \geq 0}\mathbf{x}\sigma(\mathbf{x}^T\tilde{\Theta})]|\widehat{\rho}_\infty^\lambda|(d\tilde{\Theta}) + \frac{1}{m}\lambda\Theta_j$$

$$\text{Multiply by } \Theta_j^T, \ \ \theta_j \widehat{\mathbb{E}}[\mathbf{y}\sigma(\mathbf{x}^T\Theta_j)] = \int \theta_j\tilde{\theta}\widehat{\mathbb{E}}[\sigma(\mathbf{x}^T\Theta_j)\sigma(\mathbf{x}^T\tilde{\Theta})]|\widehat{\rho}_\infty^\lambda|(d\tilde{\Theta}) + \frac{\lambda}{m}\|\Theta_j\|^2$$

$$\theta_j \widehat{\mathbb{E}}[\mathbf{y}\sigma(\mathbf{x}^T\Theta_j)] = \int \theta_j\tilde{\theta}\widehat{\mathbb{E}}[\sigma(\mathbf{x}^T\Theta_j)\sigma(\mathbf{x}^T\tilde{\Theta})]|\widehat{\rho}_\infty^\lambda|(d\tilde{\Theta}) + \lambda\int \theta_j\tilde{\theta}\mathbf{1}_{\tilde{\Theta}=\Theta_j}|\widehat{\rho}_\infty^\lambda|(d\tilde{\Theta})$$

where the last step uses the condition $\theta_j^2(\infty) = \|\Theta_j(\infty)\|^2$, and the fact that $|\widehat{\rho}_\infty^\lambda| = \frac{1}{m}\sum_{j=1}^m \delta_{\Theta_j}$ and

$$\int \theta_j\tilde{\theta}\mathbf{1}_{\tilde{\Theta}=\Theta_j}|\widehat{\rho}_\infty^\lambda|(d\tilde{\Theta}) = \frac{1}{m}\theta_j^2 = \frac{1}{m}\|\Theta_j\|^2.$$

In the matrix form, where $\widehat{\rho}_{\infty}^{\lambda} = \frac{1}{m}\sum_{l\in[m]} \mathrm{sgn}(\theta_l)\delta_{\Theta_l}$

$$\sum_{l\in[m]} \left[n\widehat{U}(\Theta_j,\Theta_l) + n\lambda \mathbb{I}_{\Theta_l=\Theta_j}\right]\theta_l/m = \sigma(\Theta_j^T X)Y.$$

Therefore, define $\sigma(x^T\Xi) := [\sigma(x^T\Theta_1)\ldots,\sigma(x^T\Theta_m)] \in \mathbb{R}^{1\times m}$, and $\sigma(X\Xi) := [\sigma(x_1^T\Xi)^T,\ldots,\sigma(x_n^T\Xi)^T] \in \mathbb{R}^{m\times n}$, we have

$$\begin{aligned}
\widehat{f}_{\infty}^{\mathrm{nn},\lambda}(x) &= \sum_{l\in[m]} \theta_l\sigma(x^T\Theta_l)/m = \sigma(x^T\Xi)[\sigma(X\Xi)\sigma(X\Xi)^T + n\lambda I_m]^{-1}\sigma(X\Xi)Y \\
&= \sigma(x^T\Xi)\sigma(X\Xi)[\sigma(X\Xi)^T\sigma(X\Xi) + n\lambda I_n]^{-1}Y \\
&= \widehat{H}_{\infty}^{\lambda}(x,X)\left[\widehat{H}_{\infty}^{\lambda}(X,X) + n/m\cdot\lambda I_n\right]^{-1} Y \ .
\end{aligned}$$

The last line follows as $\widehat{H}^{\lambda}(x,\tilde{x}) := \int \sigma(x^T\Theta)\sigma(\tilde{x}^T\Theta)|\widehat{\rho}_{\infty}^{\lambda}|(d\Theta) = 1/m\cdot\sigma(x^T\Xi)\sigma(\tilde{x}^T\Xi)^T$.

□

Proof of Proposition 5.3.2.

$$\begin{aligned}
\|\lim_{\lambda\to 0}\widehat{f}_{\infty}^{\mathrm{nn},\lambda} - f_*\|_{\mu}^2 &\precsim \|\widehat{f}_{\infty}^{\mathrm{rkhs}} - f_{\infty}^{\mathrm{rkhs}}\|_{\mu}^2 + \|f_{\infty}^{\mathrm{rkhs}} - f_*\|_{\mu}^2 \\
\|f_{\infty}^{\mathrm{rkhs}} - f_*\|_{\mu}^2 &= \|H_{\infty}(x,X)H_{\infty}(X,X)^{+}[Y - f_*(X) + f_*(X) - f_{\infty}(X) + f_{\infty}(X)] \\
&\quad - f_*(x)\|_{\mu}^2 \\
&\precsim \|H_{\infty}(x,X)H_{\infty}(X,X)^{+}(Y - f_*(X))\|_{\mu}^2 \\
&\quad + \mathbb{E}_{\mathbf{x}\sim\mu}\langle H_{\infty}(X,X)^{+}H_{\infty}(X,\mathbf{x}), f_*(X) - f_{\infty}(X)\rangle^2 \\
&\quad + \|H_{\infty}(x,X)H_{\infty}(X,X)^{+}f_{\infty}(X) - f_{\infty}(x)\|_{\mu}^2 \\
&\quad + \|f_{\infty}(x) - f_*(x)\|_{\mu}^2 \ .
\end{aligned}$$

For the first term, we can upper bound by $\sigma^2\mathbb{E}_{\mathbf{x}\sim\mu}\|H_{\infty}(X,X)^{-1}H_{\infty}(X,\mathbf{x})\|^2$. The second

term can be upper bounded by

$$\mathbb{E}_{\mathbf{x}\sim\mu}\|H_\infty(X,X)^{-1}H_\infty(X,\mathbf{x})\|^2\cdot n\|f_\infty(x)-f_*(x)\|_{\hat{\mu}}^2.$$

Proof is completed. □

5.7 Appendix

5.7.1 Supporting Results

Proof of Lemma 5.4.3. Let's first show that in the infinite neuron limit $m\to\infty$, $\rho_{+,t},\rho_{-,t}$ are properly defined. Therefore Eqn. (5.4.12) in the above theorem also characterize the distribution dynamics for infinite neurons NN, induced by gradient flow training. For simplicity, we assume the initialization $\rho_{+,0},\rho_{-,0}$ with bounded support. We add the superscript m, $\rho_{+,t}^m,\rho_{-,t}^m,\rho_t^m$ to (5.4.8) to indicate their dependence on m. Consider $\nabla_\Theta V$, $\nabla_\Theta U(\Theta,\tilde{\Theta})$ in (5.4.11) are bounded and uniform Lipchitz continuous as in [117, A3]. With the same proof as in [117, Theorem 3], one can show that with $m\to\infty$, the initial distribution $\rho_0^m\xrightarrow{d}\tilde{\rho}_0=\rho_{+,0}-\rho_{-,0}$ by law of large number, and by the solution's continuity depending on the initial value. Therefore we have $\rho_t^m\xrightarrow{d}\rho_t$ as $m\to\infty$ well defined.

