Subquantile Minimization: A Robust Meta-Algorithm

Arvind Rathnashyam, Fatih Orhan, Joshua Myers, & Jake Herman *
Department of Computer Science
Rensselaer Polytechnic University
Troy, NY 12180, USA
{rathna, orhanf, myersj5, hermaj2}@rpi.edu

April 30, 2023

Abstract

Robust Linear Regression is the problem of fitting data to a distribution, P when there exists contaminated samples, Q. We consider the Huber Contamination modeled as $\hat{P} = (1-\varepsilon)P + \varepsilon Q$ where $\varepsilon \in (0, 0.5)$. Traditional Least Squares Methods fit the empirical risk model to all training data in $\hat{P}$. In this paper we show theoretical and experimental results of Subquantile optimization to extract the target distribution, P from $\hat{P}$, where we optimize with respect to the p-quantile of the empirical loss. Our algorithm produces state of the art results in various baselines and is theoretically proven to converge.

1 Introduction

Linear Regression is one of the most widely used statistical estimators throughout science. Robustness Learning in High Dimensions on Huber Contamination Models, Huber & Ronchetti (2009), has gained much attention in the last decade, Diakonikolas & Kane (2019). The key motivating factor in investigating robust linear regression is the sheer vastness of probability distributions that are not drawn from a normal distribution schema. Given that outliers in data sets occur so frequent, the ability for a linear regression model to be robust is necessary to compensate for the various distributions being analyzed.

1.1 Motivations

The failure of classical regression techniques being unable to model data highly corrupted by outliers can be conveyed clearly in numerous datasets, including those featuring data in the medical, economic, and meteorological fields. Ultimately, in many real data sets, the samples may not be collected from even or fair distributions; thus, classical analyses such as standard regression or least-squares may not represent the actual distribution of the data well.

The quantile is a statistical measure that is distribution-agnostic, this makes it very suitable for robust estimation in the Huber Contamination Model.

1.2 Contributions

In this paper we provide a theoretical analysis of Subquantile minimization for various learning tasks and show state of the art results in various baselines.

Subquantile Optimization aims to address the shortcomings of ERM in applications such as noisy/corrupted data (Khetan et al. (2018),Jiang et al. (2018)), classification with imbalanced classes, (Lin et al. (2017),He & Garcia (2009)), as well as fair learning (Corbett-Davies & Goel (2018)).

As seen in the above comparison, current models fail to estimate data sets corrupted by structured noise, with some models even failing to estimate trends plagued with unstructured noise. Through this, Subquantile optimization is shown to prevail at overcoming these challenges current models currently face. In Table

*Work done as a part of ML and Optimization Spring 2023 Group Project.

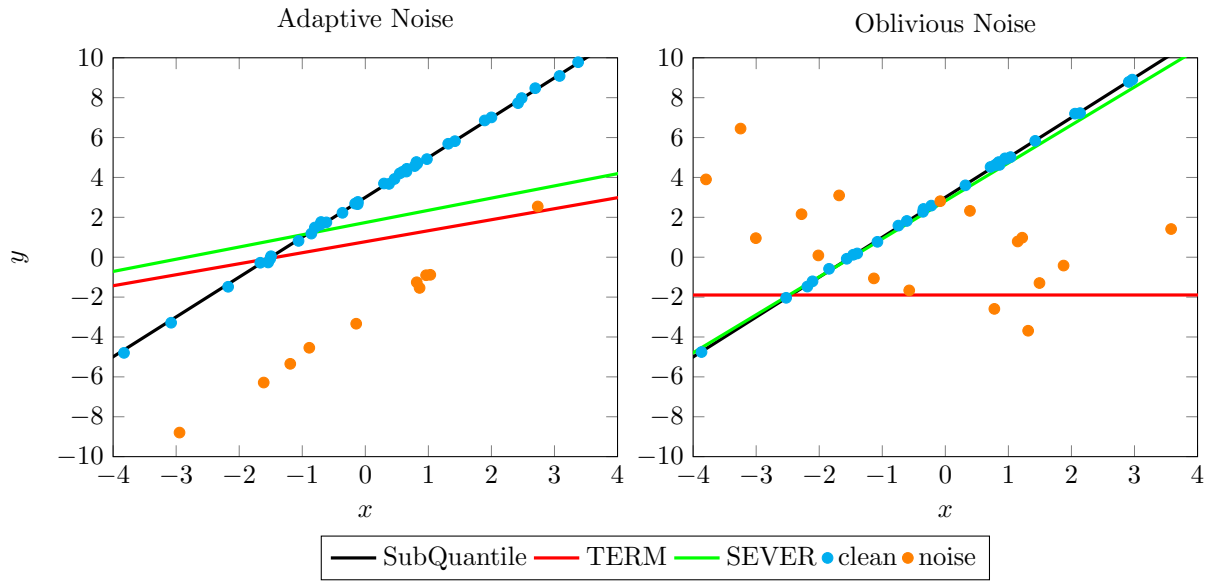

Figure 1: Oblivious Outliers are generated without knowledge of the clean distribution. Adaptive outliers are generated with knowledge of the clean distribution.

Paper	Adversary	Threshold	Resampling	Iteration Complexity
SEVER Diakonikolas et al. (2019)	Adaptive	Gradient of Loss	✗	$\mathcal{O}(nd^2)$
TMLE Awasthi et al. (2022)	Adaptive	Likelihood	✗	$\mathcal{O}(n)$
Subquantile Minimization (Ours)	Adaptive	Loss	✓	$\mathcal{O}(n)$

Table 1: Algorithms which run a base learner in each iteration. The Iteration Complexity is time complexity of the thresholding after the base learner is run.

2 Related Work

Here we will discuss iterative thresholding algorithm for specific robust learning tasks such as linear regression and classification. We will also discuss varioius robust-meta algorithms which can .

Diakonikolas et al. (2019) proposed SEVER, a gradient filtering algorithm which removes elements whose points whose distance from the centered gradient projected on to the top right singular vector of the gradients of losses is greatest, described in equation 1.

$$\tau_i = \left((\nabla f_i(\mathbf{w}) - \hat{\nabla}) \cdot \mathbf{v}\right)^2 \tag{1}$$

SEVER does not resample points removed in earlier iterations and requires a base learner in each iteration.

Li et al. (2020) propose Tilted Empirical Risk Minimization (TERM), a framework that handles the shortcomings of empirical risk minimization (ERM) with respect to robustness. TERM minimizes the objective function in 2 where t is the tilt hyperparameter.

$$\tilde{R}(t;\boldsymbol{\theta}) \coloneqq \frac{1}{t}\log\left(\frac{1}{N}\sum_{i\in[N]} e^{tf(\mathbf{x}_i;\boldsymbol{\theta})}\right) \tag{2}$$

The tilt hyperparameter changes the individual impact of each loss to make the model more resistant to outliers found in the data.

Awasthi et al. (2022) proposed the *iterative trimmed maximum likelihood estimator* against adversarially corrupted samples in General Linear Models (GLM). The estimator is defined as follows, where

$S = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ represents the training data and f is an objective function for a GLM.

$$\hat{\boldsymbol{\theta}}(S) = \min_{\boldsymbol{\theta}} \min_{\hat{S} \subset S, |\hat{S}| = (1-\epsilon)n} \sum_{(\mathbf{x}_i, y_i) \in S} -\log f(y_i | \mathbf{x}_i^\top \boldsymbol{\theta}) \tag{3}$$

This estimator is proven to return near-optimal risk on a variety of linear models, including Gaussian regression, Poisson regression, and binomial regression with label and covariate corruption. In each iteration, a certain number of samples are trimmed and not resampled.

3 Subquantile Optimization

Definition 1. *Let F_X represent the Cumulative Distribution Function (CDF) of the random variable X. The* **p-*Quantile*** *of a Random Variable X is defined as follows*

$$Q_p(p) = \inf\{x \in \mathbb{R} : p \leq F(x)\} \tag{4}$$

Definition 2. *Let ℓ be the loss function.* ***Risk*** *is defined as follows*

$$U = \mathbb{E}_{(\mathbf{x},y)\sim\mathbb{P}} \left[\ell\left(f(\mathbf{x}; \boldsymbol{\theta}, y)\right)\right] \tag{5}$$

The **p-Quantile** of the Empirical Risk is given

$$\mathbb{L}_p(U) = \frac{1}{p}\int_0^p \mathcal{Q}_q(U)\, dq = \mathbb{E}\left[U | U \leq \mathcal{Q}_p(U)\right] = \max_{t\in\mathbb{R}} \left\{ t - \frac{1}{p}\mathbb{E}\left[(t-U)^+\right] \right\} \tag{6}$$

In equation 6, t represents the p-quantile of U. We also show that we can calculate t by a maximizing optimization function. The Subquantile Optimization problem is posed as follows:

$$\boldsymbol{\theta}_{SM} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^d} \max_{t\in\mathbb{R}} \left\{ t - \frac{1}{p}\mathbb{E}(t - \ell(f(\mathbf{x}; \boldsymbol{\theta}), y))^+ \right\} \tag{7}$$

This is a similar characterization to equation (4) in Laguel et al. (2021) We give the objective function for multiple learning tasks:
Linear Regression: $f(\mathbf{x}; \boldsymbol{\theta}, y) = \left(\mathbf{x}^\top \boldsymbol{\theta} - y\right)$
Logistic Regression: $f(\mathbf{x}; \boldsymbol{\theta}, y) = \log\left(1 + e^{-y(\mathbf{x}^\top \boldsymbol{\theta})}\right)$
Support Vector Machine (SVM):$f(\mathbf{x}; \boldsymbol{\theta}, y) = \left(1 - y_i(\mathbf{x}_i^\top \boldsymbol{\theta})\right)^+$
The objective function for Subquantile Minimization is:

$$g(t, \boldsymbol{\theta}) = t - \frac{1}{np}\sum_{i=1}^n \left(t - (\mathbf{x}_i^\top \boldsymbol{\theta} - y_i)^2\right)^+ \tag{8}$$

The two-step optimization for Subquantile optimization is given as follows

$$t_{(k+1)} = \arg\max_t g(t, \boldsymbol{\theta}_{(k)}) \tag{9}$$

$$\boldsymbol{\theta}_{(k+1)} = \boldsymbol{\theta}_{(k)} + \alpha \nabla_{\boldsymbol{\theta}_{(k)}} g(t_{(k+1)}, \boldsymbol{\theta}_{(k)}) \tag{10}$$

This algorithm is adopted from Razaviyayn et al. (2020). Theoretically, it has been proven to converge to a local nash equilibrium in Jin et al. (2019) when $g(t, \boldsymbol{\theta})$ is ℓ-smooth with respect to $\boldsymbol{\theta}$ and there exists

an ϵ-maximizer for $g(t, \boldsymbol{\theta})$ with respect to t .

Algorithm 1: Subquantile Minimization Mini-Batch Gradient Descent

Input: Epochs: T, Batch Size: $|B| = j$, Quantile p, Data Matrix: $\mathbf{X}, (n \times d), n \gg d$, Learning schedule: $\alpha_1, \cdots, \alpha_T$
Output: Trained Parameters, $\boldsymbol{\theta}_{(T)}$

$\boldsymbol{\theta}_{(0)} \leftarrow (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$
for $k \in 1, 2, \ldots, T$ **do**
  $S_{(k)} \leftarrow \textsc{Subquantile}(\boldsymbol{\theta}_{(k)}, X)$
  for $B \in \mathcal{B}$ **do**
    $L_{(b)} \leftarrow \frac{1}{|\mathcal{B}|} \sum_{b \in B} \nabla f(\mathbf{x}_b; \boldsymbol{\theta}, y_b)$
    $\boldsymbol{\theta}_{(k+1)} \leftarrow \boldsymbol{\theta}_k - \alpha_{(k)} L_{(b)}$
  end
end
return $\boldsymbol{\theta}_{(T)}$

Algorithm 2: Subquantile Minimization for Ridge Regression

Input: Training Epochs T,Quantile p
Output: Trained Parameters, $\boldsymbol{\theta}_{(T)}$

$\boldsymbol{\theta}_{(0)} \leftarrow (X^\top X + \lambda I)^{-1} X^\top \mathbf{y}$
for $k \in \{1, 2, \ldots, T\}$ **do**
  $S_{(k)} \leftarrow \textsc{Subquantile}(\boldsymbol{\theta}_{(k)}, X)$
  $\boldsymbol{\theta}_{(k+1)} \leftarrow (S_{(k)}^\top S_{(k)} + \lambda I)^{-1} S_{(k)}^\top \mathbf{y}_S$
end
return $\boldsymbol{\theta}_{(T)}$

Algorithm 3: Subquantile

Input: Parameters $\boldsymbol{\theta}$, Data Matrix: $X, (n \times d)$
Output: Subquantile Matrix S

$\hat{\boldsymbol{\nu}} \leftarrow \text{sorted}(X\boldsymbol{\theta}_{(k)} - \mathbf{y})^2$
$t \leftarrow \hat{\boldsymbol{\nu}}_{np}$
Let $\mathbf{x}_1, ..., \mathbf{x}_{np}$ be np points such that $\left(\mathbf{x}_i^\top \boldsymbol{\theta} - y_i\right)^2 \leq t$
$S \leftarrow \begin{pmatrix} \mathbf{x}_1^\top & \ldots & \mathbf{x}_{np}^\top \end{pmatrix}^\top$
return S

4 Theory

In this section, we will explore the fundamental aspects of $g(t, \boldsymbol{\theta})$. This will motivate the convergence analysis in the next section. Throughout this section we will denote $\mathbf{x}_i$ as the ith row in the data matrix X and y_i as its corresponding label. We will denote η_P as the number of data points from P within the subquantile, i.e. within the lowest np losses, and η_Q as the number of data points from Q within the subquantile. We also define $\varepsilon^{(t)} \triangleq \frac{\eta_P}{\eta_P + \eta_Q} = \frac{\eta_P}{np}$, as the ratio of corrupted points within the subquantile at optimization iteration t, where $p \in (0, 1)$ is the subquantile we are optimizing over.

