
ROBUST LINEAR REGRESSION BY SUB-QUANTILE OPTIMIZATION

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

ABSTRACT

Robust Linear Regression is the problem of fitting data to a distribution, \mathbb{P} when there exists contaminated samples, \mathbb{Q} . We model this as $\hat{\mathbb{P}} = (1 - \varepsilon)\mathbb{P} + \varepsilon\mathbb{Q}$. Traditional Least Squares Methods fit the empirical risk model to all training data in $\hat{\mathbb{P}}$. In this paper we show theoretical and experimental results of sub-quantile optimization, where we optimize with respect to the p -quantile of the empirical loss.

1 INTRODUCTION

Linear Regression is one of the most widely used statistical estimators throughout science. Although robustness is only a somewhat recent topic in machine learning, it has been a topic in statistics for many decades. 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 by ? in predicting death rates (per million people) of a country given its cigarette consumption per capita twenty years prior. When numerous outliers have considerably high leverage on a data set, they can vastly skew a model's predictions if said outliers are not accounted for. When utilizing more robust techniques, including MM-estimation or robust and efficient weighted least squares estimator (REWLSE), the data can be represented more accurately without being overwhelmed by outlying data.

One of the most prominent cases in which robust linear regression serves as a sufficiently resilient method of prediction is in ? analyzing economic growth, or the rise in gross domestic product (GDP) of a nation. The econometric application of robust regression techniques serves to identify how various robust regression techniques perform at assessing how the rapid population growth seen in India and Pakistan affect the economic growth of said countries. The outliers seen in the data sets analyzed by Khan et al. affect both independent and dependent variables, thus having a robust model that is resilient to the sheer number of outliers is crucial to produce accurate results.

The value and necessity of robust linear regression can also be demonstrated in meteorology and climatology by ?. In analysing temperature and precipitation time-series data collected in Switzerland, classical regression techniques (namely least-squares regression) were compared to two robust regression techniques: least median of squares (LMS) and the aforementioned LTS estimators. In comparing the two techniques, it was apparent that classical regression techniques can be consequentially sensitive to outliers, while the rudimentary robust regression models proved to be resilient to outliers. Furthermore, with the ever-increasing likelihood of outlying meteorological observations

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

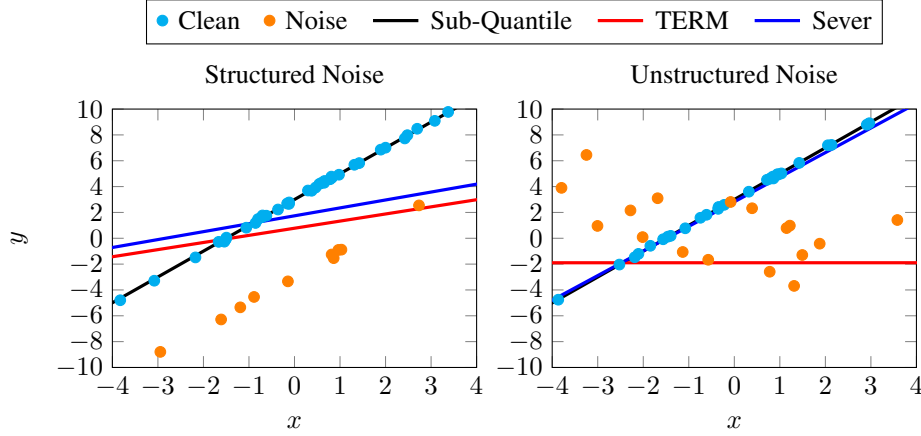


Figure 1: Sub-Quantile Performance on Linearly Dependent and Independent Noise

(heat waves, heavy precipitation, etc.), it is imperative that models be robust enough to quantify an accurate trend that accounts for the existence of outlier-corrupted data.

1.2 CONTRIBUTIONS

Our goal is to provide a theoretic analysis and convergence conditions for sub-quantile optimization and offer practitioners a method for robust linear regression. Several popular methods have been utilized due to their simplicity and high effectiveness including quantile regression ?, Theil-Sen Estimator ?, and Huber Regression ?. These methods, although rudimentary, serve to show the effectiveness of building resistance against outliers in data. By improving upon existing methods, namely least-squares estimation in these cases, models can be designed to better estimate data sets with considerably corruptive outliers.

Sub-Quantile Optimization aims to address the shortcomings of ERM in applications such as noisy/corrupted data (?,?), classification with imbalanced classes (?,?), as well as fair learning (?).

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, sub-quantile optimization is shown to prevail at overcoming these challenges current models currently face.

2 RELATED WORK

Least Trimmed Squares (LTS) ? is an estimator that relies on minimizing the sum of the smallest h residuals given a $(d - 1)$ -dimension hyperplane calculated given n data points in \mathbf{R}^d and an integer trimming parameter h . Given that the outliers comprise less than half the data, this algorithm is more efficient than the more common LMS estimator. However, this algorithm unfortunately suffers from the curse of dimensionality; the computational cost of the algorithm grows exponentially with increasing dimensions of the data. Thus, the necessity to design a more computationally efficient algorithm is expressed.

Tilted Empirical Risk Minimization (TERM) ? is a framework built to similarly handle the shortcomings of empirical risk minimization (ERM) with respect to robustness. The TERM framework instead minimizes the following quantity, where t is a hyperparameter

$$\tilde{R}(t; \theta) := \frac{1}{t} \log \left(\frac{1}{N} \sum_{i \in [N]} e^{t f(\mathbf{x}_i; \theta)} \right) \quad (1)$$

SMART ? proposes 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.

$$\hat{\theta}(S) = \min_{\theta} \min_{\hat{S} \subset S, |\hat{S}|=(1-\epsilon)n} \sum_{(\mathbf{x}_i, y_i) \in \hat{S}} -\log f(y_i | \theta^\top \mathbf{x}_i) \quad (2)$$

SEVER ? is a gradient filtering algorithm which removes elements whose gradients have the furthest distance from the average gradient of all points

$$\tau_i = \left((\nabla f_i(\mathbf{w}) - \hat{\nabla}) \cdot \mathbf{v} \right)^2 \quad (3)$$

Super-Quantile Optimization ? ?