The velocity of a particle Θ in the positive part as a rewrite of (5.4.6)-(5.4.7) is

$$\mathcal{V}(\Theta,\rho_t)=\|\Theta\|\left(\nabla_\Theta V(\Theta)+\nabla_\Theta\int U(\Theta,\tilde{\Theta})\|\tilde{\Theta}\|\rho_t(d\tilde{\Theta})\right),\tag{5.7.1}$$

resp. for the negative part and (5.4.7), we have

$$-\mathcal{V}(\Theta,\rho_t)=-\|\Theta\|\left(\nabla_\Theta V(\Theta)+\nabla_\Theta\int U(\Theta,\tilde{\Theta})\|\tilde{\Theta}\|\rho_t(d\tilde{\Theta})\right).$$

Given the velocity of particle, we have the transport equation for gradient flow,

$$\partial_t \rho_{+,t} = -\nabla_\Theta \cdot \left(\rho_{+,t} \cdot \mathcal{V}(\Theta, \rho_t)\right),$$
$$\partial_t \rho_{-,t} = -\nabla_\Theta \cdot \left(-\rho_{-,t} \cdot \mathcal{V}(\Theta, \rho_t)\right).$$

To see this, recall the definition of weak derivative $\partial_t \rho_t$: for any bounded smooth function g, $\partial_t \rho_t$ is defined in the following sense

$$d \cdot \int g \rho_t = -\int g \partial_t \rho_t \cdot dt. \tag{5.7.2}$$

We take any bounded smooth function $g(\Theta)$, given the velocity of Θ's , then we have

$$-\int g \partial_t \rho_t \cdot dt = d \cdot \int g(\Theta) \rho_{+,t}(\Theta) = \int \nabla g(\Theta) \cdot \mathcal{V}(\Theta, \rho_t) \rho_{+,t}(\Theta) \cdot dt, \tag{5.7.3}$$

and $\rho_{-,t}$ correspondingly. By the weak derivative, we get the above PDE. We use the above dynamic description as the training process for infinite neuron NN. Plug above equation into $\rho_t = \rho_{+,t} - \rho_{-,t}$ and $|\rho_t| = \rho_{+,t} + \rho_{-,t}$, we get

$$\partial_t \rho_t(\Theta) = -\nabla_\Theta \cdot \left(|\rho_t|(\Theta) \mathcal{V}(\Theta, \rho_t)\right),$$
$$\partial_t |\rho_t|(\Theta) = -\nabla_\Theta \cdot \left(\rho_t(\Theta) \mathcal{V}(\Theta, \rho_t)\right). \tag{5.7.4}$$

□

Proof of Proposition 5.4.1. It suffices to show $\theta^2_{+,i}(t) = \|\Theta_{+,i}(t)\|_2^2$ and resp. $\theta^2_{-i}(t) =$

$\|\Theta_{-,i}(t)\|_2^2$. By our path dynamics, we have

$$\frac{d\theta_{+,i}^2}{dt} = 2\theta_{+,i}\frac{d\theta_{+,i}}{dt} = -2\mathbf{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, f(\mathbf{x}))}{\partial f}\theta_{+,i}\sigma(\mathbf{x}^T\Theta_{+,i})\right], \tag{5.7.5}$$

$$\frac{d\|\Theta_{+,i}\|_2^2}{dt} = 2\Theta_{+,i}^T\frac{d\Theta_{+,i}}{dt} = -2\mathbf{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, f(\mathbf{x}))}{\partial f}\theta_{+,i}\mathbf{1}_{\mathbf{x}^T\Theta_{+,i}\geq 0}\mathbf{x}^T\Theta_{+,i}\right] = \frac{d\theta_{+,i}^2}{dt}. \tag{5.7.6}$$

Thus, by the initialization, we have $\theta_{+,i}(t) = \|\Theta_{+,i}(t)\|$, and resp. $\theta_{-,i}(t) = -\|\Theta_{-,i}(t)\|$. □

Proposition 5.7.1 (No sign change)**.** For the training process (5.1.3) for problem (5.1.1) with NN (5.1.2), once $w_j(t)$ and $u_j(t)$ hit zero at t_0, for $t > t_0$ at least there exists a solution that can be viewed as training without the j-th neuron.

Proof of Proposition 5.7.1. Using $w_j(t_0)$, $u_j(t_0)$, for $j \neq i$, as an initial value for ODE (5.1.3) without the i-th node. By assumption, we have a solution of this $2 \cdot (2m-1)$-dimensional initial value problem. Then padding the solution with $u_i \equiv 0$ and $w_i \equiv 0$, which can be a solution for ODE (5.1.3) with i-th neuron included. □

Proof of Lemma 5.4.1. First we write down the dynamic of prediction $f(\tilde{x})$ at each point $\tilde{x}$ based on Eqn. (5.1.3). For notational simplicity, let u_j, w_j be $u_j(t), w_j(t)$, and let $o_j^1(\tilde{x}) =$

$\sigma(u_j^T \tilde{x})$, and with the square loss $\ell(y, f) = \frac{1}{2}(y - f)^2$, we have