4.1 Analysis of $g(t, \theta)$

Lemma 4.1. *$g(t_{k+1}, \boldsymbol{\theta}_k)$ is concave with respect to t.*

Proof. We provide a simple argument for concavity. Note t is a concave and convex function. Also $(\cdot)^+$ is a convex strictly non-negative function. Therefore we have a concave function minus the non-negative multiple of a summation of an affine function composed with a convex function. Therefore this is a concave function with respect to t. □

Lemma 4.2. *The maximizing value of t in $g(t, \boldsymbol{\theta})$ in t-update step of optimization as described by Equation 9 is maximized when $t = Q_p(U)$*

Since $g(t, \boldsymbol{\theta})$ with respect to t is a concave function. Maximizing $g(t, \boldsymbol{\theta})$ is equivalent to minimizing $-g(t, \boldsymbol{\theta})$. We will find fermat's optimality condition for the function $-g(t, \boldsymbol{\theta})$, which is convex. Let $\hat{\boldsymbol{\nu}} = \textit{sorted}\left((\boldsymbol{\theta}^\top X - y)^2\right)$ and note $0 < p < 1$

$$\partial(-g(t, \boldsymbol{\theta})) = -1 + \frac{1}{np} \sum_{i=1}^{n} \left\{ \begin{array}{ll} 1, & \text{if } t > \hat{\boldsymbol{\nu}}_i \\ 0, & \text{if } t < \hat{\boldsymbol{\nu}}_i \\ [0, 1], & \text{if } t = \hat{\boldsymbol{\nu}}_i \end{array} \right\} \tag{11}$$

$$= 0 \text{ when } t = \hat{\boldsymbol{\nu}}_{np} \tag{12}$$

This is the p-quantile of U. A full derivation is provided in Appendix B.1.

Lemma 4.3. *Let $t = \hat{\boldsymbol{\nu}}_{np}$. The $\boldsymbol{\theta}$-update step described in Equation **??** is equivalent to minimizing the least squares loss of the np elements with the lowest squared loss. Let $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ be ordered such that $f(\mathbf{x}_1; \boldsymbol{\theta}, y_1) \leq f(\mathbf{x}_2; \boldsymbol{\theta}, y_2) \leq \cdots \leq f(\mathbf{x}_n; \boldsymbol{\theta}, y_n)$, it then follows:*

$$\nabla_{\boldsymbol{\theta}} g(t_{(k+1)}, \boldsymbol{\theta}_{(k)}) = \frac{1}{np} \sum_{i=1}^{np} \nabla_{\boldsymbol{\theta}} f(\mathbf{x}_i; \boldsymbol{\theta}, y_i) \tag{13}$$

We provide a derivation in Appendix B.2. However, this result is quite intuitive as it shows we are optimizing over the p Subquantile of the Risk.

Interpretation 1. *Subquantile Minimization continously minimizes the risk over the p-quantile of the error. In each iteration, this means we reduce the error of the points within the lowest np errors.*

Lemma 4.4. *Let $f(\mathbf{x}; \boldsymbol{\theta}, y)$ be a convex function with respect to $\boldsymbol{\theta}$. The elements $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$ are ordered such that $f(\mathbf{x}_1; \boldsymbol{\theta}, y_1) \leq f(\mathbf{x}_2; \boldsymbol{\theta}, y_2) \leq \cdots \leq f(\mathbf{x}_n; \boldsymbol{\theta}, y_n)$. It then follows $g(t_{k+1}, \boldsymbol{\theta}_k)$ is convex with respect to $\boldsymbol{\theta}_k$.*

Proof. We see by lemma 4.2 and interpretation 1, we are optimizing by the np points with the lowest squared error. Mathematically,

$$\begin{aligned} g(t_{(k+1)}, \boldsymbol{\theta}_{(k)}) &= t_{(k+1)} - \frac{1}{np} \sum_{i=1}^{n} \left(t_{(k+1)} - f(\mathbf{x}_i; \boldsymbol{\theta}_{(k)}, y_i) \right)^+ \\ &= t_{(k+1)} - t_{(k+1)} + \frac{1}{np} \sum_{i=1}^{np} f(\mathbf{x}_i; \boldsymbol{\theta}_{(k)}, y_i) = \frac{1}{np} \sum_{i=1}^{np} f(\mathbf{x}_i; \boldsymbol{\theta}_{(k)}, y_i) \end{aligned}$$

Now we can make a simple argument for convexity. We have a non-negative multiple of the sum of the composition of an affine function with a convex function. Thus $g(t, \boldsymbol{\theta})$ is convex with respect to $\boldsymbol{\theta}$. □

Lemma 4.5. *$g(t, \boldsymbol{\theta})$ is L-smooth with respect to $\boldsymbol{\theta}$ with $L = \left\| \frac{2}{np} \sum_{i=1}^{np} \|\mathbf{x}_i\|^2 \right\|$*

Now we will state two properties regarding the effect of the t-update step and the $\boldsymbol{\theta}$-update step as described in Equations 9 and 10, respectively.

Lemma 4.6. *If $t_{k+1} \leq t_k$ then $g(t_{k+1}, \boldsymbol{\theta}_k) = g(t_k) + \frac{1}{np} \sum_{i=np}^{n} (t_k - \hat{\boldsymbol{\nu}}_i)^+$. If $t_{k+1} > t_k$, then $g(t_{k+1}, \boldsymbol{\theta}_k) = g(t_k) + \frac{1}{np} \sum_{i=n(p-\delta)}^{np} (t - \hat{\boldsymbol{\nu}}_i)^+ - \delta t$. For a small δ. Note $\hat{\boldsymbol{\nu}}$ represents the ascending order of the sorted errors over all data points with respect to $\boldsymbol{\theta}_{(k)}$.*

Clearly, this result is not overly intuitive, thus it is difficult to analyze the convergence of this algorithm as the effect of t on the objective function g is not consistent. Empirically, we find t is not a monotonically decreasing value. Therefore, in the next section, we will provide a different characterization of g so we can better analyze its convergence.

4.2 Optimization

We first start with a novel characterization of the Subquantile minimization algorithm.

From the intuitions we have gained on Subquantile Minimization. We can state the following:

Theorem 1. *Let $g(t, \boldsymbol{\theta})$ be differentiable and $f(\mathbf{x}; \boldsymbol{\theta}, y)$ be L-smooth in $\boldsymbol{\theta}$. Let $t_{(k)}$ and $\boldsymbol{\theta}_{(k)}$ be iterates from algorithm 1. Then, $\lim_{k \to \infty} \mathbb{E}\left[\left\| \nabla_{\boldsymbol{\theta}} g(t_{(k+1)}, \boldsymbol{\theta}_{(k)}) \right\| \right] = 0$.*

Proof Sketch.

$$\arg\min_{\boldsymbol{\theta}\in\mathbb{R}^d}\min_{\mathbf{S}\in\Pi}\left(\frac{1}{np}\sum_{\mathbf{x}\in S}f(\mathbf{x}_i;\boldsymbol{\theta},y_i)\right)\iff\arg\min_{\boldsymbol{\theta}\in\mathbb{R}^d}\max_{t\in\mathbb{R}}\left(t-\frac{1}{np}\sum_{i=1}^{np}\left(t-f(\mathbf{x}_i;\boldsymbol{\theta},y_i)\right)^+\right)\tag{14}$$

where Π represents the $\binom{n}{np}$ set of Subquantile matrices. This characterization of the min-max optimization problem into a min-min optimization problem allows us to intuitively see the convergence properties of this algorithm. □

We are solving a min-max convex-concave problem, thus we are looking for a Nash Equilibrium Point.

Definition 3. *$(t^*,\boldsymbol{\theta}^*)$ is a **Nash Equilibrium** of g if $\nabla_t g(t^*,\boldsymbol{\theta}^*)=0$ and $\nabla_{\boldsymbol{\theta}}g(t^*,\boldsymbol{\theta}^*)=\mathbf{0}$*

Definition 4. *Let $\mathbf{X}$ be a $(n\times d)$ data matrix and $f(x;\boldsymbol{\theta},y)$ be a convex differentiable loss function. Then $\boldsymbol{\theta}\in\mathbb{R}^d$ is a γ-approximal stationary point of $\mathbf{X}$ if $\frac{1}{n}\sum_{i=1}^n\|\nabla_{\boldsymbol{\theta}}f(\mathbf{x}_i;\boldsymbol{\theta},y_i)\|\leq\gamma$.*

We are now interested in what it means to be at a Local Nash Equilibrium. By Proposition 3, this means both first-order partial derivatives are equal to 0. By lemma 4.2, we have shown $\nabla_t g(t,\boldsymbol{\theta})=0$ when $\boldsymbol{\nu}_{np}\leq t<\boldsymbol{\nu}_{np+1}$. Furthermore, by lemma 4.3, we have shown $\nabla_{\boldsymbol{\theta}}(g,\boldsymbol{\theta})=0$ when the least squares error is minimized for the np points with lowest squared error. This means that for a subset of np points from X, the least squares error is minimized. What we are interested in is how many points within those np points come from P and how many of those points from Q. Our goal is to minimize the number of points within the np lowest squared losses from Q, as they will introduce error to our predictions on points from P.

4.3 Convergence of Algorithm 1

Lemma 4.7. *The expected value of error on points in P will be lower than the expected value of error on points in Q if $\left\|\text{proj}_{\boldsymbol{\beta}_P}(\boldsymbol{\theta})-\boldsymbol{\beta}_P\right\|<\left\|\text{proj}_{\boldsymbol{\beta}_Q}(\boldsymbol{\theta})-\boldsymbol{\beta}_Q\right\|$*

Lemma 4.7 gives us an intuitive result. If in each optimization step, our projection on $\boldsymbol{\beta}_P$ is closer than our projection on to $\boldsymbol{\beta}_Q$, we know the number of data points from $\mathbb{Q}$ in the Subquantile will increase from the previous iteration.

Theorem 2. *Given a data matrix $X=\begin{pmatrix}P\\Q\end{pmatrix}$ where the rows of P are sampled from $\mathcal{N}_d(\mathbf{0},\xi_P I)$ and the rows of Q are sampled from $\mathcal{N}_d(\mathbf{0},\xi_Q I)$. After a $\boldsymbol{\theta}$ update,*

$$\left\|\text{proj}_{\boldsymbol{\beta}_P}\boldsymbol{\theta}_{(t+1)}-\boldsymbol{\beta}_P\right\|-\left\|\text{proj}_{\boldsymbol{\beta}_P}\boldsymbol{\theta}_{(t)}-\boldsymbol{\beta}_P\right\|<\left\|\text{proj}_{\boldsymbol{\beta}_Q}\boldsymbol{\theta}_{(t+1)}-\boldsymbol{\beta}_Q\right\|-\left\|\text{proj}_{\boldsymbol{\beta}_Q}\boldsymbol{\theta}_{(t)}-\boldsymbol{\beta}_Q\right\|$$

if the following holds

$$\left\|\left(\alpha_1^{(t)}\left(\Xi^{(t)}-1\right)+\gamma\left(1-\varepsilon^{(t)}\right)\xi_P\right)\boldsymbol{\beta}_P\right\|>\left\|\left(\alpha_2^{(t)}\left(\Xi^{(t)}-1\right)+\gamma\varepsilon^{(t)}\xi_Q\right)\boldsymbol{\beta}_Q\right\|\tag{15}$$

where α_1 and α_2 represents the coefficients for the linear combination of $\boldsymbol{\theta}$ in the basis defined as $\mathbf{B}=\begin{bmatrix}\boldsymbol{\beta}_P & \boldsymbol{\beta}_Q & \mathbf{R}\end{bmatrix}$ and $\Xi^{(t)}\triangleq\left(1-\gamma\left(\left(1-\varepsilon^{(t)}\right)\xi_P+\varepsilon^{(t)}\xi_Q\right)\right)$ and $\gamma\triangleq np\alpha$ where α is the learning rate.