Robust Risk Minimization ?

Quantile Regression ?

3 SUB-QUANTILE 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

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

Note $Q_p(0.5)$ represents the median of the random variable.

Definition 2. The **Empirical Distribution Function** is defined as follows

$$\hat{F}_n(t) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{X_i \leq t} \quad (5)$$

Definition 3. Let ℓ be the loss function. **Risk** is defined as follows

$$U = \mathbb{E}[\ell(f(\mathbf{x}; \theta), \mathbf{y})] \quad (6)$$

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

$$\mathbb{L}_p(U) = \frac{1}{p} \int_0^p Q_q(U) dq = \mathbb{E}[U | U \leq Q_p(U)] = \max_{t \in \mathbb{R}} \left\{ t - \frac{1}{p} \mathbb{E}[(t - U)^+] \right\} \quad (7)$$

In equation 7, t represents the p -quantile of U . We also show that we can calculate t by a maximizing optimization function. The Sub-Quantile Optimization problem is posed as follows

$$\theta_{SM} = \arg \min_{\theta \in \mathbb{R}^d} \max_{t \in \mathbb{R}} \left\{ t - \frac{1}{p} \mathbb{E}(t - \ell(f(\mathbf{x}; \theta), \mathbf{y}))^+ \right\} \quad (8)$$

For the linear regression case, this equation becomes

$$\theta_{SM} = \arg \min_{\theta \in \mathbb{R}^d} \max_{t \in \mathbb{R}} \left\{ t - \frac{1}{np} \sum_{i=1}^n (t - (\theta^\top \mathbf{x}_i - y_i)^2)^+ \right\} \quad (9)$$

The two-step optimization for Sub-Quantile optimization is given as follows

$$t_{k+1} = \arg \max_t g(t, \theta_k) \quad (10)$$

$$\theta_{k+1} = \theta_k + \alpha \nabla_{\theta_k} g(t, \theta_k) \quad (11)$$

This algorithm is adopted from ?. Theoretically, it has been proven to converge in research by ?.

3.1 MOTIVATION

Assumption 1. To provide theoretical bounds on the effectiveness of Sub-Quantile Minimization, we make the General Linear Model Assumption that

$$\mathbf{y}_P = \mathbf{P}\beta_P + \epsilon_P \quad (12)$$

and similarly

$$\mathbf{y}_Q = \mathbf{Q}\beta_Q + \epsilon_Q \quad (13)$$

where β_P and β_Q the oracle regressors for \mathbb{P} and \mathbb{Q} and ϵ_P and ϵ_Q are both Normally Distributed with mean 0.

Since we are interested in learning the optimal model for distributions, our goal is to learn the parameters β_P from the distribution $\hat{\mathbb{P}}$. We want to clarify the corruption is not adversarially chosen. In this section we quantify the effect of corruption on the desired model. To introduce notation, let \mathbf{P} represent the data from distribution \mathbb{P} and let \mathbf{Q} represent the training data for \mathbb{Q} . Let \mathbf{y}_P represent the target data for \mathbb{P} and let \mathbf{y}_Q represent the target data for \mathbb{Q} .

Assumption 2. *We assume the rows of \mathbf{P} and \mathbf{Q} are sampled from the same multivariate normal distribution.*

$$\mathbf{P}_i, \mathbf{Q}_j \sim \mathcal{N}_p(\mathbf{0}, \Sigma) \quad (14)$$

We will use our assumptions to quantify the effect of the corrupted data on an optimal least squares regression model. We are interested in $(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y} - (\mathbf{P}^\top \mathbf{P})^{-1} \mathbf{P}^\top \mathbf{y}$. It is known the least squares optimal solution for \mathbf{X} is equal to $(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$

Note $\mathbf{X} = \begin{pmatrix} \mathbf{P} \\ \mathbf{Q} \end{pmatrix}$ and $\mathbf{y} = \begin{pmatrix} \mathbf{y}_P \\ \mathbf{y}_Q \end{pmatrix}$ so $\mathbf{X}^\top = (\mathbf{P}^\top \quad \mathbf{Q}^\top)$

Theorem 1. *The expected optimal parameters of the corrupted model $\hat{\mathbb{P}}$*

$$\mathbb{E} [\mathbf{X}^\dagger \mathbf{y}] = \beta_P + \epsilon(\beta_Q - \beta_P) \quad (15)$$

The proof is reliant on assumption 2, this allows us to utilize the Wishart Distribution, \mathcal{W} , and the inverse Wishart Distribution, \mathcal{W}^{-1} . Please refer to Appendix A.1. By Theorem 1 we can see the level of corruption is dependent upon ϵ , which represents the percentage of corrupted samples, and the distance between the optimal parameters for \mathbb{P} , which is β_P and the optimal parameters for \mathbb{Q} , which is β_Q .

Here we utilize the idea of *influence* from ?.

Theorem 1 finds the optimal model when the corrupted distribution is sampled from the same distribution as the target distribution but has different optimal parameters. We will now look at the case of feature corruption. This is where the optimal parameters of the two distributions are the same but the data from \mathbb{P} and \mathbb{Q} are sampled differently.

Theorem 2. *In the case of \mathbb{P} and \mathbb{Q} being from different Normal Distributions. The expected optimal parameters of the corrupted model $\hat{\mathbb{P}}$*

$$\mathbb{E} [\mathbf{X}^\dagger \mathbf{y}] = \beta_P - n(1 - \epsilon)\Sigma_P^{-1}\beta_P \quad (16)$$

The proof can be found in Appendix A.2. We will show our results hold in Numerical Experiments. As seen in the results in table 1, the theory we provide is supported by Numerical Experimentation.