$$\begin{aligned}
\frac{df_t(\tilde{x})}{dt} &= \sum_{j=1}^{m} \left[\frac{dw_j}{dt} o_j^1(\tilde{x}) + w_j \frac{do_j^1(\tilde{x})}{dt} \right] \\
&= \sum_{j=1}^{m} \Bigg\{ \mathbf{E}_{\mathbf{z}} \left[(\mathbf{y} - f_t(\mathbf{x})) \sigma(\mathbf{x}^T u_j) \right] o_j^1(\tilde{x}) \\
&+ w_j \mathbf{1}_{\tilde{x}^T u_j \geq 0} \tilde{x}^T \mathbf{E}_{\mathbf{z}} \left[(\mathbf{y} - f_t(\mathbf{x})) w_j \mathbf{1}_{\mathbf{x}^T u_j \geq 0} \mathbf{x} \right] \Bigg\} \\
&= \sum_{j=1}^{m} \left\{ \mathbf{E}_{\mathbf{x}} \left[(f_*(\mathbf{x}) - f_t(\mathbf{x})) \left(\sigma(\tilde{x}^T u_j) \sigma(\mathbf{x}^T u_j) + w_j^2 \mathbf{1}_{\tilde{x}^T u_j \geq 0} \mathbf{1}_{\mathbf{x}^T u_j \geq 0} \tilde{x}^T \mathbf{x} \right) \right] \right\} \\
&= \mathbf{E}_{\mathbf{x}} \Bigg\{ \sum_{j=1}^{m} \left[\sigma(\tilde{x}^T u_j) \sigma(\mathbf{x}^T u_j) + w_j^2 \mathbf{1}_{\tilde{x}^T u_j \geq 0} \mathbf{1}_{\mathbf{x}^T u_j \geq 0} \tilde{x}^T \mathbf{x} \right] (f_*(\mathbf{x}) - f_t(\mathbf{x})) \Bigg\} \\
&= \mathbf{E}_{\mathbf{x}} \left[K_t(\tilde{x}, \mathbf{x}) \Delta_t(\mathbf{x}) \right].
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\frac{d\mathbf{E}_{\mathbf{x}} \left[\frac{1}{2} \Delta_t(\mathbf{x})^2 \right]}{dt} &= -\mathbf{E}_{\mathbf{x}} \left[(f_*(\mathbf{x}) - f_t(\mathbf{x})) \frac{df_t(\mathbf{x})}{dt} \right] \\
&= -\mathbf{E}_{\mathbf{x}} \left[\Delta_t(\mathbf{x}) \mathbb{E}_{\tilde{\mathbf{x}}} \left[K_t(\mathbf{x}, \tilde{\mathbf{x}}) \Delta(\tilde{\mathbf{x}}) \right] \right] \\
&= -\mathbf{E}_{\mathbf{x}, \tilde{\mathbf{x}}} \left[\Delta_t(\mathbf{x}) K_t(\mathbf{x}, \tilde{\mathbf{x}}) \Delta_t(\tilde{\mathbf{x}}) \right].
\end{aligned} \tag{5.7.7}$$

□

Proof of Corollary 5.4.1. The first equality follows from the proof in Lemma 5.4.1. Recall the property for strongly convex function

$$\ell(y_i, f_t(x_i)) - \ell(y_i, f_*(x_i)) \leq \frac{1}{2\alpha} \left[\frac{\partial \ell(y_i, f_t(x_i))}{\partial f} \right]^2 = \frac{1}{2\alpha} \Delta_t(x_i)^2. \tag{5.7.8}$$

Therefore, we have

$$-\mathbf{E}_{\mathbf{x},\tilde{\mathbf{x}}}\left[\Delta_t(\mathbf{x})K_t(\mathbf{x},\tilde{\mathbf{x}})\Delta_t(\tilde{\mathbf{x}})\right] \le -\frac{\lambda_t}{n}\sum_{i=1}^{n}\Delta_t(x_i)^2 \le -2\alpha\lambda_t\cdot\widehat{\mathbb{E}}\left[\ell(\mathbf{y},f_t(\mathbf{x}))-\ell(\mathbf{y},f_*(\mathbf{x}))\right].$$

□

Proof of Lemma 5.4.2. We know

$$\mathbb{E}_{\mathbf{u}\sim\pi}\left[\sigma(\mathbf{u}^T x)\sigma(\mathbf{u}^T\tilde{x})\right] = \mathbb{E}_{\mathbf{u}\sim\pi}\tilde{x}^T\left[\mathbf{u}\mathbf{u}^T\mathbf{1}_{\mathbf{u}^T x>0}\mathbf{1}_{\mathbf{u}^T\tilde{x}>0}\right]x \tag{5.7.9}$$

Consider the coordinate system $e_1, e_2, \dots e_d$ such that e_1, e_2 spans the space of $x, \tilde{x}$, with

$$x = e_1, \tilde{x} = \cos\theta\cdot e_1 + \sin\theta\cdot e_2, \tag{5.7.10}$$

where $\theta = \arccos(x^T\tilde{x})$. Note $\mathbf{u} = [v_1, v_2, \dots v_d]$ is still an isotropic Gaussian under this coordinate system. The constraint reads

$$\mathbf{1}_{\mathbf{u}^T x>0}\mathbf{1}_{\mathbf{u}^T\tilde{x}>0}, \tag{5.7.11}$$

$$\text{equivalent to} \quad v_1 > 0,\ v_1\cos\theta + v_2\sin\theta > 0, \tag{5.7.12}$$

and one can see that $v_2, \dots v_d$ integrate out.

Let's focus on the spherical coordinates of $v_1 = r\cos\phi, v_2 = r\sin\phi$, then $r^2 \sim \chi^2(2)$ and

$\phi \sim U[-\pi, \pi]$. W.l.o.g., we can consider the case when $\theta \in [0, \pi]$.

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{u}\sim\pi}\left[\mathbf{u}\mathbf{u}^T\mathbf{1}_{\mathbf{u}^T x>0}\mathbf{1}_{\mathbf{u}^T\tilde{x}>0}\right]x \\
&= \mathbb{E}[r^2]\left(e_1\cdot\frac{1}{2\pi}\int_{-\pi}^{\pi}\cos^2\phi\mathbf{1}_{\phi\in[\theta-\pi/2,\pi/2]}d\phi + e_2\cdot\frac{1}{2\pi}\int_{-\pi}^{\pi}\cos\phi\sin\phi\mathbf{1}_{\phi\in[\theta-\pi/2,\pi/2]}d\phi\right) \\
&\text{because the above are equivalent to } e_1\mathbb{E}[v_1^2\mathbf{1}_{\mathbf{u}^T x>0}\mathbf{1}_{\mathbf{u}^T\tilde{x}>0}] + e_2\mathbb{E}[v_1v_2\mathbf{1}_{\mathbf{u}^T x>0}\mathbf{1}_{\mathbf{u}^T\tilde{x}>0}] \\
&= 2\cdot\frac{1}{2\pi}\left[e_1\cdot\frac{\pi-\theta}{2} + (e_1\cos\theta + e_2\sin\theta)\cdot\frac{\sin\theta}{2}\right] \\
&\text{just evaluate } \int_{\theta-\pi/2}^{\pi/2}\cos^2\phi d\phi,\ \int_{\theta-\pi/2}^{\pi/2}\cos\phi\sin\phi d\phi \\
&= \frac{\pi-\theta}{2\pi}x + \frac{\sin\theta}{2\pi}\tilde{x}.
\end{aligned}
$$

Therefore, we get

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{u}\sim\pi}x^T\left[\mathbf{u}\mathbf{u}^T\mathbf{1}_{\mathbf{u}^T x>0}\mathbf{1}_{\mathbf{u}^T\tilde{x}>0}\right]\tilde{x} \\
&= \frac{\pi-\theta}{2\pi}\cos\theta + \frac{\sin\theta}{2\pi}
\end{aligned}
$$