Theorem 3. *Given a data matrix $X=\begin{pmatrix}P\\Q\end{pmatrix}$ where the rows of P are sampled from $\mathcal{N}_d(\mathbf{0},\xi_P I)$ and the rows of Q are sampled from $\mathcal{N}_d(\mathbf{0},\xi_Q I)$. Given $\varepsilon^{(t)}$ of the subquantile at iteration t and $\varepsilon^{(t+1)}$ at iteration $t+1$, assuming $\varepsilon^{(t+1)}<\varepsilon^{(t)}$, the subquantile update improves the objective function in expectation by :*

$$\mathbb{E}\left[f\left(\boldsymbol{\theta}_{(k)},S_{(k+1)}\right)-f\left(\boldsymbol{\theta}_{(k)},S_{(k)}\right)\right]=n\left(\varepsilon^{(t)}-\varepsilon^{(t-1)}\right)\left(\xi_P\left\|\boldsymbol{\theta}_{(k)}^\top-\boldsymbol{\beta}_P^\top\right\|_2^2+\xi_Q\left\|\boldsymbol{\theta}_{(k)}^\top-\boldsymbol{\beta}_Q^\top\right\|_2^2\right)$$

4.4 Convergence of Algorithm 2

Theorem 4. *Given a subquantile matrix* $S = \begin{pmatrix} P \\ Q \end{pmatrix}$ *where the rows of* P *are sampled from* $\mathcal{N}_d(\mathbf{0}, \xi_P I)$ *and the rows of* Q *are sampled from* $\mathcal{N}_d(\mathbf{0}, \xi_Q I)$ *and for all data points in the subquantile matrix it holds* $(\mathbf{x}_i\boldsymbol{\theta} - y_i) \leq t_{(k)}$. *Let* $\sigma_{\max}(P)$ *be the maximum singular value of* P *and* $\sigma_{\max}(Q)$ *be the maximum singular value of* Q. *Then at any iteration, it holds:*

$$\left\|\boldsymbol{\beta}_P - \boldsymbol{\theta}_{(k)}\right\|_2 \leq \frac{2\sigma^2_{\max}(P)\left\|\boldsymbol{\beta}_P\right\| + 6\sigma_{\max}(P)n(1-\varepsilon^{(k)})\eta_P + \sigma^2_{\max}(Q)\left\|\boldsymbol{\beta}_Q\right\| + 3\sigma_{\max}(Q)n\varepsilon^{(k)}\eta_Q}{\sqrt{\sigma^2_{\max}(P) + \lambda}}$$

where $\mathrm{Var}(\epsilon_P) = \eta_P$ *and* $\mathrm{Var}(\epsilon_Q) = \eta_Q$

Theorem 4 gives us a bound on the 2-norm distance from the optimal parameters in each iteration.

Lemma 4.8. *Given a subquantile matrix* $X = \begin{pmatrix} P \\ Q \end{pmatrix}$ *where the rows of* P *are sampled from* $\mathcal{N}_d(\mathbf{0}, \xi_P I)$ *and the rows of* Q *are sampled from* $\mathcal{N}_d(\mathbf{0}, \xi_Q I)$, *where there are* ℓ *data points of* P *and* m *data points of* Q. *Then the PDF of the hard convergence distribution,i.e. there are no points from* Q *within the subquantile, is given as follows:*

$$h(x) = \ell f_P(x)\left(F_P(x)\right)^{\ell-1}\left(1 - F_Q(x)\right)^m \tag{16}$$

where $f_P(x) = \frac{1}{\sqrt{2\pi}}\left(\frac{x}{\phi}\right)$ *is the PDF for the* χ^2 *distribution with* 1 *degree of freedom, and similarly* $F_P(z) = \Phi\left(\frac{\sqrt{z}}{\sqrt{\phi}}\right) - \Phi\left(\frac{-\sqrt{z}}{\sqrt{\phi}}\right)$ *and* $F_Q(z) = \Phi\left(\frac{\sqrt{z}}{\sqrt{\psi}}\right) - \Phi\left(\frac{-\sqrt{z}}{\sqrt{\psi}}\right)$ *where* $\phi = \left\|\boldsymbol{\theta} - \boldsymbol{\beta}_P\right\|_2$ *and* $\psi = \left\|\boldsymbol{\theta} - \boldsymbol{\beta}_Q\right\|_2$

This formulation follows from the probability theory concept of order statistics and the normal distribution of the data vectors. Intuitively, if ϕ is significantly smaller than ψ the probability of hard-convergence should be higher. The maximal value of $h(x)$ will be at $\mathbb{E}\left[F_{P(\ell)}\right]$, the ℓ-th order statistic of P.

5 Empirical Results

In all experiments, we utilize Algorithm 2 due to its fast convergence and strong theoretical properties. Ransac Fischler & Bolles (1981), and Huber Huber & Ronchetti (2009), are standard regression techniques implemented in `sklearn` for outlier detection.

5.1 Linear Regression

We now demonstrate SubQuantile Regression in the presence of Gaussian Random Noise.

In our first synthetic experiment, we run Algorithm 1 on synthetically generated structured linear regression data, the noise is sampled from a linear distribution that is dependent on the vector of X. Our results show the near optimal performance of Subquantile Minimization. The results and comparison with other methods can be seen in Table 5. We see in Table 5, Subquantile Minimization produces State of the Art Results in the Quadratic Regression Case. Furthermore, it performs significantly better than baseline methods in the high-noise regimes ($\epsilon = 0.4$), this is confirmed in both the small data and large data datasets. Please refer to Appendix G for more details on the `Structured Linear Regression` Dataset.

We provide results on the `Drug Discovery` Dataset in Diakonikolas et al. (2019) utilizing the noise procedure described in Li et al. (2020). For each algorithm, if possible we use Ridge Regression for its robust properties, otherwise we use typical least squares. SubQuantile Minimization, ERM, RANSAC, SEVER, TERM, and TMLE are all capable of Ridge Regression.

As we can see in Table 2, we obtain state of the art results throughout all noise regimes. This makes our model the strongest among the tested, due to our strength throughout the whole range of noises.

Objectives	Test RMSE (`Drug Discovery`)			
	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
ERM	$1.303_{(0.0665)}$	$1.790_{(0.0849)}$	$2.198_{(0.0645)}$	$2.623_{(0.1010)}$
CRR Bhatia et al. (2017)	$1.079_{(0.0899)}$	$1.125_{(0.0832)}$	$1.385_{(0.1372)}$	$1.725_{(0.1136)}$
STIR Mukhoty et al. (2019)	$1.087_{(0.1256)}$	$1.167_{(0.0750)}$	$1.403_{(0.0987)}$	$1.668_{(0.1142)}$
Robust Risk Osama et al. (2020)	$1.176_{(0.1110)}$	$1.336_{(0.1882)}$	$1.437_{(0.1723)}$	$1.800_{(0.0820)}$
TMLE Awasthi et al. (2022)	$1.094_{(0.1065)}$	$1.323_{(0.0758)}$	$1.578_{(0.0799)}$	$1.984_{(0.2020)}$
TERM Li et al. (2020)	$1.100_{(0.0948)}$	$1.126_{(0.1181)}$	$1.143_{(0.1058)}$	$1.160_{(0.0799)}$
SEVER Diakonikolas et al. (2019)	$\mathbf{1.066_{(0.0907)}}$	$\mathbf{1.042_{(0.0659)}}$	$\mathbf{1.058_{(0.0950)}}$	$\mathbf{1.052_{(0.1201)}}$
Huber Huber & Ronchetti (2009)	$1.412_{(0.0474)}$	$1.501_{(0.2918)}$	$2.231_{(0.9054)}$	$2.247_{(1.0399)}$
RANSAC Fischler & Bolles (1981)	$1.238_{(0.0529)}$	$1.643_{(0.1331)}$	$2.092_{(0.1935)}$	$2.679_{(0.1365)}$
Subquantile (Ours)	$\mathbf{0.994_{(0.1024)}}$	$\mathbf{0.995_{(0.1227)}}$	$\mathbf{1.012_{(0.1029)}}$	$\mathbf{1.029_{(0.1019)}}$
Genie ERM	$0.960_{(0.0845)}$	$0.982_{(0.0842)}$	$1.006_{(0.0879)}$	$1.017_{(0.1100)}$

Table 2: `Drug Discovery` Dataset. Empirical Risk over P with oblivious noise. The Genie has knowledge of where corruptions are so only trains on clean data in training set. We use $\lambda = 6$ for all methods capable of ridge regression. For subquantile, we use $p = 0.9$ for $\epsilon = 0.1$, $p = 0.85$ for $\epsilon = 0.2$, $p = 0.75$ for $\epsilon = 0.3$, and $p = 0.65$ for $\epsilon = 0.4$

5.2 Logistic Regression

The objective function for Logistic Regression is:

$$f(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1 + e^{-y_i(\mathbf{X}_i\boldsymbol{\theta})}\right) \tag{17}$$

In this section we describe a very tough logistic regression problem. Where fitting to more than a few outliers will change the model significantly. We will describe the inliers and outliers:

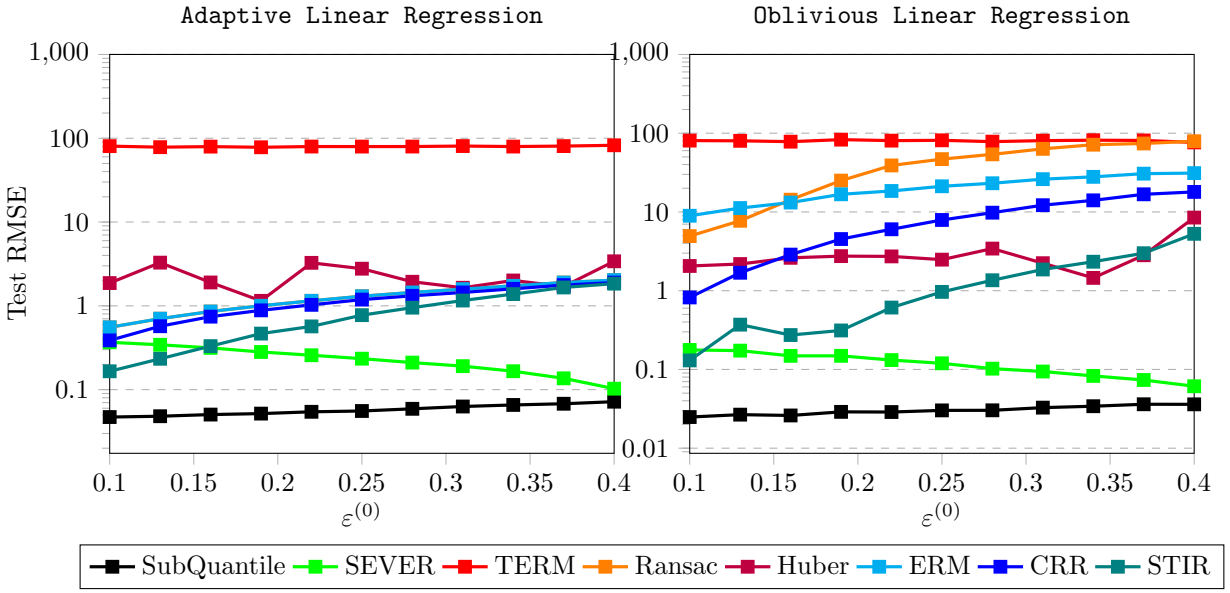

Figure 2: `Structured Linear Regression` & `Noisy Linear Regression` Datasets. Data is sampled from $\mathcal{N}_{200}(0, I_{200})$. Oblivious noise is sampled from $\mathcal{N}(5, 5)$, adaptive noise multiplies the labels by -1. TERM was ran with open source code, we are unsure why it performed poorly in the synthetic case.

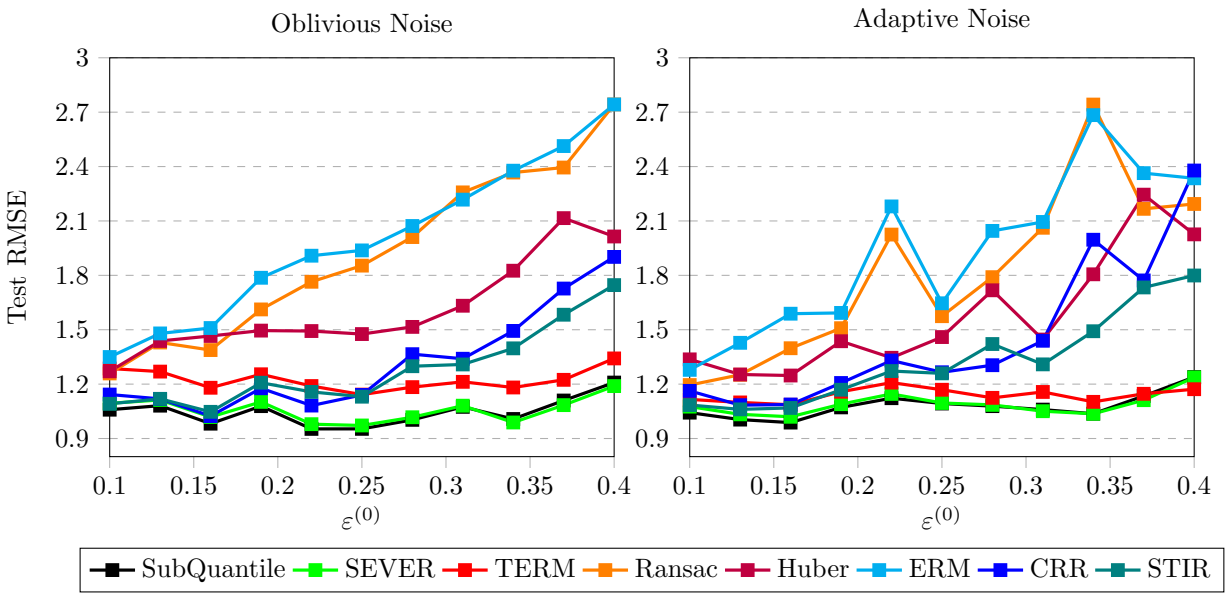

Figure 3: `Drug Discovery` Dataset with Normal Noise and Structured Noise. Oblivious noise is sampled from $\mathcal{N}(5,5)$, adaptive noise multiplies the labels by -1

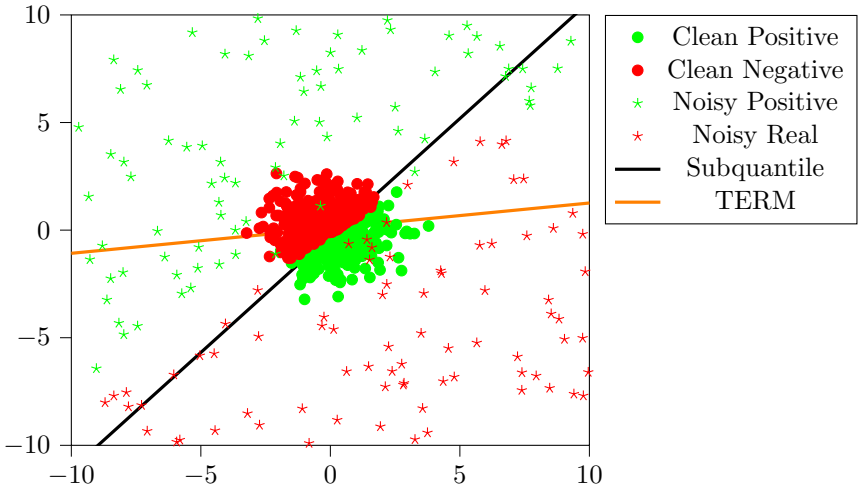

Figure 4: Logistic Regression

Inliers: $x_{\text{in}} \sim \mathcal{N}(0, I_2)$, $\mathbf{w} \sim \mathcal{N}(0, I_2)$, $b \sim \mathcal{N}(0,1)$, $y|\mathbf{x} \sim \text{sign}\left(\mathcal{N}(\mathbf{x}\mathbf{w} + b, 0.01)\right)$
Outliers: $x_{\text{out}} \sim \mathcal{U}\left([-10,10]^2\right)$, $y|\mathbf{x} \sim \text{sign}\left(-\mathbf{x}\mathbf{w} - b\right)$
$x_{\text{train}} = 800$, $X_{\text{test}} = 200$

Our comparisons are only against other robust meta algorithms.