Dataset	$\epsilon = 0.2$		$\epsilon = 0.4$	
	$\mathbb{E} [\mathbf{X}^\dagger \mathbf{y}]$	Experimental	$\mathbb{E} [\mathbf{X}^\dagger \mathbf{y}]$	Experimental
Quadratic Regression	(0.6, -0.6, 2.4)	0.777 (0.007)	(0.2, -0.2, 2.8)	7.749 (0.009)
Drug Discovery	0.895(0.009)	0.775(0.006)	8.944(0.007)	7.742(0.006)

Table 1: Verification of Theorem 1 over Quadratic Regression Synthetic Dataset

In equation 15, note as $\epsilon \rightarrow 0$ we are returned β_P . This is the intuition behind SubQuantile Minimization. By minimizing over the SubQuantile, we seek to reduce ϵ , and thus our model will return a model which is by expectation closer to β_P .

4 THEORY

4.1 ANALYSIS OF $g(t, \theta)$

In this section, we will explore the fundamental aspects of $g(t, \theta)$. This will motivate the convergence analysis in the next section.

Lemma 4.1. $g(t_{k+1}, \theta_k)$ 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 . \square

Lemma 4.2. The maximizing value of t in $g(t, \theta)$ in t -update step of optimization as described by Equation 10 is maximized when $t = Q_p(U)$

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

$$\partial(-g(t, \theta)) = -1 + \frac{1}{np} \sum_{i=1}^n \begin{cases} 1, & \text{if } t > \hat{\nu}_i \\ 0, & \text{if } t < \hat{\nu}_i \\ [0, 1], & \text{if } t = \hat{\nu}_i \end{cases} \quad (17)$$

$$= 0 \text{ when } t = \hat{\nu}_{np} \quad (18)$$

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

Lemma 4.3. Let $t = \hat{\nu}_{np}$. The θ -update step described in Equation 9 is equivalent to minimizing the least squares loss of the np elements with the lowest squared loss.

$$\nabla_{\theta} g(t_{k+1}, \theta_k) = \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i(\theta_k^\top \mathbf{x}_i - y_i) \quad (19)$$

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

Interpretation 1. Sub-Quantile Minimization continuously 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. $g(t_{k+1}, \theta_k)$ is convex with respect to θ_k .

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

$$g(t_{k+1}, \theta_k) = t_{k+1} - \frac{1}{np} \sum_{i=1}^n (t_{k+1} - (\theta^\top \mathbf{x}_i - y_i)^2)^+ \quad (20)$$

$$= t_{k+1} - \frac{1}{np} \sum_{i=1}^{np} (t_{k+1} - (\theta^\top \mathbf{x}_i - y_i)^2)^+ \quad (21)$$

$$= t - t + \frac{1}{np} \sum_{i=1}^{np} (\theta^\top \mathbf{x}_i - y_i)^2 \quad (22)$$

$$= \frac{1}{np} \sum_{i=1}^{np} (\theta^\top \mathbf{x}_i - y_i)^2 \quad (23)$$

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 $g(t, \theta)$ is convex with respect to θ . \square

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

4.2 OPTIMIZATION

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

Definition 4. (t^*, θ^*) is a **Nash Equilibrium** of g if for any $(t, \theta) \in \mathbb{R} \times \mathbb{R}^d$

$$g(t^*, \theta) \leq g(t^*, \theta^*) \leq g(t, \theta^*) \quad (24)$$

Definition 5. (t^*, θ^*) is a **Local Nash Equilibrium** of g if there exists $\delta > 0$ such that for any t, θ satisfying $\|t - t^*\| \leq \delta$ and $\|\theta - \theta^*\| \leq \delta$ then:

$$g(t^*, \theta) \leq g(t^*, \theta^*) \leq g(t, \theta^*) \quad (25)$$

Proposition 1. As g is first-order differentiable, any local Nash Equilibrium satisfies $\nabla_{\theta} g(t, \theta) = \mathbf{0}$ and $\nabla_t g(t, \theta) = 0$

We are now interested in what it means to be at a Local Nash Equilibrium. By Proposition 1, this means both first-order partial derivatives are equal to 0. By lemma 4.2, we have shown $\nabla_t g(t, \theta) = 0$ when $\nu_{np} \leq t < \nu_{np+1}$. Furthermore, by lemma 4.3, we have shown $\nabla_{\theta}(g, \theta) = 0$ when the least squares error is minimized for the np points with lowest squared error. In other words:

$$\mathbb{E} [\nabla_{\theta} g(t_{k+1}, \theta_k)] = 0$$

$$2(\mu\mu^T + \Sigma)(\theta_k - (1 - \varepsilon)\beta_P - \varepsilon\beta_Q) = 0$$

Since the first term is non-zero, the equality is satisfied when:

$$(\theta_k - (1 - \varepsilon)\beta_P - \varepsilon\beta_Q) = 0$$

$$\theta_k = (1 - \varepsilon)\beta_P + \varepsilon\beta_Q$$

Note this aligns with the results of Theorem 1. This means that for a subset of np points from \mathbf{X} , the least squares error is minimized. What we are interested in is how many points within those np points come from \mathbb{P} and how many of those points from \mathbb{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 .

Lemma 4.6. If $t_{k+1} \leq t_k$ then $g(t_{k+1}, \theta_k) = g(t_k) + \frac{1}{np} \sum_{i=np}^n (t_k - \nu_i)^+$. If $t_{k+1} > t_k$, then $g(t_{k+1}, \theta_k) = g(t_k) + \frac{1}{np} \sum_{i=n(p-\delta)}^{np} (t - \nu_i)^+ - \delta t$. For a small δ . *This is just a general property, unsure if worth keeping.*

Proof Sketch. When $t_{k+1} \leq t_k$ this result is quite intuitive, as we are simply removing the error of the elements outside elements within the lowest np squared losses. We delegate the rest of the proof to Appendix B.4 \square

Lemma 4.7. We will utilize lemmas 4.3 and 4.5 to quantify the change of g after the theta-update.

Proof Sketch. We are specifically interested in the case when g increases after a theta-update. \square

4.3 CONVERGING TO β_P

We will start by defining the two types of noise we are interesting in.