Similarly, we have

$$
\mathbb{E}_{\mathbf{u}\sim\pi}\tilde{x}^T\left[\mathbf{1}_{\mathbf{u}^T x\geq 0}\mathbf{1}_{\mathbf{u}^T\tilde{x}\geq 0}\right]x = \frac{\pi-\theta}{2\pi}\cos\theta. \tag{5.7.13}
$$

Summing them up, we get the result. □

Proof of Corollary 5.4.2. Our proof essentially follows the same steps for (5.4.1). First, we write down the dynamic of $f_t(x)$,

$$
\frac{df_t(x)}{dt} = \frac{\int\|\Theta\|\sigma(x^T\Theta)\rho_t(d\Theta)}{dt}. \tag{5.7.14}
$$

Plug-in the training dynamic (5.7.4), we get

$$\begin{aligned}
\frac{df_t(x)}{dt} &= -\int -\nabla_\Theta \left[\|\Theta\|\sigma(x^T\Theta)\right]\cdot \mathcal{V}(\Theta,\rho_t)|\rho_t|(d\Theta) \\
&= \int \nabla_\Theta \left[\|\Theta\|\sigma(x^T\Theta)\right]\cdot\|\Theta\| \\
&\left\{\mathbf{E}_{\tilde{\mathbf{x}}}[f_*(\tilde{\mathbf{x}})\mathbf{1}_{\tilde{\mathbf{x}}^T\Theta\geq 0}\tilde{\mathbf{x}}] - \mathbf{E}_{\tilde{\mathbf{x}}}\left[\int\left(\|\tilde{\Theta}\|\sigma(\tilde{\mathbf{x}}^T\tilde{\Theta})\mathbf{1}_{\tilde{\mathbf{x}}^T\Theta\geq 0}\tilde{\mathbf{x}}\right)\rho_t(d\tilde{\Theta})\right]\right\}|\rho_t|(d\Theta) \\
&= \mathbf{E}_{\tilde{\mathbf{x}}}\left\{\int \nabla_\Theta\left[\|\Theta\|\sigma(x^T\Theta)\right]\cdot\|\Theta\|\left[\Delta_t(\tilde{\mathbf{x}})\mathbf{1}_{\tilde{\mathbf{x}}^T\Theta\geq 0}\tilde{\mathbf{x}}\right]|\rho_t|(d\Theta)\right\} \\
&= \mathbf{E}_{\tilde{\mathbf{x}}}\left\{\Delta_t(\tilde{\mathbf{x}})\cdot\int\|\Theta\|^2\mathbf{1}_{x^T\Theta\geq 0}\mathbf{1}_{\tilde{\mathbf{x}}^T\Theta\geq 0}x^T\tilde{\mathbf{x}}+\sigma(x^T\Theta)\sigma(\tilde{\mathbf{x}}^T\Theta)|\rho_t|(d\Theta)\right\}.
\end{aligned}$$

Therefore, we have

$$\frac{d\mathbf{E}_{\mathbf{x}}\left[\frac{1}{2}\Delta_t(\mathbf{x})^2\right]}{dt} = -\mathbf{E}_{\mathbf{x},\tilde{\mathbf{x}}}[\Delta_t(\mathbf{x})K_t(\mathbf{x},\tilde{\mathbf{x}})\Delta_t(\tilde{\mathbf{x}})]. \tag{5.7.15}$$

□

5.7.2 Extensions

In this section, we extend the definition of the dynamic kernel in Section 5.4 to the multi-layer neural networks case. We construct a recursive expression for the kernel defined by the multi-layer perceptron (MLP). Let $\Theta_{i,j}^l$, $l = 0,\cdots,h-1$ denote the coefficient from the i-th node on the l-th layer to the j-th node on the $(l+1)$-th layer. Let the input (before activation) of the i-th node on l-th layer be $v_i^l(x) = \sum_j \Theta_{j,i}^{l-1}o_j^{l-1}(x)$ and let the output at that node be $o_i^l = \sigma(v_i^l)$, for $l \notin \{0,h\}$, and $o_i^l = x_i$, for $l = 0$. The final output $g(x) = (v_1^h(x), v_2^h(x),\cdots,v_{L_h}^h(x))^T$. Let $L_0 = d$ and L_i is the number of nodes at the i-th layer. Denote $K_t^h(x,\tilde{x};\{\Theta^l\}_{l=0,\ldots,h})$ the kernel of h layers NN. The training dynamic is still

the gradient flow, for all Θ

$$\frac{d\Theta}{dt} = -\mathbb{E}_{\mathbf{z}}\left[\frac{\partial \ell(\mathbf{y}, g(\mathbf{x}))}{\partial g}\frac{\partial g(\mathbf{x})}{\partial \Theta}\right].$$

Proposition 5.7.2. For a $(h+1)$-layer NN function denoted by $g(x)$, for simplicity, let

$$K_t^{h+1}(x, \tilde{x}) = K_t^{h+1}(x, \tilde{x}; \{\Theta^l\}_{l=0,\ldots,h+1}), \tag{5.7.16}$$

$$K_t^{h}(z, \tilde{z}) = K_t^{h}(z, \tilde{z}; \{\Theta^l\}_{l=1,\ldots,h+1}). \tag{5.7.17}$$

With gradient flow training process, we have the following recursive representation of the corresponding kernel matrix

$$K_t^{h+1}(x, \tilde{x}) = K_t^{h}(o^1(x), o^1(\tilde{x})) + \sum_{i=1,j=1}^{L_0,L_1} \frac{\partial g(x)}{\partial \Theta_{i,j}^0}\frac{\partial g(\tilde{x})}{\partial \Theta_{i,j}^0}.$$

Here the kernel matrix is always positive semidefinite.