5.3 Support Vector Machines

The objective function for SVM is:

$$f(\mathbf{x}; \boldsymbol{\theta}) = \frac{1}{n} \sum_{i=1}^{n} \left(1 - y_i \left(\mathbf{x}_i^\top \boldsymbol{\theta}\right)\right)_+ \tag{18}$$

Objectives	(`Logistic Regression`)		(`Logistic Regression`)	
	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
ERM	$0.620_{(0.0200)}$	$0.523_{(0.0299)}$	$0.648_{(0.2958)}$	$0.660_{(0.2817)}$
TERM Li et al. (2020)	$0.640_{(0.0067)}$	$0.642_{(0.0088)}$	$0.549_{(0.0263)}$	$0.041_{(0.0267)}$
SEVER Diakonikolas et al. (2019)	$0.786_{(0.1357)}$	$\mathbf{0.810_{(0.1290)}}$	$0.767_{(0.0717)}$	$\mathbf{0.872_{(0.0731)}}$
Subquantile (Ours)	$\mathbf{0.950_{(0.0500)}}$	$0.707_{(0.1825)}$	$\mathbf{0.802_{(0.1619)}}$	$\mathbf{0.875_{(0.1355)}}$
Genie ERM	$1.000_{(0.000)}$	$1.000_{(0.000)}$	$1.000_{(0.000)}$	$1.000_{(0.000)}$

Table 3: `Logistic Regression`. Test accuracy on P. Subquantile is trained with $p = (1 - \epsilon)$ and max 32 iterations.

Our comparisons are only against other robust meta algorithms.

Objectives	`Enron Spam Classification`		`Adult Dataset`	
	$\epsilon = 0.2$	$\epsilon = 0.4$	$\epsilon = 0.2$	$\epsilon = 0.4$
ERM	$0.633_{(0.0246)}$	$0.496_{(0.0608)}$	$0.785_{(0.0040)}$	$0.776_{(0.0033)}$
TERM Li et al. (2020)	∞	∞	∞	∞
SEVER Diakonikolas et al. (2019)	$0.649_{(0.0276)}$	$0.528_{(0.0305)}$	$0.681_{(0.0586)}$	$0.551_{(0.1020)}$
Subquantile (Ours)	$\mathbf{0.815_{(0.0204)}}$	$\mathbf{0.653_{(0.0490)}}$	$\mathbf{0.841_{(0.0185)}}$	$\mathbf{0.811_{(0.0040)}}$
Genie ERM	$0.963_{(0.0049)}$	$0.959_{(0.0041)}$	$0.847_{(0.0033)}$	$0.959_{(0.0041)}$

Table 4: Test accuracy on P. Subquantile is trained with $p = (1 - \epsilon)$ and max 32 iterations. SEVER is trained with 8 iterations and $p = 0.1$. Noisy labels are flipped from -1 to 1 or -1 to 1.

Subquantile Minimization has near optimal performance across all noise regimes in the `Adult Dataset` from the UCI Machine Learning Repository Dua & Graff (2017).

6 Conclusion

In this work we provide a theoretical analysis for robust linear regression by minimizing the *Subquantile* of the Empirical Risk. Furthermore, we run various numerical experiments and compare against the current State of the Art in Robust Linear Regression. Since minimizing over the subquantile is a general machine learning framework, it is scalable to larger scale machine learning problems. In future work, more real world applications can be explored and the theory can be expanded beyond linear regression. It is also possible to further explore the theorems in this paper to upper bound the number of iterations it takes for convergence of algorithm 2.

References

Pranjal Awasthi, Abhimanyu Das, Weihao Kong, and Rajat Sen. Trimmed maximum likelihood estimation for robust learning in generalized linear models, 2022. URL https://arxiv.org/abs/2206.04777.

Kush Bhatia, Prateek Jain, Parameswaran Kamalaruban, and Purushottam Kar. Consistent robust regression. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/e702e51da2c0f5be4dd354bb3e295d37-Paper.pdf.

Sam Corbett-Davies and Sharad Goel. The measure and mismeasure of fairness: A critical review of fair machine learning. *CoRR*, abs/1808.00023, 2018. URL http://arxiv.org/abs/1808.00023.

Ilias Diakonikolas and Daniel M. Kane. Recent advances in algorithmic high-dimensional robust statistics. *ArXiv*, abs/1911.05911, 2019.

Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. In *Proceedings of the 36th International Conference on Machine Learning*, ICML '19, pp. 1596–1606. JMLR, Inc., 2019.

Dheeru Dua and Casey" Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, jun 1981. ISSN 0001-0782. doi: 10.1145/358669.358692. URL https://doi.org/10.1145/358669.358692.

Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009. doi: 10.1109/TKDE.2008.239.

Peter J. Huber and Elvezio. Ronchetti. *Robust statistics*. Wiley series in probability and statistics. Wiley, Hoboken, N.J., 2nd ed. edition, 2009. URL http://catdir.loc.gov/catdir/toc/ecip0824/2008033283.html.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*, 2018.

Chi Jin, Praneeth Netrapalli, and Michael I. Jordan. What is local optimality in nonconvex-nonconcave minimax optimization?, 2019. URL https://arxiv.org/abs/1902.00618.

Ashish Khetan, Zachary C. Lipton, and Anima Anandkumar. Learning from noisy singly-labeled data. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1sUHgb0Z.

Yassine Laguel, Krishna Pillutla, Jérôme Malick, and Zaid Harchaoui. Superquantiles at work: Machine learning applications and efficient subgradient computation. *Set-Valued and Variational Analysis*, 29(4):967–996, Dec 2021. ISSN 1877-0541. doi: 10.1007/s11228-021-00609-w. URL https://doi.org/10.1007/s11228-021-00609-w.

Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. Tilted empirical risk minimization. *arXiv preprint arXiv:2007.01162*, 2020.

T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2999–3007, Los Alamitos, CA, USA, oct 2017. IEEE Computer Society. doi: 10.1109/ICCV.2017.324. URL https://doi.ieeecomputersociety.org/10.1109/ICCV.2017.324.

Bhaskar Mukhoty, Govind Gopakumar, Prateek Jain, and Purushottam Kar. Globally-convergent iteratively reweighted least squares for robust regression problems. In Kamalika Chaudhuri and Masashi Sugiyama (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 313–322. PMLR, 16–18 Apr 2019. URL https://proceedings.mlr.press/v89/mukhoty19a.html.

Muhammad Osama, Dave Zachariah, and Petre Stoica. Robust risk minimization for statistical learning from corrupted data. *IEEE Open Journal of Signal Processing*, 1:287–294, 2020.

Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics 'I& Probability Letters*, 33(3):291–297, 1997. URL https://EconPapers.repec.org/RePEc:eee:stapro:v:33:y:1997:i:3:p:291-297.

Meisam Razaviyayn, Tianjian Huang, Songtao Lu, Maher Nouiehed, Maziar Sanjabi, and Mingyi Hong. Non-convex min-max optimization: Applications, challenges, and recent theoretical advances, 06 2020.

D.D Wackerly, W. Mendenhall, and R.L. Scheaffer. *Mathematical Statistics with Applications,7th Edition*. Thompson Learning, Inc., USA, 2008.

A Linear Algebra and Probability Theory Preliminaries

Fact 1. *The spectral norm of a matrix, A, an $(m \times n)$ matrix, is defined as follows*

$$\|A\|_2 = \sqrt{\lambda_{\max}\left(A^\top A\right)} = \sigma_{\max}(A) \tag{19}$$

It similarly follows:

$$\left\|A^\top A\right\|_2 = \|A\|_2^2 \tag{20}$$

Fact 2. ***Weyl's Inequality*** *states the following:*
If M, N, and R are $n \times n$ Hermitian Matrices with the following eigenvalues where $M = N + R$:

$$M : \mu_1 \geq \cdots \geq \mu_n$$

$$N : \nu_1 \geq \cdots \geq \nu_n$$

$$R : \rho_1 \geq \cdots \geq \rho_n$$

Then the following equalities hold:

$$\nu_i + \rho_n \leq \mu_i \leq \nu_i + \rho_1 \text{ for } i = 1, \ldots, n$$

Fact 3. *Let A be a $n \times m$ matrix with $n \gg m$. It then follows:*

$$A^\top A = \left(U\Sigma V^\top\right)^\top \left(U\Sigma V^\top\right) = \left(V\Sigma^\top U^\top\right)\left(U\Sigma V^\top\right) = V\Sigma^\top \Sigma V^\top = VDV^\top \tag{21}$$

where $D = \Sigma^\top\Sigma = \begin{pmatrix} \sigma_1^2 & & & \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_m^2 \end{pmatrix}$

Fact 4. *This is a restatement from Wackerly et al. (2008).*
Let $X_1, \cdots, X_n$ be i.i.d continuous random variables with common distribution function $F(y)$ and common probability density function $f(x)$. If $X_{(k)}$ denotes the kth-order statistic, then the density function of $X_{(k)}$ is given by:

$$g_{(k)}(x_k) = \frac{n!}{(k-1)!(n-k)!}[F(x_k)]^{k-1}[1 - F(x_k)]^{n-k} f(x_k) \tag{22}$$

Fact 5. *The cdf of the kth order statistic from a sample of n is:*

$$F_{(k,n)} = \mathbb{P}\left[X_{(k)} \leq x\right] = \sum_{j=k}^{n} \binom{n}{j} \left(1 - F(x)\right)^{n-j} F(x)^j \tag{23}$$

Fact 6. ***Hoeffding bound****. Suppose the variables X_i, $i = 1, \ldots, n$ are i.i.d. with mean μ_i and sub-Gaussian parameter σ_i. Then for all $t \geq 0$, it follows:*

$$\mathbb{P}\left[\sum_{i=1}^{n}\left(X_i - \mu\right) \geq t\right] \leq \exp\left\{-\frac{t^2}{2\sum_{i=1}^{n}\sigma_i^2}\right\} \tag{24}$$

Fact 7. *A random variable X with mean $\mu = \mathbb{E}[X]$ is* ***sub-gaussian*** *if there exists σ such that:*

$$\mathbb{E}\left[e^{\lambda(X-\mu)}\right] \leq e^{\sigma^2\lambda^2/2} \, \forall \lambda \in \mathbb{R} \tag{25}$$

σ is denoted as the sub-gaussian parameter of X

B Theory for Subquantile Minimization Algorithm 1

B.1 Derivation of Lemma 4.2

Since $g(t, \boldsymbol{\theta})$ is a concave function. Maximizing $g(t, \boldsymbol{\theta})$ is equivalent to minimizing $-g(t, \boldsymbol{\theta})$. We will find fermat's optimality condition for the function $-g(t, \boldsymbol{\theta})$, which is convex. Let $\hat{\boldsymbol{\nu}} = \text{sorted}\left((\boldsymbol{\theta}^\top X - y)^2\right)$ and note $0 < p < 1$

$$\partial(-g(t, \boldsymbol{\theta})) = \partial\left(-t + \frac{1}{np}\sum_{i=1}^{n}(t - \hat{\boldsymbol{\nu}}_i)^+\right) \tag{26}$$

$$= \partial(-t) + \partial\left(\frac{1}{np}\sum_{i=1}^{n}(t - \hat{\boldsymbol{\nu}}_i)^+\right) \tag{27}$$

$$= -1 + \frac{1}{np}\sum_{i=1}^{n}\partial(t - \hat{\boldsymbol{\nu}}_i)^+ \tag{28}$$

$$= -1 + \frac{1}{np}\sum_{i=1}^{n}\left\{\begin{array}{ll} 1, & \text{if } t > \hat{\boldsymbol{\nu}}_i \\ 0, & \text{if } t < \hat{\boldsymbol{\nu}}_i \\ [0,1]\,, & \text{if } t = \hat{\boldsymbol{\nu}}_i \end{array}\right\} \tag{29}$$

$$= 0 \text{ when } t = \hat{\boldsymbol{\nu}}_{np} \tag{30}$$

This is the p-quantile of $\boldsymbol{\nu}$. Assuming no two points are equal in the dataset, this means the minimizing value for t has a range of values, $\hat{\boldsymbol{\nu}}_{np} \leq t < \hat{\boldsymbol{\nu}}_{np+1}$. This means $g(t, \boldsymbol{\theta})$ is not strongly convex with respect to t.