Definition 6. **Unstructured Noise** is noise that is not dependent on the input data, i.e., $\mathbb{P}[y|\mathbf{X}] = \mathbb{P}[\mathbf{X}]$

Definition 7. **Linearly Structured Noise** is noise that is made from a linear combination of the input data, i.e. $y = \beta_Q \mathbf{X} + \epsilon$

Also note we often consider Gaussian Noise as Unstructured Noise, but it can be modeled as Structured Noise where $\beta_Q = \mathbf{0}$.

Theorem 3. The Expected Value of Epsilon and the p -Quantile can be calculated as:

$$\mathbb{E}[\varepsilon] = \frac{\mathbb{E}[P^+]}{np} \quad (26)$$

$$Q_p = \dots \quad (27)$$

We provide a proof in Appendix D.1. Furthermore, we validate our theoretical estimation of the number of elements of \mathbb{P} less than Q_p as well as our estimation of Q_p in figures ... & Our strongest theoretical results for convergence come in the case of *Linearly Structured Noise*.

Theorem 4. *The convergence of SubQuantile Optimization is dependent on the initial weights θ_0 in the case of linearly structured noise.*

Proof Sketch. Let us consider a worst-case scenario where $\theta_k = \beta_Q$. Note this is not an impossible situation as if you naively choose $\theta_0 = \mathbf{0}$ and $\mathbb{Q} \sim \mathcal{N}(\mathbf{0}, \Sigma_Q)$. \square

Theorem 4 tells us we can not randomly select weights and expect to converge to the majority class. We must choose the initial weights in a specific way as such so we can guarantee the majority elements by expectation have more representation within the lowest np squared losses after the loss iteration.

Theorem 5. *If we choose $\theta_0 = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$ then SubQuantile optimization converges to a local minima by expectation. This should likely be a corollary.*

Proof Sketch. By Theorem 1, $\theta_0 = (1 - \epsilon)\beta_P + \epsilon\beta_Q$ \square

We are now interested in theoretical guarantees with no distributional assumptions. First we will consider some intuition on why this problem is not as hard as compared to when the corruption is linearly structured. Let us say the noise is of non-linear regression, in other words $\mathbf{y}_Q \sim f(\mathbf{X}, \beta_Q)$, where f is a non-linear combination of features of \mathbf{X} . In this case, it is not possible to model the non-linear regression by a linear combination of the features, thus, if we have elements from \mathbb{Q} within the lowest np losses, then training on these points will not generalize well to points from \mathbb{Q} , so their error will not decrease.

4.4 COMPLEXITY OF SUBQUANTILE OPTIMIZATION

In this section we will provide the expected complexity in the case of linearly structured noise.

5 EMPIRICAL RESULTS

The first experiment we will run will display the difference of the following two t updates

$$t_{k+1} = \hat{\nu}_{np} \tag{28}$$

$$t_{k+1} = \frac{1}{np} \sum_{i=1}^{np} \hat{\nu}_i \tag{29}$$

In general, if the $\hat{\nu}_1, \hat{\nu}_2, \dots, \hat{\nu}_{np}$ are closely distributed, then $\frac{1}{np} \sum_{i=1}^{np} \hat{\nu}_i \approx \hat{\nu}_{np}$. In Algorithm 1, we display our training method for Sub-quantile Optimization with the t update as described in equation 28. We also compare against the t -update as described in equation 29.

Algorithm 1: Sub-Quantile Minimization Optimization Algorithm

Input: Training iterations T , Quantile p , Corruption Percentage ϵ , Input Parameters m

Output: Trained Parameters, θ

Data: Inliers: $y|x \sim \mathcal{N}(x^2 - x + 2, 0.01)$, Outliers: $y|x \sim \mathcal{N}(-x^2 + x + 4, 0.01)$

```
1:  $\theta_0 \leftarrow \frac{2}{L} \sum_{i=1}^n (x_i y_i)$ 
2: for  $k \in 1, 2, \dots, m$  do
3:    $\nu = (X\theta_k - y)^2$ 
4:    $\hat{\nu} = \text{sorted}(\nu)$ 
5:    $t_{k+1} = \hat{\nu}_{np}$ 
6:    $t_{k+1} = \frac{1}{np} \sum_{i=1}^{np} \nu_i$ 
7:    $L := \sum_{i=1}^{np} x_i^\top x_i$ 
8:    $\alpha := \frac{1}{2L}$ 
9:    $\theta_{k+1} = \theta_k - \alpha \nabla_{\theta_k} g(t_{k+1}, \theta_k)$ 
10: end
11: if  $\epsilon > 0.5$  then
12:    $P = \hat{\nu} [np : ]$ 
13:    $y_P = y [np : ]$ 
14:    $\theta_T = (P^\top P)^{-1} P^\top y_P$ 
15: end
16: return  $\theta_T$ 
```

We also present a batch algorithm which improves training speed significantly. In accordance with Minibatch theory, if the subset I of all data is representative of all the data, then this will have similar results to Algorithm 1.

5.1 SYNTHETIC DATA

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

From the results we can see in Figure ??, Subquantile Minimization performs better throughout all noise ranges. The one struggle exists when ϵ is around 0.5, thus we face issues similar to the power method where there exists the top two eigenvalues such that $|\lambda_1| \approx |\lambda_2|$.

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 \mathbf{X} . The results of Sub-Quantile Minimization can be seen in Figure 2. Our results show the near optimal performance of Sub-Quantile Minimization. The results and comparison with other methods can be seen in Table 2. Note we are not interested in $\epsilon \geq 0.5$ as the concept of corruptness becomes unclear. We see in Table 2, Sub-Quantile 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 F for more details on the Structured Linear Regression Dataset.

In our second synthetic experiment, we run Algorithm 1 similarly on synthetically generated linear regression data. However, in this experiment, the noise is sampled from a Gaussian that is independent of the \mathbf{X} coordinates.