Proof of Proposition 5.7.2. For notational simplicity, let
$K_t^{h+1}(x, \tilde{x}) = K_t^{h+1}(x, \tilde{x}; \{\Theta^l\}_{l=0,\ldots,h+1})$, and

$$K_t^{h}(z, \tilde{z}) = K_t^{h}(z, \tilde{z}; \{\Theta^l\}_{l=1,\ldots,h+1}).$$

For the proof, we calculate the dynamic of prediction $g(x)$, by elementary calculus, we have

$$\frac{dg(x)}{dt} = -\mathbf{E}_{\mathbf{x}}[f_*(\mathbf{x}) - g(\mathbf{x})]\Big[\sum_{\text{all } \Theta}\frac{\partial g(x)}{\partial \Theta}\cdot\frac{\partial g(\mathbf{x})}{\partial \Theta}\Big]. \tag{5.7.18}$$

With same calculation for the dynamic of Δ_t as in (5.7.7), we get

$$K_t^{h+1}(x, x') = \sum_{\Theta\in\Theta^0}\frac{\partial g(x)}{\partial \Theta}\cdot\frac{\partial g(\tilde{x})}{\partial \Theta} + \sum_{\text{other } \Theta}\frac{\partial g(x)}{\partial \Theta}\cdot\frac{\partial g(\tilde{x})}{\partial \Theta}. \tag{5.7.19}$$

By induction, we get

$$K_t^{h+1}(x,\tilde{x}) = K_t^h(o^1(x), o^1(\tilde{x})) + \sum_{i=1,j=1}^{L_0,L_1} \frac{\partial g(x)}{\partial \Theta_{i,j}^0} \frac{\partial g(\tilde{x})}{\partial \Theta_{i,j}^0}. \tag{5.7.20}$$

Now, we prove the positive semi-definiteness of the kernel. By induction, we only need to prove that the second term above is non-negative. We construct a canonical mapping $\phi_{h+1}(x) := v(x), \mathbb{R}^d \to \mathbb{R}^{L_0 \times L_1}$, whereas the i,j-th coordinate $v(x)_{i,j} = \frac{\partial g(x)}{\partial \Theta_{i,j}^0}$. Then the second term can be seen as a inner product $\langle \phi_{h+1}(x), \phi_{h+1}(\tilde{x}) \rangle$, which implies the non-negativity. □

REFERENCES

[1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24:2312–2320, 2011.

[2] Charu C Aggarwal. An introduction to outlier analysis. In *Outlier analysis*, pages 1–34. Springer, 2017.

[3] Dan Alistarh, Zeyuan Allen-Zhu, and Jerry Li. Byzantine stochastic gradient descent. *arXiv preprint arXiv:1803.08917*, 2018.

[4] Martin Anthony and Peter L Bartlett. *Neural network learning: Theoretical foundations*. cambridge university press, 2009.

[5] Karl J Åström. *Introduction to stochastic control theory*. Courier Corporation, 2012.

[6] Francis Bach. Breaking the curse of dimensionality with convex neural networks. *Journal of Machine Learning Research*, 18(19):1–53, 2017.

[7] Andrew R Barron, Albert Cohen, Wolfgang Dahmen, Ronald A DeVore, et al. Approximation and learning by greedy algorithms. *The annals of statistics*, 36(1):64–94, 2008.

[8] Sumanta Basu, George Michailidis, et al. Regularized estimation in sparse high-dimensional time series models. *The Annals of Statistics*, 43(4):1535–1567, 2015.

[9] Güzin Bayraksan and David K Love. Data-driven stochastic programming using phi-divergences. In *The Operations Research Revolution*, pages 1–19. INFORMS, 2015.

[10] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[11] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[12] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine learning and the bias-variance trade-off. *arXiv preprint arXiv:1812.11118*, 2018.

[13] Mikhail Belkin, Siyuan Ma, and Soumik Mandal. To understand deep learning we need to understand kernel learning. *arXiv preprint arXiv:1802.01396*, 2018.

[14] Aharon Ben-Tal and Arkadi Nemirovski. Robust convex optimization. *Mathematics of operations research*, 23(4):769–805, 1998.

[15] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization*. Princeton university press, 2009.

[16] Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

[17] Yoshua Bengio, Nicolas L Roux, Pascal Vincent, Olivier Delalleau, and Patrice Marcotte. Convex neural networks. In *Advances in neural information processing systems*, pages 123–130, 2006.

[18] Dimitris Bertsimas, David B Brown, and Constantine Caramanis. Theory and applications of robust optimization. *SIAM review*, 53(3):464–501, 2011.

[19] Kush Bhatia, Prateek Jain, and Purushottam Kar. Robust regression via hard thresholding. *arXiv preprint arXiv:1506.02428*, 2015.

[20] Kush Bhatia, Prateek Jain, Parameswaran Kamalaruban, and Purushottam Kar. Efficient and consistent robust time series analysis. *arXiv preprint arXiv:1607.00146*, 2016.

[21] Kush Bhatia, Prateek Jain, Parameswaran Kamalaruban, and Purushottam Kar. Consistent robust regression. In *Advances in Neural Information Processing Systems*, pages 2110–2119, 2017.

[22] John R Birge and Francois Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.

[23] Thomas Blumensath and Mike E Davies. Iterative hard thresholding for compressed sensing. *Applied and computational harmonic analysis*, 27(3):265–274, 2009.

[24] François Bolley, Arnaud Guillin, and Cédric Villani. Quantitative concentration inequalities for empirical measures on non-compact spaces. *Probability Theory and Related Fields*, 137(3-4):541–593, 2007.

[25] Denis Bosq. *Nonparametric statistics for stochastic processes: estimation and prediction*, volume 110. Springer Science & Business Media, 2012.

[26] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[27] Peter J Brockwell and Richard A Davis. *Introduction to time series and forecasting*. springer, 2016.

[28] Sébastien Bubeck. Convex optimization: Algorithms and complexity. *arXiv preprint arXiv:1405.4980*, 2014.

[29] Giuseppe Carlo Calafiore and Laurent El Ghaoui. On distributionally robust chance-constrained linear programs. *Journal of Optimization Theory and Applications*, 130 (1):1–22, 2006.

[30] Eduardo F Camacho and Carlos Bordons Alba. *Model predictive control*. Springer science & business media, 2013.

[31] George Casella and Roger L Berger. *Statistical inference*. Cengage Learning, 2021.

[32] Bill Casselman. Essays in analysis. 2014.

[33] Mengjie Chen, Chao Gao, and Zhao Ren. A general decision theory for huber's ϵ-contamination model. *Electronic Journal of Statistics*, 10(2):3752–3774, 2016.

[34] Yudong Chen, Lili Su, and Jiaming Xu. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *arXiv preprint arXiv:1705.05491*, 2017.

[35] Xiuyuan Cheng and Amit Singer. The spectrum of random inner-product kernel matrices. *Random Matrices: Theory and Applications*, 2(04):1350010, 2013.

[36] Lenaic Chizat and Francis Bach. A note on lazy training in supervised differentiable programming. *arXiv preprint arXiv:1812.07956*, 2018.

[37] Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. *arXiv preprint arXiv:1812.07956*, 2018.

[38] Youngmin Cho and Lawrence K Saul. Kernel methods for deep learning. In *Advances in neural information processing systems*, pages 342–350, 2009.

[39] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

[40] Amit Daniely, Roy Frostig, and Yoram Singer. Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity. In *Advances In Neural Information Processing Systems*, pages 2253–2261, 2016.