B.2 Derivation of Lemma 4.3

Note that $t_k = \boldsymbol{\nu}_{np}$ which is equivalent to $(\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np})^2$

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}_k} g(t_{k+1}, \boldsymbol{\theta}_k) &= \nabla_{\boldsymbol{\theta}_k}\left(\boldsymbol{\nu}_{np} - \frac{1}{np}\sum_{i=1}^{n}(\boldsymbol{\nu}_{np} - (\boldsymbol{\theta}_k^\top \mathbf{x}_i - y_i)^2)^+\right) \\
&= \nabla_{\boldsymbol{\theta}_k}\left((\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np})^2 - \frac{1}{np}\sum_{i=1}^{n}\left((\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np})^2 - (\boldsymbol{\theta}_k^\top \mathbf{x}_i - y_i)^2\right)^+\right) \\
&= \nabla_{\boldsymbol{\theta}_k}(\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np})^2 - \frac{1}{np}\sum_{i=1}^{n}\nabla_{\boldsymbol{\theta}_k}\left((\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np})^2 - (\boldsymbol{\theta}_k^\top \mathbf{x}_i - y_i)^2\right)^+ \\
&= 2\mathbf{x}_{np}(\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np}) - \frac{1}{np}\sum_{i=1}^{n}2\mathbf{x}_{np}(\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np}) \\
&\qquad - 2\mathbf{x}_i(\boldsymbol{\theta}_k^\top \mathbf{x}_i - y_i)\left\{\begin{array}{ll} 1, & \text{if } t > v_i \\ 0, & \text{if } t < v_i \\ [0,1]\,, & \text{if } t = v_i \end{array}\right\} \\
&= 2\mathbf{x}_{np}(\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np}) - \frac{1}{np}\sum_{i=1}^{np}2\mathbf{x}_{np}(\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np}) - 2\mathbf{x}_i(\boldsymbol{\theta}_k^\top \mathbf{x}_i - y_i) \\
&= 2\mathbf{x}_{np}(\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np}) - 2\mathbf{x}_{np}(\boldsymbol{\theta}_k^\top \mathbf{x}_{np} - y_{np}) + \frac{1}{np}\sum_{i=1}^{np}2\mathbf{x}_i(\boldsymbol{\theta}_k^\top \mathbf{x}_i - y_i) \\
&= \frac{1}{np}\sum_{i=1}^{np}2\mathbf{x}_i(\boldsymbol{\theta}_k^\top \mathbf{x}_i - y_i)
\end{aligned}$$

This is the derivative of the np samples with lowest error with respect to $\boldsymbol{\theta}$.

B.3 Derivation of Lemma 4.5

The objective function $g(\boldsymbol{\theta}, t)$ is L-smooth w.r.t $\boldsymbol{\theta}$ iff

$$||\nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}', t) - \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}, t)|| \leq L||\boldsymbol{\theta}' - \boldsymbol{\theta}|| \tag{31}$$

$$\left\|\nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}^{'}, t) - \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}, t)\right\| = \left\| \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i(\boldsymbol{\theta}_k^{'\top}\mathbf{x}_i - y_i) - \frac{1}{np}\sum_{i=1}^{np} 2\mathbf{x}_i(\boldsymbol{\theta}_k^{\top}\mathbf{x}_i - y_i) \right\| \tag{32}$$

$$= \left\| \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i(\boldsymbol{\theta}_k^{'\top}\mathbf{x}_i - \boldsymbol{\theta}_k^{\top}\mathbf{x}_i) \right\| \tag{33}$$

$$= \left\| \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i\mathbf{x}_i^{\top}(\boldsymbol{\theta}_k^{'\top} - \boldsymbol{\theta}_k^{\top}) \right\| \tag{34}$$

$$\overset{\text{Cauchy}-\text{Schwarz}}{\leq} \left\| \frac{2}{np} \sum_{i=1}^{np} \mathbf{x}_i\mathbf{x}_i^{\top} \right\| \left\| \boldsymbol{\theta}_k^{'\top} - \boldsymbol{\theta}_k^{\top} \right\| \tag{35}$$

$$= L \left\| \boldsymbol{\theta}_k^{'\top} - \boldsymbol{\theta}_k^{\top} \right\| \tag{36}$$

where $L = \left\| \frac{2}{np} X^{\top} X \right\|$
This concludes the derivation.

B.4 Proof of Lemma 4.6

Proof. We will investigate the two cases $t_{k+1} \leq t$ and $t_{k+1} > t_k$.
Case (i) $t_{k+1} \leq t_k$
Let us first expand out $g(t_k, \boldsymbol{\theta}_k)$ with the knowledge that $t_k \geq \hat{\boldsymbol{\nu}_k}$

$$g(t_k, \boldsymbol{\theta}_k) = t_k - \frac{1}{np}\sum_{i=1}^{n}(t_k - \boldsymbol{\nu}_i)^+ \tag{37}$$

$$= t_k - \frac{1}{np}(np)t_k + \frac{1}{np}\sum_{i=1}^{np}\boldsymbol{\nu}_i + \frac{1}{np}\sum_{i=np}^{n}(t_k - \boldsymbol{\nu}_i)^+ \tag{38}$$

$$= \frac{1}{np}\sum_{i=1}^{np}\boldsymbol{\nu}_i + \frac{1}{np}\sum_{i=np}^{n}(t_k - \boldsymbol{\nu}_i)^+ \tag{39}$$

$$g(t_{k+1}, \boldsymbol{\theta}_k) - g(t_k, \boldsymbol{\theta}_k) = \frac{1}{np}\sum_{i=1}^{np}\boldsymbol{\nu}_i - \left(\frac{1}{np}\sum_{i=1}^{np}\boldsymbol{\nu}_i + \frac{1}{np}\sum_{i=np}^{n}(t_k - \boldsymbol{\nu}_i)^+ \right) \tag{40}$$

$$= -\frac{1}{np}\sum_{i=np}^{n}(t_k - \boldsymbol{\nu}_i)^+ \tag{41}$$

Case (ii) $t_{k+1} > t_k$
Since we know t_k is less than $\boldsymbol{\nu}_{np}$, WLOG we will say t_k is greater than the lowest $n(p-\delta)$ elements, where

$\delta \in (0, p)$.

$$g(t_k, \boldsymbol{\theta}_k) = t_k - \frac{1}{np}\sum_{i=1}^{n}(t_k - \boldsymbol{\nu}_i)^+ \tag{42}$$

$$= t_k - \frac{1}{np}\sum_{i=1}^{n(p-\delta)}(t_k - \boldsymbol{\nu}_i)^+ \tag{43}$$

$$= t_k - \frac{1}{np}(n(p-\delta))t_k + \frac{1}{np}\sum_{i=1}^{n(p-\delta)}\boldsymbol{\nu}_i \tag{44}$$

$$g(t_k, \boldsymbol{\theta}_{k+1}) - g(t_k, \boldsymbol{\theta}_k) = \frac{1}{np}\sum_{i=1}^{np}\boldsymbol{\nu}_i - \left(\delta t_k + \frac{1}{np}\sum_{i=1}^{n(p-\delta)}\boldsymbol{\nu}_i\right) \tag{45}$$

$$= \left(\frac{1}{np}\sum_{i=n(p-\delta)}^{n}\boldsymbol{\nu}_i\right) - \delta t_k \tag{46}$$

This concludes the proof. □

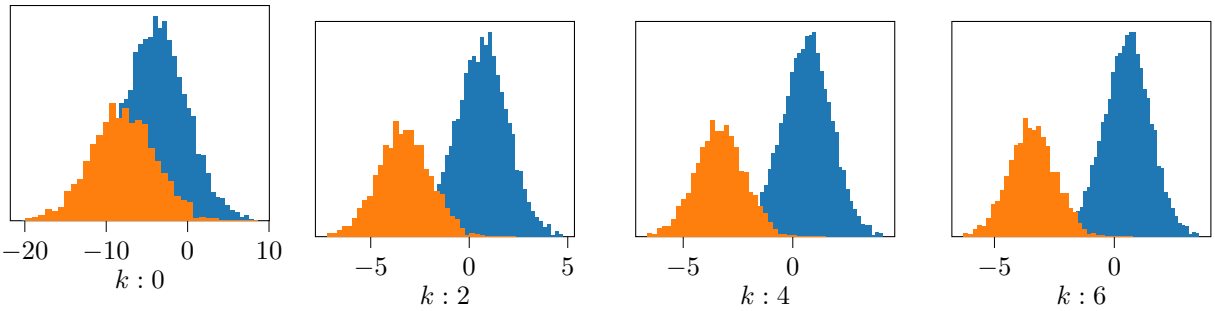

Figure 5: Residuals with respect to $\mathbb{P}$ and $\mathbb{Q}$, k represents optimization step.

C Theory for Adaptive Linear Corruption

In this section, we provide the conditions for Subquantile Minimization to improve in the case of corruption of the form $\boldsymbol{\beta}_Q^\top \boldsymbol{\mu} = y_p + \epsilon_Q$.

Assumption 1. *The residuals of $\boldsymbol{\theta}_k$ are normally distributed with respect to $\mathbb{P}$ and $\mathbb{Q}$. In other words, $\boldsymbol{\theta}_k^\top \mathbf{p} - \mathbf{y}_P$ and $\boldsymbol{\theta}_k^\top \mathbf{q} - \mathbf{y}_Q$ are normally distributed.*

Assumption 1 can be visually verified in figure 5. Even after multiple iteration steps the residuals with respect to $\mathbb{P}$ and $\mathbb{Q}$ are still normal. Thus it follows by decreasing $\|\boldsymbol{\theta} - \boldsymbol{\beta}_P\|_1$ more relative to $\|\boldsymbol{\theta} - \boldsymbol{\beta}_Q\|_1$ then the SubQuantile will contain more points from P by expectation.

C.1 Proof of Theorem 2

Proof. To show the change in ε we will first calculate the expected change in $\boldsymbol{\theta}$ by the $\boldsymbol{\theta}$-update described in Equation 10. We will also introduce some notation, $\mathcal{S}$ represents all $\mathbf{x} \in X$ that are within the lowest np losses, i.e. within the subquantile, $\mathbf{p} \in \mathcal{S}$ represent all data vectors from $\mathbb{P}$ that are within the SubQuantile, similarly $\mathbf{q} \in \mathbb{Q}$ represent all data vectors from $\mathbb{Q}$ that are within the SubQuantile. Furthermore, $|\mathcal{S}| = np$, there are εnp points from $\mathbb{Q}$ in $\mathcal{S}$ and $(1-\varepsilon)np$ points from $\mathbb{P}$ within $\mathcal{S}$. We assume $\mathbf{p} \sim \mathcal{N}(\mathbf{0}, \Sigma_P)$ and $\mathbf{q} \sim \mathcal{N}(\mathbf{0}, \Sigma_Q)$ where $\Sigma_P = \xi_P I$ and $\Sigma_Q = \xi_Q I$ where ξ_P and ξ_Q are greater than 0, then we can assume it follows $\boldsymbol{\mu}\boldsymbol{\mu}^\top = \mathbf{0}\mathbf{0}^\top = 0$.

$$
\begin{aligned}
\mathbb{E}\left[\boldsymbol{\theta}_{k+1}\right] &= \boldsymbol{\theta}_k - \mathbb{E}\left[\alpha \nabla g(\boldsymbol{\theta}_k, t_{k+1})\right] \\
&= \boldsymbol{\theta}_k - \alpha \mathbb{E}\left[\sum_{\mathbf{x}\in\mathcal{S}} \mathbf{x}(\boldsymbol{\theta}^\top \mathbf{x} - y)\right] \\
&= \boldsymbol{\theta}_k - \alpha \mathbb{E}\left[\sum_{\mathbf{x}\in\mathcal{S}} \mathbf{x}\mathbf{x}^\top \boldsymbol{\theta}_k - \mathbf{x} y\right] \\
&= \boldsymbol{\theta}_k - \alpha \mathbb{E}\left[\sum_{\mathbf{p}\in\mathcal{S}} \mathbf{p}\mathbf{p}^\top \boldsymbol{\theta}_k - \mathbf{p} y_p + \sum_{\mathbf{q}\in\mathcal{S}} \mathbf{q}\mathbf{q}^\top \boldsymbol{\theta}_k - \mathbf{q} y_q\right]
\end{aligned}
$$

We will use Assumption **??** to rewrite y_p and y_q

$$
\begin{aligned}
&= \boldsymbol{\theta}_k - \alpha \mathbb{E}\left[\sum_{\mathbf{p}\in\mathcal{S}} \mathbf{p}\mathbf{p}^\top \boldsymbol{\theta}_k - \mathbf{p}(\boldsymbol{\beta}_P \mathbf{p} + \epsilon_P) + \sum_{\mathbf{q}\in\mathcal{S}} \mathbf{q}\mathbf{q}^\top \boldsymbol{\theta}_k - \mathbf{q}(\boldsymbol{\beta}_Q \mathbf{p} + \epsilon_Q)\right] \\
&= \boldsymbol{\theta}_k - \alpha \left(\sum_{\mathbf{p}\in\mathcal{S}} \left(\boldsymbol{\mu}\boldsymbol{\mu}^\top + \Sigma_P\right)\boldsymbol{\theta}_k - \left(\boldsymbol{\mu}\boldsymbol{\mu}^\top + \Sigma_P\right)\boldsymbol{\beta}_P + \sum_{\mathbf{q}\in\mathcal{S}} \left(\boldsymbol{\mu}\boldsymbol{\mu}^\top + \Sigma_Q\right)\boldsymbol{\theta}_k - \left(\boldsymbol{\mu}\boldsymbol{\mu}^\top + \Sigma_Q\right)\boldsymbol{\beta}_Q\right)
\end{aligned}
$$

$$\mathbb{E}\left[\boldsymbol{\theta}_{(t+1)}\right] = \boldsymbol{\theta}_{(t)} - \alpha np\left((1-\varepsilon^{(t)})\xi_P(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_P) + \varepsilon^{(t)}\xi_Q(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_Q)\right) \tag{47}$$

Now that we have the expected update for $\boldsymbol{\theta}$ in terms of the linear regression coefficients, we now want to utilize Lemma 4.7.