Methods such as TERM, ?, are unable to capture the target distribution through structurally generated noise, which can also be called *adversarial*. SubQuantile Optimization, on the other hand, is robust to such adversarial attacks.

5.2 REAL DATA

We provide results on the Drug Discovery Dataset in ? utilizing the noise procedure described in ?.

As we can see in Table 3, we obtain state of the art results in the lower range of range of noise, and futher more, we obtain results on par with the current state of the art. This makes our model the

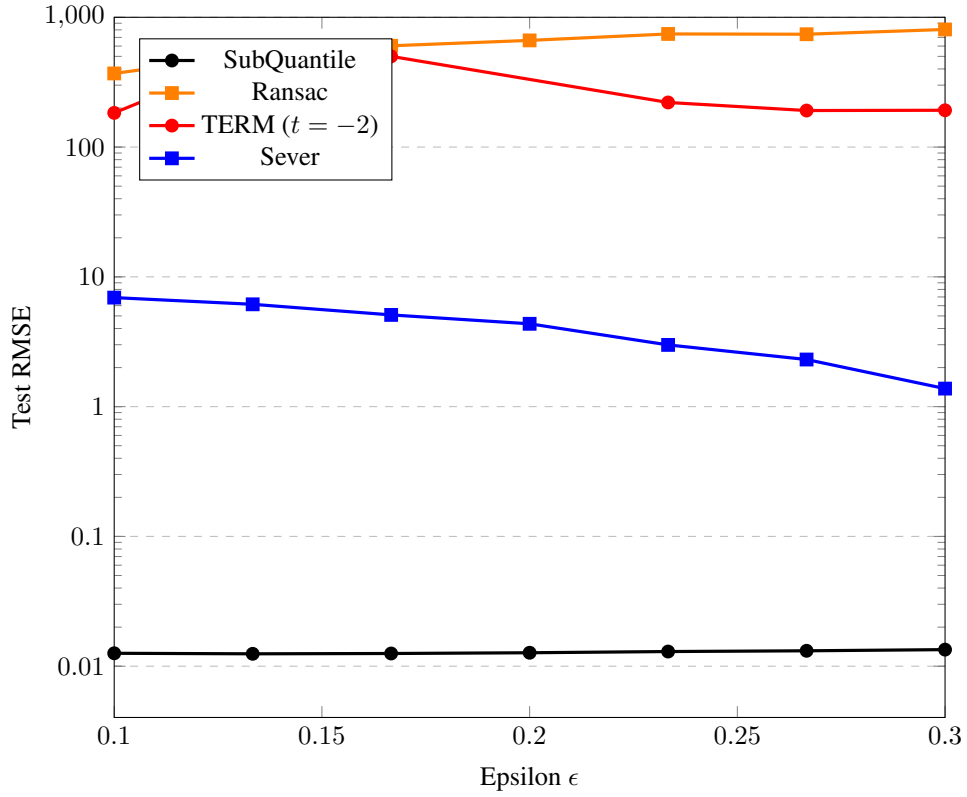


Figure 2: Accuracy of Structured Linear Regression Dataset

Objectives	Test RMSE (Quadratic Regression)		
	$\epsilon = 0$	$\epsilon = 0.2$	$\epsilon = 0.4$
OLS 124	0.0099 _(0.0002)	2.078 _(0.146)	4.104 _(0.442)
Huber ?	1.000 _(0.0002)	1.000 _(0.0003)	1.13 _(0.087)
RANSAC ?	0.010 _(0.0002)	0.011 _(0.0002)	0.061 _(0.053)
TERM ?	0.010 _(0.0001)	0.012 _(0.0008)	0.017 _(0.0016)
SEVER ?	0.0166 _(0.007)	0.011 _(0.0004)	0.0267 _(0.036)
SubQuantile($p = 0.6$)	0.0099_(0.0002)	0.00998_(0.0002)	0.010_(0.0001)
Genie ERM	0.0099 _(0.0002)	0.00997 _(0.0002)	0.010 _(0.0001)

Table 2: Qudatic Regression Synthetic Dataset. Empirical Risk over \mathbb{P}

strongest among the tested, due to our strength throughout the whole range of noises. This dataset is also

6 CONCLUSION

In this work we provide a theoretical analysis for robust linear regression by minimizing the *Sub-Quantile* of the Empirical Risk. Furthermore, we run various numerical experiments and compare against the current State of the Art in Robust Linear Regression.