[41] Erick Delage and Yinyu Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations research*, 58(3):595–612, 2010.

[42] Olivier Devolder, François Glineur, and Yurii Nesterov. First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146(1):37–75, 2014.

[43] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. *arXiv preprint arXiv:1803.02815*, 2018.

[44] Hacene Djellout, Arnaud Guillin, Liming Wu, et al. Transportation cost-information inequalities and applications to random dynamical systems and diffusions. *The Annals of Probability*, 32(3B):2702–2732, 2004.

[45] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52 (4):1289–1306, 2006.

[46] David L Donoho and Peter J Huber. The notion of breakdown point. *A festschrift for Erich L. Lehmann*, 157184, 1983.

[47] Xialiang Dou and Mihai Anitescu. Distributionally robust optimization with correlated data from vector autoregressive processes. *Operations Research Letters*, 47(4):294–299, 2019.

[48] Xialiang Dou and Tengyuan Liang. Training neural networks as learning data-adaptive kernels: Provable representation and approximation benefits. *Journal of the American Statistical Association*, pages 1–14, 2020.

[49] Xialiang Dou and Tengyuan Liang. Training neural networks as learning data-adaptive kernels: Provable representation and approximation benefits. *Journal of the American Statistical Association*, 116(535):1507–1520, 2021.

[50] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR, 2019.

[51] Simon S Du, Wei Hu, and Jason D Lee. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. *arXiv preprint arXiv:1806.00900*, 2018.

[52] Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*, 2018.

[53] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

[54] John C Duchi, Alekh Agarwal, and Martin J Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic control*, 57(3):592–606, 2011.

[55] Rick Durrett. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

[56] Noureddine El Karoui et al. The spectrum of kernel random matrices. *The Annals of Statistics*, 38(1):1–50, 2010.

[57] E Erdoğan and Garud Iyengar. Ambiguous chance constrained problems and robust optimization. *Mathematical Programming*, 107(1-2):37–61, 2006.

[58] Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, pages 1–52, 2015.

[59] Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1):115–166, 2018.

[60] Max H Farrell, Tengyuan Liang, and Sanjog Misra. Deep neural networks for estimation and inference: Application to causal effects and other semiparametric estimands. *arXiv preprint arXiv:1809.09953*, 2018.

[61] Nicolas Fournier and Arnaud Guillin. On the rate of convergence in wasserstein distance of the empirical measure. *Probability Theory and Related Fields*, 162(3-4):707–738, 2015.

[62] John Fox. *Applied regression analysis, linear models, and related methods.* Sage Publications, Inc, 1997.

[63] Marguerite Frank, Philip Wolfe, et al. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

[64] Karl Friston. Causal modelling and brain connectivity in functional magnetic resonance imaging. *PLoS biology*, 7(2):e1000033, 2009.

[65] J-J Fuchs. An inverse problem approach to robust regression. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, volume 4, pages 1809–1812. IEEE, 1999.

[66] Rui Gao and Anton J Kleywegt. Distributionally robust stochastic optimization with wasserstein distance. *arXiv preprint arXiv:1604.02199*, 2016.

[67] Rui Gao and Anton J Kleywegt. Distributionally robust stochastic optimization with dependence structure. *arXiv preprint arXiv:1701.04200*, 2017.

[68] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on learning theory*, pages 797–842. PMLR, 2015.

[69] Rong Ge, Jason D Lee, and Tengyu Ma. Matrix completion has no spurious local minimum. *arXiv preprint arXiv:1605.07272*, 2016.

[70] Stuart Geman and Chii-Ruey Hwang. Nonparametric maximum likelihood estimation by the method of sieves. *The Annals of Statistics*, pages 401–414, 1982.

[71] Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[72] Saeed Ghadimi and Guanghui Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156(1-2):59–99, 2016.

[73] Saeed Ghadimi, Guanghui Lan, and Hongchao Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1-2):267–305, 2016.

[74] Behrooz Ghorbani, Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Linearized two-layers neural networks in high dimension. *arXiv preprint arXiv:1904.12191*, 2019.

[75] A Giloni and M Padberg. Least trimmed squares regression, least median squares regression, and mathematical programming. *Mathematical and Computer Modelling*, 35(9-10):1043–1060, 2002.

[76] Clark R Givens and Rae Michael Shortt. A class of wasserstein metrics for probability distributions. *Michigan Mathematical Journal*, 31(2):231–240, 1984.

[77] Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[78] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[79] Shixiang Gu and Luca Rigazio. Towards deep neural network architectures robust to adversarial examples. *arXiv preprint arXiv:1412.5068*, 2014.

[80] James Douglas Hamilton. *Time series analysis*. Princeton university press, 2020.

[81] Bruce E Hansen. Uniform convergence rates for kernel estimation with dependent data. *Econometric Theory*, 24(3):726–748, 2008.

[82] Wolfgang Härdle, Alexandre Tsybakov, and Lijian Yang. Nonparametric vector autoregression. *Journal of Statistical Planning and Inference*, 68(2):221–245, 1998.

[83] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *arXiv preprint arXiv:1903.08560*, 2019.

[84] Douglas M Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

[85] Elad Hazan and Satyen Kale. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1):2489–2512, 2014.

[86] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. In *The collected works of Wassily Hoeffding*, pages 409–426. Springer, 1994.

[87] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[88] Cong Huang, Gerald HL Cheang, and Andrew R Barron. *Risk of penalized least squares, greedy selection and l1-penalization for flexible function libraries*. PhD thesis, Yale University, 2008.

[89] Peter J Huber. A robust version of the probability ratio test. *The Annals of Mathematical Statistics*, pages 1753–1758, 1965.

[90] Peter J Huber. Robust regression: asymptotics, conjectures and monte carlo. *The annals of statistics*, pages 799–821, 1973.

[91] Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pages 492–518. Springer, 1992.

[92] Peter J Huber. *Robust statistics*, volume 523. John Wiley & Sons, 2004.

[93] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pages 8571–8580, 2018.

[94] Martin Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, pages 427–435. PMLR, 2013.

[95] Ruiwei Jiang and Yongpei Guan. Data-driven chance constrained stochastic program. *Mathematical Programming*, 158(1-2):291–327, 2016.

[96] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pages 1724–1732. PMLR, 2017.

[97] Chi Jin, Praneeth Netrapalli, Rong Ge, Sham M Kakade, and Michael I Jordan. A short note on concentration inequalities for random vectors with subgaussian norm. *arXiv preprint arXiv:1902.03736*, 2019.

[98] Lee K Jones. A simple lemma on greedy approximation in hilbert space and convergence rates for projection pursuit regression and neural network training. *The annals of Statistics*, 20(1):608–613, 1992.