Let $\boldsymbol{\theta}_{(t)} = \alpha_1^{(t)}\boldsymbol{\beta}_P + \alpha_2^{(t)}\boldsymbol{\beta}_Q + \sum_{i=3}^{d}\alpha_i^{(t)}\mathbf{r}_i$ in the same basis $\mathbf{B}$ defined in Lemma 4.7.Let $\gamma \triangleq \alpha np$. Then the following manipulations hold:

$$\begin{aligned}
\mathbb{E}\left[\boldsymbol{\theta}_{(t+1)}\right] &= \boldsymbol{\theta}_{(t)} - \gamma\left((1-\varepsilon^{(t)})\xi_P(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_P) + \varepsilon^{(t)}\xi_Q(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_Q)\right) \\
&= \boldsymbol{\theta}_{(t)}\left(1 - \gamma\left(\left(1-\varepsilon^{(t)}\right)\xi_P + \varepsilon^{(t)}\xi_Q\right)\right) + \gamma\left(1-\varepsilon^{(t)}\right)\xi_P\boldsymbol{\beta}_P + \gamma\varepsilon^{(t)}\xi_Q\boldsymbol{\beta}_Q \\
&= \left(1 - \gamma\left(\left(1-\varepsilon^{(t)}\right)\xi_P + \varepsilon^{(t)}\xi_Q\right)\right)\left(\alpha_1^{(t)}\boldsymbol{\beta}_P + \alpha_2^{(t)}\boldsymbol{\beta}_Q + \sum_{i=3}^{d}\alpha_i^{(t)}\mathbf{r}_i\right) + \gamma\left(1-\varepsilon^{(t)}\right)\xi_P\boldsymbol{\beta}_P + \gamma\varepsilon^{(t)}\xi_Q\boldsymbol{\beta}_Q
\end{aligned}$$

To simplify the notation, let us define the constant for this iteration $\Xi^{(t)} \triangleq \left(1 - \gamma\left(\left(1-\varepsilon^{(t)}\right)\xi_P + \varepsilon^{(t)}\xi_Q\right)\right)$

$$= \left(\alpha_1^{(t)}\Xi^{(t)} + \gamma\left(1-\varepsilon^{(t)}\right)\xi_P\right)\boldsymbol{\beta}_P + \left(\alpha_2^{(t)}\Xi^{(t)} + \gamma\varepsilon^{(t)}\xi_Q\right)\boldsymbol{\beta}_Q + \Xi^{(t)}\sum_{i=3}^{d}\alpha_i^{(t)}\mathbf{r}_i$$

We will now calculate the difference.

$$\mathbb{E}\left[\boldsymbol{\theta}_{(t+1)} - \boldsymbol{\theta}_{(t)}\right] = \left(\alpha_1^{(t)}\left(\Xi^{(t)} - 1\right) + \gamma\left(1-\varepsilon^{(t)}\right)\xi_P\right)\boldsymbol{\beta}_P + \left(\alpha_2^{(t)}\left(\Xi^{(t)} - 1\right) + \gamma\varepsilon^{(t)}\xi_Q\right)\boldsymbol{\beta}_Q + \left(\Xi^{(t)} - 1\right)\sum_{i=3}^{d}\alpha_i^{(t)}\mathbf{r}_i$$

Thus the conditions for ε to decrease by expectation are:

$$\left\|\left(\alpha_1^{(t)}\left(\Xi^{(t)} - 1\right) + \gamma\left(1-\varepsilon^{(t)}\right)\xi_P\right)\boldsymbol{\beta}_P\right\| > \left\|\left(\alpha_2^{(t)}\left(\Xi^{(t)} - 1\right) + \gamma\varepsilon^{(t)}\xi_Q\right)\boldsymbol{\beta}_Q\right\| \tag{48}$$

This concludes the proof. Note we are not interested in the change on the vectors $\mathbf{r}_3, \ldots, \mathbf{r}_d$ as they do not have an effect on the projection of $\boldsymbol{\theta}$ onto $\boldsymbol{\beta}_P$ and $\boldsymbol{\beta}_Q$. □

D Proofs for Convergence

D.1 Proof of Theorem 1

Proof. We will first start by introducing new notation. Let S represent a matrix with np data points from X, in other words it is a possible SubQuantile Matrix. Let $\mathbf{\Pi}$ represent the set of all such possible matrices S of X. Note $|\mathbf{\Pi}| = \binom{n}{np}$. We can now redefine the min-max optimization problem of g to a min-min optimization problem. Let us define the function $f(\boldsymbol{\theta}, S) = \left\|\boldsymbol{\theta}^\top S - y_S\right\|_2^2$

$$\boldsymbol{\theta}^*, S^* = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^d} \arg\min_{S\in\mathbf{\Pi}} \left\|\boldsymbol{\theta}^\top S - y_S\right\|_2^2 \tag{49}$$

Note we have a $\mathcal{O}(n)$ time-complexity oracle for the $\arg\min_{S\in\mathbf{\Pi}} f(\boldsymbol{\theta}_T, S_T)$.

Lemma D.1. *The resultant $\widetilde{S} = \arg\min_{S\in\mathbf{\Pi}} f(\boldsymbol{\theta}, S)$ is a unique minimizer iff all points in X are different.*

We will now show f is a monotonically decreasing function.
First let us define $\phi(\cdot) = \min_{S\in\mathbf{\Pi}}(\cdot, S)$. Let us also note f is ℓ smooth with respect to $\boldsymbol{\theta}$. This is following notation from Jin et al. (2019).It thus follows:

$$\begin{aligned}
f(\boldsymbol{\theta}_{k+1}, S_k) &\le f(\boldsymbol{\theta}_k, S_k) + \langle \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_k, S_k), \boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k\rangle + \frac{\ell}{2}\left\|\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k\right\|_2^2 \\
&= \phi(\boldsymbol{\theta}_k) + \langle \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_k, S_k), -\frac{1}{\ell}\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_k, S_k)\rangle + \frac{\ell}{2}\left\|\frac{1}{\ell}\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_k, S_k)\right\|_2^2 \\
&= \phi(\boldsymbol{\theta}_k) - \frac{1}{\ell}\left(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_k, S_k)\right)^2 + \frac{1}{2\ell}\left(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_k, S_k)\right)^2 \\
&= \phi(\boldsymbol{\theta}_k) - \frac{1}{2\ell}\left(\nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}_k, S_k)\right)^2
\end{aligned}$$

Thus we have proved the inner optimization problem is monotonically decreasing. Since the outer minimization is strictly less than or equal to the result from the inner optimization, it follows after each two step optimization:

$$\phi(\boldsymbol{\theta}_{k+1}) \le \phi(\boldsymbol{\theta}_k)$$

Since f is lower bounded by 0. We can invoke Montonicity Convergence Theorem, since f is a monotically decreasing function and is lower bounded, it therefore converges to either a local or global minimum. □

D.2 Proof of Theorem 3

Proof. Recall $\varepsilon^{(t)}$ represents the ratio of points of Q within the subquantile matrix at iteration t and $\varepsilon^{(t+1)}$ represents the ratio of points of Q within the subquantile matrix at iteration $t+1$. If $\varepsilon^{(t)} = \varepsilon^{(t+1)}$, then we would expect 0 improvement.

$$\begin{aligned}
\mathbb{E}\left[f\left(\boldsymbol{\theta}_{(k)}, S^{(k+1)}\right) - f\left(\boldsymbol{\theta}_{(k)}, S^{(k)}\right)\right] &= \mathbb{E}\left[\sum_{i=1}^{n(\varepsilon^{(t)}-\varepsilon^{(t-1)})} \left\|\boldsymbol{\theta}_{(k)}^\top \mathbf{p}_i - y_i\right\|_2^2 - \left\|\boldsymbol{\theta}_{(k)}^\top \mathbf{q}_i - y_i\right\|_2^2\right] \\
&= \mathbb{E}\left[\sum_{i=1}^{n(\varepsilon^{(t)}-\varepsilon^{(t-1)})} \left\|\boldsymbol{\theta}_{(k)}^\top \mathbf{p}_i - y_i\right\|_2^2 - \left\|\boldsymbol{\theta}_{(k)}^\top \mathbf{q}_i - y_i\right\|_2^2\right]
\end{aligned}$$

We can now use reverse triangle inequality.

$$
\begin{aligned}
&\leq \mathbb{E}\left[\sum_{i=1}^{n\left(\varepsilon^{(t)}-\varepsilon^{(t-1)}\right)} \left\|\boldsymbol{\theta}_{(k)}^{\top}\mathbf{p}_i - \boldsymbol{\beta}_P\mathbf{p}_i - \epsilon_P - \boldsymbol{\theta}_{(k)}^{\top}\mathbf{q}_i + \boldsymbol{\beta}_Q\mathbf{q}_i + \epsilon_Q\right\|_2^2\right] \\
&= \mathbb{E}\left[\sum_{i=1}^{n\left(\varepsilon^{(t)}-\varepsilon^{(t-1)}\right)} \left\|\left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_P\right)\mathbf{p}_i + \left(\boldsymbol{\beta}_Q - \boldsymbol{\theta}_{(k)}^{\top}\right)\mathbf{q}_i - \epsilon_P + \epsilon_Q\right\|_2^2\right] \\
&\leq \mathbb{E}\left[\sum_{i=1}^{n(\varepsilon^{(t)}-\varepsilon^{(t-1)})} \left\|\left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_P^{\top}\right)\mathbf{p}_i\right\|_2^2 + \left\|\left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_Q^{\top}\right)\mathbf{q}_i\right\|_2^2 + \|\epsilon_P\| + \|\epsilon_Q\|\right]
\end{aligned}
$$

Note that $\mathbb{E}\left[X^2\right] = \mathbb{E}\left[X\right]^2 + \mathrm{Var}(X)$ for a random variable X then:

$$
\begin{aligned}
&\leq \sum_{i=1}^{n(\varepsilon^{(t)}-\varepsilon^{(t-1)})} \left(\left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_P^{\top}\right)\mathbb{E}\left[\mathbf{p}_i\right]\right)^2 + \left(\left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_Q^{\top}\right)\mathbb{E}\left[\mathbf{q}_i\right]\right)^2 + \left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_P^{\top}\right)\mathrm{Var}\left(\mathbf{p}_i\right)\left(\boldsymbol{\theta}_{(k)} - \boldsymbol{\beta}_P\right) \\
&\quad + \left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_Q^{\top}\right)\mathrm{Var}(\mathbf{q}_i)\left(\boldsymbol{\theta}_{(k)} - \boldsymbol{\beta}_Q\right) \\
&= \sum_{i=1}^{n(\varepsilon^{(t)}-\varepsilon^{(t-1)})} \left(\left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_P^{\top}\right)\mathbb{E}\left[\mathbf{p}_i\right]\right)^2 + \left(\left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_Q^{\top}\right)\mathbb{E}\left[\mathbf{q}_i\right]\right)^2 + \left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_P^{\top}\right)\Sigma_P\left(\boldsymbol{\theta}_{(k)} - \boldsymbol{\beta}_P\right) \\
&\quad + \left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_Q^{\top}\right)\Sigma_Q\left(\boldsymbol{\theta}_{(k)} - \boldsymbol{\beta}_Q\right)
\end{aligned}
$$

If we assume the data is centered around $\mathbf{0}$, then it simplifies to the following:

$$
= n\left(\varepsilon^{(t)} - \varepsilon^{(t-1)}\right)\left(\left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_P^{\top}\right)\Sigma_P\left(\boldsymbol{\theta}_{(k)} - \boldsymbol{\beta}_P\right) + \left(\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_Q^{\top}\right)\Sigma_Q\left(\boldsymbol{\theta}_{(k)} - \boldsymbol{\beta}_Q\right)\right) \tag{50}
$$

Furthermore, since we typically assume linear independence in the data, let $\Sigma_P = \xi_P I$ and $\Sigma_Q = \xi_Q I$ where ξ_P and ξ_Q are constants greater than 0.

$$
= n\left(\varepsilon^{(t)} - \varepsilon^{(t-1)}\right)\left(\xi_P\left\|\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_P^{\top}\right\|_2^2 + \xi_Q\left\|\boldsymbol{\theta}_{(k)}^{\top} - \boldsymbol{\beta}_Q^{\top}\right\|_2^2\right) \tag{51}
$$

□

E Theory for Ridge Regression Algorithm 2

E.1 Proof of Theorem 4

Proof.