AUTHOR CONTRIBUTIONS

ACKNOWLEDGMENTS

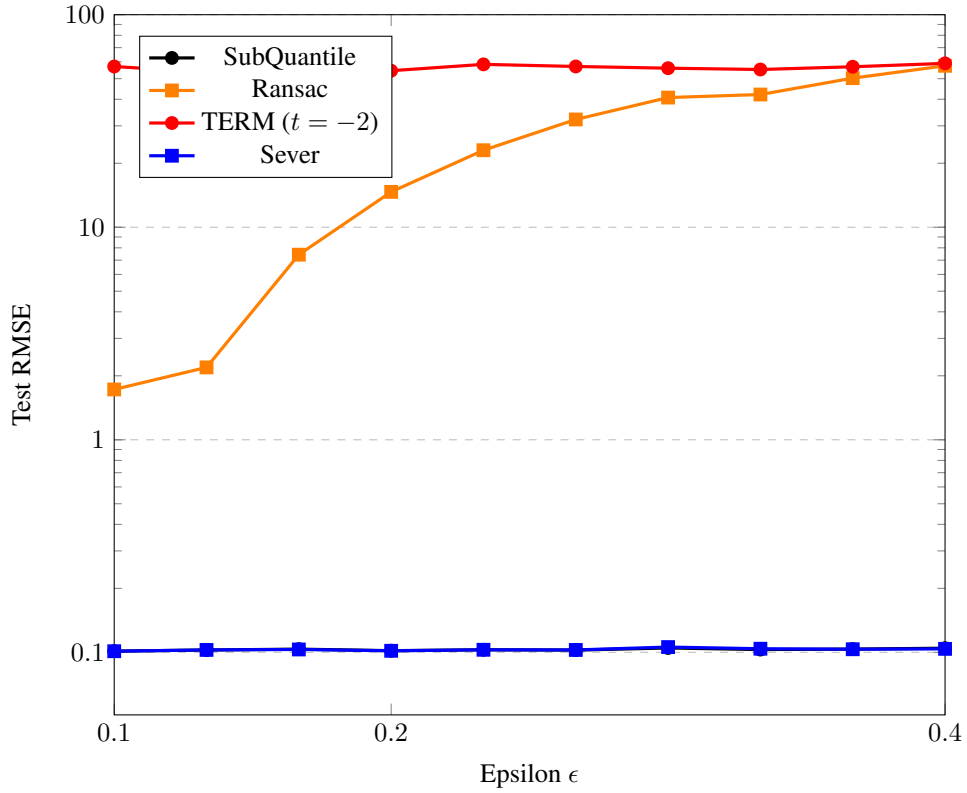


Figure 3: Accuracy of Noisy Linear Regression Dataset

Objectives	Test RMSE (Drug Discovery)			
	$\epsilon = 0$	$\epsilon = 0.2$	$\epsilon = 0.4$	$\epsilon = 0.8$
OLS 124	0.990 _(0.060)	1.969 _(0.118)	2.829 _(0.086)	4.682 _(0.101)
Huber ?	1.326 _(0.096)	1.628 _(0.253)	2.023 _(0.498)	3.442 _(0.581)
RANSAC ?	∞	∞	∞	∞
TERM ?	1.313 _(0.072)	1.334 _(0.105)	1.343 _(0.0740)	1.428 _(0.107)
SEVER ?	1.079 _(0.059)	1.076 _(0.048)	1.067 _(0.091)	3.993 _(0.203)
SubQuantile($p = 1 - \epsilon$)	1.052 _(0.062)	1.060 _(0.065)	1.073 _(0.101)	1.479 _(0.0695)
Genie ERM	0.990 _(0.060)	1.038 _(0.041)	1.037 _(0.086)	∞

Table 3: Drug Discovery Dataset. Empirical Risk over \mathbb{P}

A	Proofs on the effect of Corruption	12
A.1	Proof of Theorem 1	12
A.2	Proof of Theorem 2	13
B	General Properties of Sub-Quantile Minimization	14
B.1	Proof of Lemma 4.2	14
B.2	Proof of Lemma 4.3	14
B.3	Proof of Lemma 4.5	15
B.4	Proof of Lemma 4.6	15

C	Stochastic Sub-Quantile Optimization	17
D	Theory for Linear Corruption	18
D.1	Proof of Theorem 3	18
D.2	Proof of Theorem 4	19
D.3	Proof of Theorem 5	21
E	Additional Experiments	22
F	Experimental Details	23
F.1	Structured Linear Regression Dataset	23
F.2	Noisy Linear Regression Dataset	23
F.3	Quadratic Regression Dataset	23
F.4	Drug Discovery Dataset	23
F.5	Baseline Methods in Section 5	23

A PROOFS ON THE EFFECT OF CORRUPTION

A.1 PROOF OF THEOREM 1

Proof.

We will first calculate the pseudo-inverse

$$\mathbf{X}^\top \mathbf{X} = (\mathbf{P}^\top \quad \mathbf{Q}^\top) \begin{pmatrix} \mathbf{P} \\ \mathbf{Q} \end{pmatrix} \quad (30)$$

$$= \mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q} \quad (31)$$

Now we can calculate the Moore-Penrose Inverse

$$(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top = (\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1} (\mathbf{P}^\top \quad \mathbf{Q}^\top) \quad (32)$$

$$= ((\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1} \mathbf{P}^\top \quad (\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1} \mathbf{Q}^\top) \quad (33)$$

Now we solve for the optimal model

$$\mathbf{X}^\dagger \mathbf{y} = ((\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1} \mathbf{P}^\top \quad (\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1} \mathbf{Q}^\top) \begin{pmatrix} \mathbf{y}_P \\ \mathbf{y}_Q \end{pmatrix} \quad (34)$$

$$= (\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1} \mathbf{P}^\top \mathbf{y}_P + (\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1} \mathbf{Q}^\top \mathbf{y}_Q \quad (35)$$

By assumption 2, all rows of \mathbf{P} and \mathbf{Q} are sampled from a common Normal Distribution. Thus we are able to utilize properties of the Wishart Distribution, ?.

$$\mathbf{P}^\top \mathbf{P} = \sum_{\substack{i=1 \\ n\epsilon}}^{n*(1-\epsilon)} \mathbf{P}_i \mathbf{P}_i^\top \quad (36)$$

$$\mathbf{Q}^\top \mathbf{Q} = \sum_{j=1}^n \mathbf{Q}_j \mathbf{Q}_j^\top \quad (37)$$

Thus we can say $\mathbf{P}^\top \mathbf{P}$ and $\mathbf{Q}^\top \mathbf{Q}$ are sampled from the Wishart distribution.

$$\mathbf{P}^\top \mathbf{P} \sim \mathcal{W}(n(1-\epsilon), \mathbf{\Sigma}) \quad (38)$$

$$\mathbf{Q}^\top \mathbf{Q} \sim \mathcal{W}(n\epsilon, \mathbf{\Sigma}) \quad (39)$$

We can now use the Expected Value of the Wishart Distribution.

$$\mathbb{E}(\mathbf{P}^\top \mathbf{P}) = n(1-\epsilon)\mathbf{\Sigma} \quad (40)$$

$$\mathbb{E}(\mathbf{Q}^\top \mathbf{Q}) = n\epsilon\mathbf{\Sigma} \quad (41)$$

It thus follows

$$\mathbb{E}[\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q}] = n\mathbf{\Sigma} \quad (42)$$

Since we are interested in the pseudo-inverse, we will utilize the Inverse Wishart Distribution.