[99] Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker–planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

[100] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

[101] Michael J Kearns, Umesh Virkumar Vazirani, and Umesh Vazirani. *An introduction to computational learning theory*. MIT press, 1994.

[102] Diego Klabjan, David Simchi-Levi, and Miao Song. Robust stochastic lot-sizing by means of histograms. *Production and Operations Management*, 22(3):691–710, 2013.

[103] Frederic Koehler and Andrej Risteski. Representational power of relu networks and polynomial kernels: Beyond worst-case analysis. *arXiv preprint arXiv:1805.11405*, 2018.

[104] Galina M Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.

[105] Panqanamala Ramana Kumar and Pravin Varaiya. *Stochastic systems: Estimation, identification, and adaptive control*, volume 75. SIAM, 2015.

[106] Vitaly Kuznetsov and Mehryar Mohri. Generalization bounds for non-stationary mixing processes. *Machine Learning*, 106(1):93–117, 2017.

[107] Tze Leung Lai and Ching Zong Wei. Least squares estimates in stochastic regression models with applications to identification and control of dynamic systems. *The Annals of Statistics*, pages 154–166, 1982.

[108] Yoonkyung Lee, Steven N MacEachern, and Yoonsuh Jung. Regularization of case-specific parameters for robustness and efficiency. *Statistical Science*, 27(3):350–372, 2012.

[109] Shuo Li, Xialiang Dou, Ruiqi Gao, Xinzhou Ge, Minping Qian, and Lin Wan. A remark on copy number variation detection methods. *PLoS One*, 13(4):e0196226, 2018.

[110] Tengyuan Liang and Alexander Rakhlin. Just interpolate: Kernel" ridgeless" regression can generalize. *arXiv preprint arXiv:1808.00387*, 2018.

[111] Russell Lyons and Yuval Peres. *Probability on trees and networks*, volume 42. Cambridge University Press, 2017.

[112] Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of sgd in modern over-parametrized learning. *arXiv preprint arXiv:1712.06559*, 2017.

[113] Hartmut Maennel, Olivier Bousquet, and Sylvain Gelly. Gradient descent quantizes relu network features. *arXiv preprint arXiv:1803.08367*, 2018.

[114] Colin L Mallows. On some topics in robustness. *Unpublished memorandum, Bell Telephone Laboratories, Murray Hill, NJ*, 37, 1975.

[115] R Douglas Martin. Robust methods for time series. In *Applied time series analysis II*, pages 683–759. Elsevier, 1981.

[116] Pascal Massart. Concentration inequalities and model selection. 2007.

[117] Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layers neural networks. *arXiv preprint arXiv:1804.06561*, 2018.

[118] El Mahdi El Mhamdi, Rachid Guerraoui, and Sébastien Rouault. The hidden vulnerability of distributed learning in byzantium. *arXiv preprint arXiv:1802.07927*, 2018.

[119] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[120] Elizbar A Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142, 1964.

[121] Vaishnavh Nagarajan and J Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. *arXiv preprint arXiv:1902.04742*, 2019.

[122] Arkadi Nemirovski. Information-based complexity of convex programming. *Lecture Notes*, 1995.

[123] Arkadi Nemirovski. Prox-method with rate of convergence o (1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004.

[124] Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

[125] Arkadij Semenovič Nemirovskij and David Borisovich Yudin. Problem complexity and method efficiency in optimization. 1983.

[126] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

[127] Yurii E Nesterov. A method for solving the convex programming problem with convergence rate o (1/kˆ 2). In *Dokl. akad. nauk Sssr*, volume 269, pages 543–547, 1983.

[128] Partha Niyogi and Federico Girosi. On the relationship between generalization error, hypothesis complexity, and sample complexity for radial basis functions. *Neural Computation*, 8(4):819–842, 1996.

[129] Victor M Panaretos and Yoav Zemel. Statistical aspects of wasserstein distances. *Annual review of statistics and its application*, 6:405–431, 2019.

[130] Victor M Panaretos and Yoav Zemel. *An invitation to statistics in Wasserstein space*. Springer Nature, 2020.

[131] Jooyoung Park and Irwin W Sandberg. Universal approximation using radial-basis-function networks. *Neural computation*, 3(2):246–257, 1991.

[132] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6): 355–607, 2019.

[133] Iosif Pinelis. Optimum bounds for the distributions of martingales in banach spaces. *The Annals of Probability*, pages 1679–1706, 1994.

[134] Tomaso Poggio, Hrushikesh Mhaskar, Lorenzo Rosasco, Brando Miranda, and Qianli Liao. Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review. *International Journal of Automation and Computing*, 14(5):503–519, 2017.

[135] Ioana Popescu. Robust mean-covariance solutions for stochastic optimization. *Operations Research*, 55(1):98–112, 2007.

[136] Adarsh Prasad, Arun Sai Suggala, Sivaraman Balakrishnan, and Pradeep Ravikumar. Robust estimation via robust gradient estimation. *arXiv preprint arXiv:1802.06485*, 2018.

[137] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pages 1177–1184, 2008.

[138] Ali Rahimi and Benjamin Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in neural information processing systems*, pages 1313–1320, 2009.

[139] Alexander Rakhlin and Xiyu Zhai. Consistency of interpolation with laplace kernels is a high-dimensional phenomenon. *arXiv preprint arXiv:1812.11167*, 2018.

[140] Alexander Rakhlin, Ohad Shamir, Karthik Sridharan, et al. Making gradient descent optimal for strongly convex stochastic optimization. In *ICML*. Citeseer, 2012.

[141] Sidney I Resnick. *Heavy-tail phenomena: probabilistic and statistical modeling*. Springer Science & Business Media, 2007.

[142] Philippe Rigollet. Lecture notes for high dimensional statistics. 2018. Available online at http://www-math.mit.edu/ rigollet/PDFs/RigNotes17.pdf.

[143] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[144] Grant M Rotskoff and Eric Vanden-Eijnden. Neural networks as interacting particle systems: Asymptotic convexity of the loss landscape and universal scaling of the approximation error. *arXiv preprint arXiv:1805.00915*, 2018.

[145] Peter Rousseeuw and Victor Yohai. Robust regression by means of s-estimators. In *Robust and nonlinear time series analysis*, pages 256–272. Springer, 1984.

[146] Peter J Rousseeuw. Least median of squares regression. *Journal of the American statistical association*, 79(388):871–880, 1984.

[147] Peter J Rousseeuw and Mia Hubert. Robust statistics for outlier detection. *Wiley interdisciplinary reviews: Data mining and knowledge discovery*, 1(1):73–79, 2011.

[148] Alessandro Rudi and Lorenzo Rosasco. Generalization properties of learning with random features. In *Advances in Neural Information Processing Systems*, pages 3215–3225, 2017.