Assumption 2. *The rows of P and Q are sampled from $\mathbf{0}$ centered Normal Distributions.*

$$P_i \sim \mathcal{N}(\mathbf{0}, \Sigma_P)$$

$$Q_i \sim \mathcal{N}(\mathbf{0}, \Sigma_Q) \tag{52}$$

Assumption 3. *By assumption 2, it thus follows that the matrices $P^\top P$ and $Q^\top Q$ are sampled from Wishart Distributions.*

$$P^\top P \sim \mathcal{W}(n, \Sigma_P) \tag{53}$$

$$Q^\top Q \sim \mathcal{W}(n, \Sigma_Q) \tag{54}$$

Assumption 4. *Similar to the assumption made in Bhatia et al. (2017), to give theoretical bounds on the our algorithm, we assume the following:*

$$\Sigma_P = \xi_P I \tag{55}$$

$$\Sigma_Q = \xi_Q I \tag{56}$$

where $\xi_P, \xi_Q \geq 0$

The closed form solution for Ridge Regression with regularization parameter λ is equal to the following:

$$\hat{\boldsymbol{\beta}} = \left(X^\top X + \lambda I\right)^{-1} X^\top y \tag{57}$$

We will use this to bound the difference of the $\boldsymbol{\beta}_P$ and $\boldsymbol{\theta}_{(k)}$.

$$\left\|\boldsymbol{\beta}_P - \boldsymbol{\theta}_{(k)}\right\|_2 = \left\|\boldsymbol{\beta}_P - \left(S^\top S + \lambda I\right)^{-1} S^\top \mathbf{y}\right\|_2$$

Note the subquantile matrix S, consists of data points from P and Q, we will reorganize S into the following:

$$S = \begin{pmatrix} P \\ Q \end{pmatrix} = \begin{pmatrix} \leftarrow & \mathbf{p}_1 & \rightarrow \\ \vdots & \vdots & \vdots \\ \leftarrow & \mathbf{p}_{n(1-\varepsilon^{(t)})} & \rightarrow \\ \leftarrow & \mathbf{q}_1 & \rightarrow \\ \vdots & \vdots & \vdots \\ \leftarrow & \mathbf{q}_{n\varepsilon^{(t)}} & \rightarrow \end{pmatrix} \text{ and } \mathbf{y} = \begin{pmatrix} \mathbf{y}_P \\ \mathbf{y}_Q \end{pmatrix} = \begin{pmatrix} \boldsymbol{\beta}_P \mathbf{p}_1 + \epsilon_P \\ \vdots \\ \boldsymbol{\beta}_P \mathbf{p}_{(n(1-\varepsilon^{(t)}))} + \epsilon_P \\ \boldsymbol{\beta}_Q \mathbf{q}_1 + \epsilon_Q \\ \vdots \\ \boldsymbol{\beta}_Q \mathbf{q}_{n\varepsilon^{(t)}} + \epsilon_Q \end{pmatrix}$$

Let us also assume $\mathrm{Var}(\epsilon_P) = \eta_P$ and $\mathrm{Var}(\epsilon_Q) = \eta_Q$. Then we can make the following manipulations:

$$
\begin{aligned}
&= \left\| \boldsymbol{\beta}_P - \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} \left(P^\top Q^\top\right) \begin{pmatrix} \mathbf{y}_P \\ \mathbf{y}_Q \end{pmatrix} \right\|_2 \\
&= \left\| \left(P^\top P + \lambda I\right)^{-1} P^\top \mathbf{y}_P - \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} P^\top \mathbf{y}_P - \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} Q^\top \mathbf{y}_Q \right\|_2 \\
&\leq \left\| \left(P^\top P + \lambda I\right)^{-1} P^\top \mathbf{y}_P - \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} P^\top \mathbf{y}_P \right\|_2 + \left\| \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} Q^\top \mathbf{y}_Q \right\|_2 \\
&\leq \left\| \left(P^\top P + \lambda I\right)^{-1} \right\|_2 \left\| P^\top \mathbf{y}_P \right\|_2 + \left\| \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} \right\|_2 \left\| P^\top \mathbf{y}_P \right\|_2 + \left\| \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} \right\|_2 \left\| Q^\top \mathbf{y}_Q \right\|_2 \\
&\leq \sqrt{\lambda_{\max}\left(P^\top P\right)} \left(\left\| \left(P^\top P + \lambda I\right)^{-1} \right\|_2 + \left\| \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} \right\|_2 \right) \left\| \mathbf{y}_P \right\|_2 + \sqrt{\lambda_{\max}\left(Q^\top Q\right)} \left\| \left(P^\top P + Q^\top Q + \lambda I\right)^{-1} \right\|_2 \left\| \mathbf{y}_Q \right\| \\
&= \frac{\sigma_{\max}(P) \left\| \mathbf{y}_P \right\|}{\sqrt{\lambda_{\max}(P^\top P + \lambda I)}} + \frac{\sigma_{\max}(P) \left\| \mathbf{y}_P \right\|_2}{\sqrt{\lambda_{\max}\left(P^\top P + Q^\top Q + \lambda I\right)}} + \frac{\sigma_{\max}(Q) \left\| \mathbf{y}_Q \right\|_2}{\sqrt{\lambda_{\max}\left(P^\top P + Q^\top Q + \lambda I\right)}} \\
&\overset{(a)}{\leq} \frac{2\sigma_{\max}(P) \left\| P\boldsymbol{\beta}_P + \boldsymbol{\epsilon}_P \right\|_2 + \sigma_{\max} Q \left\| Q\boldsymbol{\beta}_Q + \boldsymbol{\epsilon}_Q \right\|_2}{\sqrt{\lambda_{\max}\left(P^\top P + \lambda I\right)}} \\
&\overset{(b)}{\leq} \frac{2\sigma^2_{\max}(P) \left\| \boldsymbol{\beta}_P \right\| + 6\sigma_{\max}(P) n(1 - \varepsilon^{(t)})\eta_P + \sigma^2_{\max}(Q) \left\| \boldsymbol{\beta}_Q \right\| + 3\sigma_{\max}(Q) n\varepsilon^{(t)}\eta_Q}{\sqrt{\lambda_{\max}(P^\top P) + \lambda_{\min}(\lambda I)}} \\
&\leq \frac{2\sigma^2_{\max}(P) \left\| \boldsymbol{\beta}_P \right\| + 6\sigma_{\max}(P) n(1 - \varepsilon^{(t)})\eta_P + \sigma^2_{\max}(Q) \left\| \boldsymbol{\beta}_Q \right\| + 3\sigma_{\max}(Q) n\varepsilon^{(t)}\eta_Q}{\sqrt{\sigma^2_{\max}(P) + \lambda}}
\end{aligned}
$$

(a) is due to $Q^\top Q$ being a positive semi definite symmetric matrix.
(b) holds with high probability due to the variance of the χ^2 distribution and due to Weyl's inequality

Thus we have shown that $\left\| \boldsymbol{\beta}_P - \boldsymbol{\theta}_{(t)} \right\|_2$ is bounded above at any time-step (t) in terms of the maximal singular values of the data matrices and the variance of the white noise.

This concludes the proof.

□

E.2 Derivation for Lemma 4.8

Let us note the Subquantile Matrix $S = \begin{pmatrix} P \\ Q \end{pmatrix} = \begin{pmatrix} \leftarrow & \mathbf{p}_1 & \rightarrow \\ \vdots & \vdots & \vdots \\ \leftarrow & \mathbf{p}_{\eta_P} & \rightarrow \\ \leftarrow & \mathbf{q}_1 & \rightarrow \\ \vdots & \vdots & \vdots \\ \leftarrow & \mathbf{q}_{\eta_Q} & \rightarrow \end{pmatrix}$. Thus we can define: $\varepsilon^{(t)} \triangleq \frac{\eta_Q}{\eta_P + \eta_Q} = \frac{\eta_Q}{np}$. Where n is the number of training examples and p is the Subquantile we are minimizing over. In this section, we want to provide a theoretical upper bound on η_Q, from where we can upper bound $\varepsilon^{(t)}$ which is stronger than the trivial upper bound of $\mathcal{O}(\epsilon/p)$. We will approach this problem with Order Statistics.

Assumption 5. *The optimal regressors are linearly independent, i.e.* $\boldsymbol{\beta}_P \neq \gamma \boldsymbol{\beta}_Q \, \forall \gamma \in \mathbb{R}$

Let us denote $P_1 < P_2 < \cdots < P_{n(1-\epsilon)}$ as the order statistics of the random variable $P \sim \left(\mathbf{p}_i\boldsymbol{\theta} - (\mathbf{p}_i\boldsymbol{\beta}_P + \epsilon_P)\right)^2$ where $\mathbf{p}_i \sim \mathcal{N}_d(\mathbf{0}, \xi_P I)$. Let us also denote $Q_1 < Q_2 < \cdots < Q_{n(1-\epsilon)}$ as the order statistics of the random variable $Q \sim \left(\mathbf{q}_i\boldsymbol{\theta} - (\mathbf{p}_i\boldsymbol{\beta}_Q + \epsilon_Q)\right)^2$ where $\mathbf{q}_i \sim \mathcal{N}_d(\mathbf{0}, \xi_P I)$.

First we will formalize the CDF of P and Q. Note $\mathbf{p}_i$ and $\mathbf{q}_i$ represent the normally sampled gaussian data.

$$P_i = \left(\mathbf{p}_i\boldsymbol{\theta}_{(t)} - (\mathbf{p}_i\boldsymbol{\beta}_P + \epsilon_P)\right)^2 \tag{58}$$

$$= \left(\mathbf{p}_i(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_P) - \epsilon_P\right)^2 \tag{59}$$

$$= \left(\mathbf{p}_i(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_P)\right)^2 - 2\epsilon_P\left(\mathbf{p}_i(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_P)\right) + \epsilon_P^2 \tag{60}$$

As $\mathbb{E}[\epsilon_P] = 0$, we will only consider the case $\epsilon_P = 0$. This simplifies P_i and Q_i:

$$P_i = \left(\mathbf{P}_i(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_P)\right)^2 \tag{61}$$

$$Q_i = \left(\mathbf{Q}_i(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_Q)\right)^2 \tag{62}$$

Let us note all the entries of $\mathbf{P}_i$ are sampled from $\mathcal{N}(0, \xi_P)$ and all entries of $\mathbf{Q}_i$ are sampled from $\mathcal{N}(0, \xi_Q)$. It thus follows:

$$\begin{aligned}\left(\mathbf{P}_i(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_P)\right)^2 &= \left(\sum_{j=1}^{d}\mathcal{N}(0,\xi_P)\left(\theta_j^{(t)} - \beta_{Pj}\right)\right)^2 \\ &= \left(\sum_{j=1}^{d}\mathcal{N}\left(0, \xi_P\left(\theta_j^{(t)} - \beta_{Pj}\right)^2\right)\right)^2 \\ &= \left(\mathcal{N}\left(0, \sum_{j=1}^{d}\xi_P\left(\theta_j^{(t)} - \beta_{Pj}\right)^2\right)\right)^2\end{aligned}$$

It thus similarly follows for Q_i:

$$\left(\mathbf{Q}_i(\boldsymbol{\theta}_{(t)} - \boldsymbol{\beta}_Q)\right)^2 = \left(\mathcal{N}\left(0, \sum_{j=1}^{d}\xi_Q\left(\theta_j^{(t)} - \beta_{Qj}\right)^2\right)\right)^2$$

Now we can define the cumulative distribution functions.

$$F_P(z) = \frac{1}{\sqrt{2\pi}}\int_{-\sqrt{z}}^{\sqrt{z}}\exp\left(-\frac{u^2}{2\sum_{j=1}^{d}\xi_P\left(\theta_j^{(t)} - \beta_{Pj}\right)^2}\right)du \tag{63}$$

$$= \Phi\left(\frac{\sqrt{z}}{\sqrt{\sum_{j=1}^{d}\xi_P\left(\theta_j^{(t)} - \beta_{Pj}\right)^2}}\right) - \Phi\left(\frac{-\sqrt{z}}{\sqrt{\sum_{j=1}^{d}\xi_Q\left(\theta_j^{(t)} - \beta_{Pj}\right)^2}}\right) \tag{64}$$

Let us define $\phi = \sum_{j=1}^{d}\xi_P\left(\theta_j^{(t)} - \beta_{Pj}\right)^2$ and $\psi = \sum_{j=1}^{d}\xi_Q\left(\theta_j^{(t)} - \beta_{Qj}\right)^2$ Therefore it follows:

$$F_P(z) = \Phi\left(\frac{\sqrt{z}}{\sqrt{\phi}}\right) - \Phi\left(\frac{-\sqrt{z}}{\sqrt{\phi}}\right) \tag{65}$$

Similarly for $F_Q(z)$ it follows:

$$F_Q(z) = \Phi\left(\frac{\sqrt{z}}{\sqrt{\psi}}\right) - \Phi\left(\frac{-\sqrt{z}}{\sqrt{\psi}}\right) \tag{66}$$

where Φ represents the CDF for the standard normal. Here we can note if $\phi < \psi$, then for all $z > 0$, it follows $F_P(z) > F_Q(z)$.

Let us first note $F_P(z)$ is equal to the χ^2 CDF with 1 degree of freedom. Therefore $f_P(z)$ is equal to the PDF of the χ^2 distribution 1 degree of freedom.