$$(\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1} \sim \mathcal{W}^{-1}(n, \mathbf{\Sigma}) \quad (43)$$

It thus follows by the expectation of the Inverse Wishart Distribution

$$\mathbb{E}[(\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1}] = n\mathbf{\Sigma}^{-1} \quad (44)$$

Now we will plug this into Equation 35:

$$\mathbb{E}[\mathbf{X}^\dagger \mathbf{y}] = (n\mathbf{\Sigma}^{-1}) \mathbf{P}^\top \mathbf{y}_P + (n\mathbf{\Sigma}^{-1}) \mathbf{Q}^\top \mathbf{y}_Q \quad (45)$$

$$= (n\mathbf{\Sigma}^{-1}) \mathbf{P}^\top (\mathbf{P}\beta + \epsilon_P) + (n\mathbf{\Sigma}^{-1}) \mathbf{Q}^\top (\mathbf{Q}\beta_Q^\top + \epsilon_Q) \quad (46)$$

$$= (n\mathbf{\Sigma}^{-1}) ((\mathbf{P}^\top \mathbf{P})\beta_P + (\mathbf{Q}^\top \mathbf{Q})(\beta_P + (\beta_Q - \beta_P))) \quad (47)$$

$$= (n\mathbf{\Sigma}^{-1}) ((n(1-\epsilon)\mathbf{\Sigma})\beta_P + n\epsilon\mathbf{\Sigma}(\beta_P + \mathbf{\Psi})) \quad (48)$$

$$= (n\mathbf{\Sigma}^{-1}) (n\mathbf{\Sigma}\beta_P + n\epsilon\mathbf{\Sigma}\mathbf{\Psi}) \quad (49)$$

$$= \beta_P + \epsilon(\mathbf{\Psi}) \quad (50)$$

This concludes the proof. \square

A.2 PROOF OF THEOREM 2

Proof. The first half of the proof follows from Appendix A.1. We start by noting new notation. Σ_P represents the covariance matrix for \mathbb{P} and Σ_Q represents the covariance matrix for \mathbb{Q} .

$$\mathbb{E} [\mathbf{P}^\top \mathbf{P}] = n(1 - \epsilon) \Sigma_P \quad (51)$$

$$\mathbb{E} [\mathbf{Q}^\top \mathbf{Q}] = n\epsilon \Sigma_Q \quad (52)$$

It thus follows

$$\mathbb{E} [\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q}] = (n(1 - \epsilon) \Sigma_P + n\epsilon \Sigma_Q) \quad (53)$$

This is where the structure of the proof because we can no longer follow the Inverse Wishart Distribution.

$$\mathbb{E} [(\mathbf{P}^\top \mathbf{P} + \mathbf{Q}^\top \mathbf{Q})^{-1}] = (n(1 - \epsilon) \Sigma_P + n\epsilon \Sigma_Q)^{-1} \quad (54)$$

Now we can use the Woodbury Formula ?

$$= n(1 - \epsilon) \Sigma_P^{-1} - n(1 - \epsilon) \Sigma_P^{-1} (n\epsilon \Sigma_Q^{-1}) \quad (55)$$

We will now calculate the expected optimal parameters by plugging this into Equation 35:

$$\mathbb{E} [\mathbf{X}^\dagger \mathbf{y}] = n(1 - \epsilon) \Sigma_P^{-1} (\mathbf{P}^\top \mathbf{P}) \beta_P - n(1 - \epsilon) \Sigma_P^{-1} (n\epsilon \Sigma_Q^{-1}) (\mathbf{Q}^\top \mathbf{Q}) \beta_P \quad (56)$$

$$= n(1 - \epsilon) \Sigma_P^{-1} (n(1 - \epsilon) \Sigma_P) \beta_P - n(1 - \epsilon) \Sigma_P^{-1} (n\epsilon \Sigma_Q^{-1}) (n\epsilon \Sigma_Q) \beta_P \quad (57)$$

$$= \beta_P - n(1 - \epsilon) \Sigma_P^{-1} \beta_P \quad (58)$$

This concludes the proof. \square

B GENERAL PROPERTIES OF SUB-QUANTILE MINIMIZATION

B.1 PROOF OF LEMMA 4.2

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

$$\partial(-g(t, \theta)) = \partial\left(-t + \frac{1}{np} \sum_{i=1}^n (t - \hat{\nu}_i)^+\right) \quad (59)$$

$$= \partial(-t) + \partial\left(\frac{1}{np} \sum_{i=1}^n (t - \hat{\nu}_i)^+\right) \quad (60)$$

$$= -1 + \frac{1}{np} \sum_{i=1}^n \partial(t - \hat{\nu}_i)^+ \quad (61)$$

$$= -1 + \frac{1}{np} \sum_{i=1}^n \begin{cases} 1, & \text{if } t > \hat{\nu}_i \\ 0, & \text{if } t < \hat{\nu}_i \\ [0, 1], & \text{if } t = \hat{\nu}_i \end{cases} \quad (62)$$

$$= 0 \text{ when } t = \hat{\nu}_{np} \quad (63)$$

This is the p -quantile of ν . Not necessarily the p -quantile of $Q_p(U)$ \square