[149] David Ruppert. *Statistics and data analysis for financial engineering*, volume 13. Springer, 2011.

[150] Dino Sejdinovic, Bharath Sriperumbudur, Arthur Gretton, and Kenji Fukumizu. Equivalence of distance-based and rkhs-based statistics in hypothesis testing. *The Annals of Statistics*, pages 2263–2291, 2013.

[151] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[152] Ohad Shamir. A variant of azuma's inequality for martingales with subgaussian tails. *arXiv preprint arXiv:1110.2392*, 2011.

[153] Ohad Shamir and Tong Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *International Conference on Machine Learning*, pages 71–79, 2013.

[154] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948.

[155] Alexander Shapiro. On duality theory of conic linear problems. In *Semi-infinite programming*, pages 135–165. Springer, 2001.

[156] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on stochastic programming: modeling and theory*. SIAM, 2009.

[157] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczynski. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

[158] Yiyuan She and Art B Owen. Outlier detection using nonconvex penalized regression. *Journal of the American Statistical Association*, 106(494):626–639, 2011.

[159] Andrew F Siegel. Robust regression using repeated medians. *Biometrika*, 69(1):242–244, 1982.

[160] Christopher A Sims. Macroeconomics and reality. *Econometrica: Journal of the Econometric Society*, pages 1–48, 1980.

[161] Maurice Sion et al. On general minimax theorems. *Pacific Journal of mathematics*, 8 (1):171–176, 1958.

[162] Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks: A central limit theorem. *Stochastic Processes and their Applications*, 2019.

[163] Mei Song, Andrea Montanari, and P Nguyen. A mean field view of the landscape of two-layers neural networks. *Proceedings of the National Academy of Sciences*, 115: E7665–E7671, 2018.

[164] Matthew Staib and Stefanie Jegelka. Distributionally robust optimization and generalization in kernel methods. *Advances in Neural Information Processing Systems*, 32: 9134–9144, 2019.

[165] Charles J Stone. Optimal rates of convergence for nonparametric estimators. *The annals of Statistics*, pages 1348–1360, 1980.

[166] Lili Su and Jiaming Xu. Securing distributed machine learning in high dimensions. *arXiv preprint arXiv:1804.10140*, 2018.

[167] Hailin Sun and Huifu Xu. Convergence analysis for distributionally robust optimization and equilibrium problems. *Mathematics of Operations Research*, 41(2):377–401, 2015.

[168] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[169] Matus Telgarsky. Neural networks and rational functions. *arXiv preprint arXiv:1706.03301*, 2017.

[170] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[171] Young K Truong and Charles J Stone. Nonparametric function estimation involving time series. *The Annals of Statistics*, pages 77–97, 1992.

[172] Efthymios Tsakonas, Joakim Jaldén, Nicholas D Sidiropoulos, and Björn Ottersten. Convergence of the huber regression m-estimate in the presence of dense outliers. *IEEE Signal Processing Letters*, 21(10):1211–1214, 2014.

[173] Ruey S Tsay. *Analysis of financial time series*, volume 543. John wiley & sons, 2005.

[174] Ruey S Tsay. *Multivariate time series analysis: with R and financial applications*. John Wiley & Sons, 2013.

[175] Alexandre B Tsybakov. Introduction to nonparametric estimation., 2009.

[176] Stephen Tu and Benjamin Recht. Least-squares temporal difference learning for the linear quadratic regulator. *arXiv preprint arXiv:1712.08642*, 2017.

[177] Aad W Van Der Vaart, Adrianus Willem van der Vaart, Aad van der Vaart, and Jon Wellner. *Weak convergence and empirical processes: with applications to statistics*. Springer Science & Business Media, 1996.

[178] Tim van Erven, Sarah Sachs, Wouter M Koolen, and Wojciech Kotlowski. Robust online convex optimization in the presence of outliers. In *Conference on Learning Theory*, pages 4174–4194. PMLR, 2021.

[179] Vladimir Vapnik. *Statistical learning theory. 1998*, volume 3. Wiley, New York, 1998.

[180] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999.

[181] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. 2010. URL `https://arxiv.org/abs/1011.3027`.

[182] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

[183] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[184] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

[185] Grace Wahba. *Spline models for observational data*, volume 59. Siam, 1990.

[186] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

[187] Zizhuo Wang, Peter W Glynn, and Yinyu Ye. Likelihood robust optimization for data-driven problems. *Computational Management Science*, 13(2):241–261, 2016.

[188] Geoffrey S Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 359–372, 1964.

[189] Li Wenliang, Dougal Sutherland, Heiko Strathmann, and Arthur Gretton. Learning deep kernels for exponential family densities. *arXiv preprint arXiv:1811.08357*, 2018.

[190] Wolfram Wiesemann, Daniel Kuhn, and Melvyn Sim. Distributionally robust convex optimization. *Operations Research*, 62(6):1358–1376, 2014.

[191] Wayne L Winston and Jeffrey B Goldberg. *Operations research: applications and algorithms*, volume 3. Thomson Brooks/Cole Belmont, 2004.

[192] Lin Xiao. Dual averaging methods for regularized stochastic learning and online optimization. 2010.

[193] Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. Generalized byzantine-tolerant sgd. *arXiv preprint arXiv:1802.10116*, 2018.

[194] Dong Yin, Yudong Chen, Kannan Ramchandran, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. *arXiv preprint arXiv:1803.01498*, 2018.

[195] Xue Ying. An overview of overfitting and its solutions. In *Journal of Physics: Conference Series*, volume 1168, page 022022. IOP Publishing, 2019.

[196] Victor J Yohai. High breakdown-point and high efficiency robust estimates for regression. *The Annals of statistics*, pages 642–656, 1987.

[197] Bin Yu. Rates of convergence for empirical processes of stationary mixing sequences. *The Annals of Probability*, pages 94–116, 1994.

[198] Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 30(9):2805–2824, 2019.

[199] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

[200] Kemin Zhou, John Comstock Doyle, Keith Glover, et al. *Robust and optimal control*, volume 40. Prentice hall New Jersey, 1996.

[201] Jia-Jie Zhu, Wittawat Jitkrittum, Moritz Diehl, and Bernhard Schölkopf. Kernel distributionally robust optimization. *arXiv preprint arXiv:2006.06981*, 2020.

[202] Georgios Zioutas and Antonios Avramidis. Deleting outliers in robust regression with mixed integer programming. *Acta Mathematicae Applicatae Sinica*, 21(2):323–334, 2005.

[203] Steve Zymler, Daniel Kuhn, and Berç Rustem. Distributionally robust joint chance constraints with second-order moment information. *Mathematical Programming*, 137 (1-2):167–198, 2013.