We will first consider a simple case, calculating the probability $\varepsilon^{(t)} = 0$, i.e., the points in the subquantile are all the points in P, and there are no points from Q within the subquantile.

To simplify notation, let $\ell \triangleq n(1-\epsilon)$ which represents number of points from P in the data matrix, X, and let $m \triangleq n(\epsilon)$ represent the number of points from Q in the data matrix, X. We define $F_{P(\ell)}$ represent the CDF of the ℓth-order statistic of P and $F_{Q(m)}$ represent the CDF of the mth-order statistic of Q. Finally, let $P_{(i)}$ represent the ith order statistic of P, in other words, it represents the ith highest error among the data points in P with respect to $\boldsymbol{\theta}$, similarly let $Q_{(j)}$ represent the jth order statistic of Q.

We will first calculate the probablity there exists 0 points from Q within the subquantile. This is equivalent to:

$$\mathbb{P}\left[\bigcap_{i=1}^{m} Q_{(i)} < P_{(\ell)}\right] = F_{Q(1)}\left(P_{(\ell)}\right) \tag{67}$$

The joint cumulative distribution function of this distribution is equivalent to:

$$\begin{aligned} H(x) &= \ell f_P(x) \left(\prod_{i=1}^{\ell-1} \mathbb{P}\left[P < x\right]\right) \left(\prod_{j=1}^{m} \mathbb{P}\left[Q > x\right]\right) \\ &= \ell f_P(x) \left(F_P(x)\right)^{\ell-1} \left(1 - F_Q(x)\right)^m \end{aligned}$$

In a future work we will provide a tight lower bound on this probability.

F Additional Experiments

F.1 `Quadratic Regression`

Objectives	Test RMSE (`Quadratic Regression`)		
	$\epsilon = 0$	$\epsilon = 0.2$	$\epsilon = 0.4$
ERM	$0.0099_{(0.0002)}$	$2.078_{(0.146)}$	$4.104_{(0.442)}$
Huber Huber & Ronchetti (2009)	$1.000_{(0.0002)}$	$1.000_{(0.0003)}$	$1.13_{(0.087)}$
RANSAC Fischler & Bolles (1981)	$0.010_{(0.0002)}$	$0.011_{(0.0002)}$	$0.061_{(0.053)}$
TERM Li et al. (2020)	$0.010_{(0.0001)}$	$0.012_{(0.0008)}$	$0.017_{(0.0016)}$
SEVER Diakonikolas et al. (2019)	$0.0166_{(0.007)}$	$0.011_{(0.0004)}$	$0.0267_{(0.036)}$
SubQuantile($p = 0.6$)	$\mathbf{0.0099_{(0.0002)}}$	$\mathbf{0.00998_{(0.0002)}}$	$\mathbf{0.010_{(0.0001)}}$
Genie ERM	$0.0099_{(0.0002)}$	$0.00997_{(0.0002)}$	$0.010_{(0.0001)}$

Table 5: `Quadratic Regression` Synthetic Dataset. Empirical Risk over $\mathbb{P}$

F.2 `Abalone`

We now provide results on `Abalone` Dataset introduced in Dua & Graff (2017). This experiment has both

Objectives	Test RMSE (`Abalone Linear Regression`)	
	Clean	Noisy
ERM	$2.213_{(0.0528)}$	$4845.335_{(117.5557)}$
CRR Bhatia et al. (2017)	$2.345_{(0.0430)}$	$396.872_{(96.5632)}$
STIR Mukhoty et al. (2019)	$2.240_{(0.0473)}$	$931.845_{(32.0864)}$
Huber Huber & Ronchetti (2009)	$5.535_{(0.0665)}$	$971.362_{(28.8863)}$
RANSAC Fischler & Bolles (1981)	$2.522_{(0.1407)}$	$2.621_{(0.1719)}$
TERM Li et al. (2020)	$10.686_{(0.2616)}$	$10.853_{(0.4245)}$
SEVER Diakonikolas et al. (2019)	$\mathbf{2.238_{(0.0901)}}$	$\mathbf{2.287_{(0.0757)}}$
SubQuantile($p = 0.8$)	$\mathbf{2.292_{(0.0413)}}$	$\mathbf{2.261_{(0.0790)}}$
Genie ERM	$2.213_{(0.0528)}$	$2.238_{(0.0901)}$

Table 6: `Abalone Regression` Real Dataset. Empirical Risk over $\mathbb{P}$

feature and label noise in the Noisy Data. SubQuantile minimization no longer always converges to the $\mathbb{P}$ SubQuantile.

F.3 `Cal-Housing`

We now provide results on `Cal-Housing` Dataset introduced in Pace & Barry (1997). This experiment has both feature and label noise in the Noisy Data.

In both the `Cal-Housing` and `Abalone` datasets there exists feature and label noise that exist with 5% probability. In this the case, the probability is low, however since the noise is very large, even having a few points from $\mathbb{Q}$ in the final subquantile matrix can largely the bias the predictions away from the optimal parameters for $\mathbb{P}$. Therefore, we reduce p, the size of the subquantile to reduce the probability of obtaining corrupted samples within the subquantile. However, what we get in a decrease in variance, we do increase the bias error, albeit very slightly.

Objectives	Test RMSE (`Cal-Housing Linear Regression`)	
	Clean	Noisy
ERM	$0.598_{(0.0077)}$	$81.758_{(2.6230)}$
CRR Bhatia et al. (2017)	$\mathbf{0.602_{(0.0081)}}$	$75.777_{(2.9403)}$
STIR Mukhoty et al. (2019)	$\mathbf{0.604_{(0.0070)}}$	$65.555_{(2.1899)}$
Huber Huber & Ronchetti (2009)	$\mathbf{0.601_{(0.0077)}}$	$71.813_{(2.0755)}$
RANSAC Fischler & Bolles (1981)	$0.681_{(0.0389)}$	$0.679_{(0.0253)}$
TERM Li et al. (2020)	$0.737_{(0.0070)}$	$0.625_{(0.0083)}$
SEVER Diakonikolas et al. (2019)	$0.640_{(0.0067)}$	$0.642_{(0.0088)}$
SubQuantile($p = 0.9$)	$\mathbf{0.615_{(0.0076)}}$	$\mathbf{0.612_{(0.0096)}}$
Genie ERM	$0.598_{(0.0077)}$	$0.603_{(0.0068)}$

Table 7: `Cal-Housing Regression` Real Dataset. Empirical Risk over $\mathbb{P}$

F.4 `Logistic Regression`

We will describe the inliers and outliers:
Inliers: $x_{\text{in}} \sim \mathcal{N}(0, I_2)$, $\mathbf{w} \sim \mathcal{N}(0, I_2)$, $b \sim \mathcal{N}(0, 1)$, $y|\mathbf{x} \sim \text{sign}\left(\mathcal{N}(\mathbf{x}^\top \mathbf{w} + b, 0.01)\right)$
Outliers: $x_{\text{out}} \sim \mathcal{U}\left([-10, 10]^2\right)$, $y|\mathbf{x} \sim \text{sign}\left(-\mathbf{x}^\top \mathbf{w} - b\right)$
$x_{\text{train}} = 800$, $X_{\text{test}} = 200$

F.5 Support Vector Machine (SVM)

Objectives	Test Accuracy (`Support Vector Machine`)			
	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
ERM	$0.728_{(0.0231)}$	$0.607_{(0.0292)}$	$0.495_{(0.0331)}$	$0.393_{(0.0864)}$
TERM Li et al. (2020)	$0.640_{(0.0067)}$	$0.642_{(0.0088)}$	$0.549_{(0.0263)}$	$0.041_{(0.0267)}$
SEVER Diakonikolas et al. (2019)	$0.856_{(0.1060)}$	$\mathbf{0.808_{(0.1206)}}$	$0.808_{(0.1563)}$	$\mathbf{0.843_{(0.1452)}}$
Subquantile (Ours)	$\mathbf{0.919_{(0.0909)}}$	$0.728_{(0.1434)}$	$\mathbf{0.825_{(0.1643)}}$	$0.739_{(0.1750)}$
Genie ERM	$1.000_{(0.000)}$	$1.000_{(0.000)}$	$1.000_{(0.000)}$	$1.000_{(0.000)}$

Table 8: `SVM Synthetic`. Test accuracy on P. Subquantile is trained with $p = (1-\epsilon)$ and max 32 iterations. SEVER is trained with 4 iterations and $p = 0.3$.

Objectives	Test Accuracy (`Enron Spam Classification`)			
	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
ERM	$0.751_{(0.0156)}$	$0.633_{(0.0246)}$	$0.524_{(0.0291)}$	$0.496_{(0.0608)}$
TERM Li et al. (2020)	∞	∞	∞	∞
SEVER Diakonikolas et al. (2019)	$0.795_{(0.0167)}$	$0.649_{(0.0276)}$	$0.567_{(0.0185)}$	$0.528_{(0.0305)}$
Subquantile (Ours)	$\mathbf{0.892_{(0.0164)}}$	$\mathbf{0.815_{(0.0204)}}$	$\mathbf{0.750_{(0.0460)}}$	$\mathbf{0.653_{(0.0490)}}$
Genie ERM	$0.963_{(0.0065)}$	$0.963_{(0.0049)}$	$0.960_{(0.0072)}$	$0.963_{(0.0060)}$

Table 9: Test accuracy on P. Subquantile is trained with $p = (1 - \epsilon)$ and max 32 iterations. SEVER is trained with 8 iterations and $p = 0.1$. Noisy labels are flipped from -1 to 1 or -1 to 1.

Objectives	Test Accuracy (`Adult Dataset`)			
	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
ERM	$0.838_{(0.0029)}$	$0.785_{(0.0040)}$	$0.781_{(0.0040)}$	$0.776_{(0.0033)}$
TERM Li et al. (2020)	∞	∞	∞	∞
SEVER Diakonikolas et al. (2019)	$0.691_{(0.0545)}$	$0.681_{(0.0586)}$	$0.587_{(0.0917)}$	$0.551_{(0.1020)}$
Subquantile (Ours)	$\mathbf{0.847_{(0.0028)}}$	$\mathbf{0.841_{(0.0185)}}$	$\mathbf{0.842_{(0.0039)}}$	$\mathbf{0.811_{(0.0040)}}$
Genie ERM	$0.847_{(0.0028)}$	$0.847_{(0.0033)}$	$0.848_{(0.0032)}$	$0.848_{(0.0033)}$

Table 10: Test accuracy on P. Subquantile is trained with $p = (1 - \epsilon)$ and max 32 iterations. SEVER is trained with 8 iterations and $p = 0.1$. Noisy labels are flipped from -1 to 1 or -1 to 1.

G Experimental Details

G.1 `Adaptive Linear Regression` Dataset

We will describe $\mathbb{P}$ and $\mathbb{Q}$ in the `Structured Linear Regression` Dataset.
$\mathbf{x} \sim \mathcal{N}(4, 4)^{200}$
$\mathbf{w} \sim \mathcal{N}(4, 4)^{200}$
$b \sim \mathcal{N}(4, 4)$
$\mathbf{w}' \sim \mathcal{N}(4, 4)^{200}$
$b' \sim \mathcal{N}(4, 4)$
$n_{\text{train}} = 1\text{e}4$
$\mathbb{P} : y|\mathbf{x} \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w} + b, 0.1)$
$\mathbb{Q} : y|\mathbf{x} \sim \mathcal{N}(\mathbf{x}'^\top \mathbf{w} + b', 0.1)$
Please note $\mathbf{w}$, b, $\mathbf{w}'$, b', are all sampled independently. The noise is added after normalization of the dataset to the standard normal $\mathcal{N}(0, 1)$.

G.2 `Oblivious Linear Regression` Dataset

We will describe $\mathbb{P}$ and $\mathbb{Q}$ in the `Noisy Linear Regression` Dataset.
$\mathbf{x} \sim \mathcal{N}(0, 3)^{500}$
$\mathbf{w} \sim \mathcal{N}(4, 4)^{500}$
$b \sim \mathcal{N}(4, 4)$
$\mathbf{w}' = \mathbf{0}$
$b' \sim \mathcal{N}(5, 5)$
$n_{\text{train}} = 8\text{e}3$
$n_{\text{test}} = 2\text{e}3$
$\mathbb{P} : y|\mathbf{x} \sim \mathcal{N}(\mathbf{x}^\top \mathbf{w} + b, 0.01)$
$\mathbb{Q} : y|\mathbf{x} \sim \mathcal{N}(5, 5)$
Please note $\mathbf{w}$, b, $\mathbf{w}'$, b', are all sampled independently. The noise is added after normalization of the dataset to the standard normal.

G.3 `Quadratic Regression` Dataset

We will describe $\mathbb{P}$ and $\mathbb{Q}$ in the `Quadratic Regression` dataset.
$x \sim \mathcal{N}(0, 1)$
$n_{\text{train}} = 1\text{e}4$
$\mathbb{P} : y|x \sim \mathcal{N}(x^2 - x + 2, 0.01)$
$\mathbb{Q} : y|x \sim \mathcal{N}(-x^2 + x + 4, 0.01)$

G.4 `Drug Discovery` Dataset

This dataset is downloaded from Diakonikolas et al. (2019). We utilize the same noise procedure as in Li et al. (2020).
$\mathbb{P}$ is given from an 80/20 train test split from the dataset.
$\mathbb{Q}$ is random noise sampled from $\mathcal{N}(5, 5)$.
The noise represents a noisy worker

G.5 Feature Noise

Take 5% of the training data and multiply features by 100 and responses by 10000.