B.2 PROOF OF LEMMA 4.3

Proof. Note that $t_k = \nu_{np}$ which is equivalent to $(\theta_k^\top \mathbf{x}_{np} - y_{np})^2$

$$\nabla_{\theta_k} g(t_{k+1}, \theta_k) = \nabla_{\theta_k} \left(\nu_{np} - \frac{1}{np} \sum_{i=1}^n (\nu_{np} - (\theta_k^\top \mathbf{x}_i - y_i)^2)^+ \right) \quad (64)$$

$$= \nabla_{\theta_k} \left((\theta_k^\top \mathbf{x}_{np} - y_{np})^2 - \frac{1}{np} \sum_{i=1}^n ((\theta_k^\top \mathbf{x}_{np} - y_{np})^2 - (\theta_k^\top \mathbf{x}_i - y_i)^2)^+ \right) \quad (65)$$

$$= \nabla_{\theta_k} (\theta_k^\top \mathbf{x}_{np} - y_{np})^2 - \frac{1}{np} \sum_{i=1}^n \nabla_{\theta_k} ((\theta_k^\top \mathbf{x}_{np} - y_{np})^2 - (\theta_k^\top \mathbf{x}_i - y_i)^2)^+ \quad (66)$$

$$= 2\mathbf{x}_{np}(\theta_k^\top \mathbf{x}_{np} - y_{np}) - \frac{1}{np} \sum_{i=1}^n 2\mathbf{x}_{np}(\theta_k^\top \mathbf{x}_{np} - y_{np}) - 2\mathbf{x}_i(\theta_k^\top \mathbf{x}_i - y_i) \begin{cases} 1, & \text{if } t > v_i \\ 0, & \text{if } t < v_i \\ [0, 1], & \text{if } t = v_i \end{cases} \quad (67)$$

$$= 2\mathbf{x}_{np}(\theta_k^\top \mathbf{x}_{np} - y_{np}) - \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_{np}(\theta_k^\top \mathbf{x}_{np} - y_{np}) - 2\mathbf{x}_i(\theta_k^\top \mathbf{x}_i - y_i) \quad (68)$$

$$= 2\mathbf{x}_{np}(\theta_k^\top \mathbf{x}_{np} - y_{np}) - 2\mathbf{x}_{np}(\theta_k^\top \mathbf{x}_{np} - y_{np}) + \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i(\theta_k^\top \mathbf{x}_i - y_i) \quad (69)$$

$$= \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i(\theta_k^\top \mathbf{x}_i - y_i) \quad (70)$$

This is the derivative of the np samples with lowest error with respect to θ . \square

B.3 PROOF OF LEMMA 4.5

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

$$\|\nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}', t) - \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}, t)\| \leq L \|\boldsymbol{\theta}' - \boldsymbol{\theta}\| \quad (71)$$

$$\left\| \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}', t) - \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta}, t) \right\| = \left\| \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i(\boldsymbol{\theta}'^T \mathbf{x}_i - y_i) - \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i(\boldsymbol{\theta}^T \mathbf{x}_i - y_i) \right\| \quad (72)$$

$$= \left\| \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i(\boldsymbol{\theta}'^T \mathbf{x}_i - \boldsymbol{\theta}^T \mathbf{x}_i) \right\| \quad (73)$$

$$= \left\| \frac{1}{np} \sum_{i=1}^{np} 2\mathbf{x}_i^T \mathbf{x}_i (\boldsymbol{\theta}'^T - \boldsymbol{\theta}^T) \right\| \quad (74)$$

$$\stackrel{\text{Cauchy-Schwarz}}{\leq} \left\| \frac{2}{np} \sum_{i=1}^{np} \|\mathbf{x}_i\|^2 \right\| \left\| \boldsymbol{\theta}'^T - \boldsymbol{\theta}^T \right\| \quad (75)$$

$$= L \left\| \boldsymbol{\theta}'^T - \boldsymbol{\theta}^T \right\| \quad (76)$$

where $L = \left\| \frac{2}{np} \sum_{i=1}^{np} \|\mathbf{x}_i\|^2 \right\|$

This concludes the proof.

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 $t_k \geq \hat{\nu}_k$

$$g(t_k, \boldsymbol{\theta}_k) = t_k - \frac{1}{np} \sum_{i=1}^n (t_k - \nu_i)^+ \quad (77)$$

$$= t_k - \frac{1}{np} (np)t_k + \frac{1}{np} \sum_{i=1}^{np} \nu_i + \frac{1}{np} \sum_{i=np}^n (t_k - \nu_i)^+ \quad (78)$$

$$= \frac{1}{np} \sum_{i=1}^{np} \nu_i + \frac{1}{np} \sum_{i=np}^n (t_k - \nu_i)^+ \quad (79)$$

$$g(t_{k+1}, \boldsymbol{\theta}_k) - g(t_k, \boldsymbol{\theta}_k) = \frac{1}{np} \sum_{i=1}^{np} \nu_i - \left(\frac{1}{np} \sum_{i=1}^{np} \nu_i + \frac{1}{np} \sum_{i=np}^n (t_k - \nu_i)^+ \right) \quad (80)$$

$$= -\frac{1}{np} \sum_{i=np}^n (t_k - \nu_i)^+ \quad (81)$$

Case (ii) $t_{k+1} > t_k$

Since we know t_k is less than ν_{np} , WLOG we will say t_k is greater than the lowest $n(p-\delta)$ elements,

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

$$g(t_k, \boldsymbol{\theta}_k) = t_k - \frac{1}{np} \sum_{i=1}^n (t_k - \boldsymbol{\nu}_i)^+ \quad (82)$$

$$= t_k - \frac{1}{np} \sum_{i=1}^{n(p-\delta)} (t_k - \boldsymbol{\nu}_i)^+ \quad (83)$$

$$= t_k - \frac{1}{np} (n(p-\delta))t_k + \frac{1}{np} \sum_{i=1}^{n(p-\delta)} \boldsymbol{\nu}_i \quad (84)$$

$$g(t_k, \boldsymbol{\theta}_{k+1}) - g(t_k, \boldsymbol{\theta}_k) = \frac{1}{np} \sum_{i=1}^{np} \boldsymbol{\nu}_i - \left(\delta t_k + \frac{1}{np} \sum_{i=1}^{n(p-\delta)} \boldsymbol{\nu}_i \right) \quad (85)$$

$$= \left(\frac{1}{np} \sum_{i=n(p-\delta)}^n \boldsymbol{\nu}_i \right) - \delta t_k \quad (86)$$

This concludes the proof. \square

C STOCHASTIC SUB-QUANTILE OPTIMIZATION

In the age of big data, stochastic methods are necessary for fast training of models to handle large amounts of data. In this section we will provide an algorithm for Stochastic Sub-Quantile Optimization and [prove convergence](#).

Algorithm 2: Stochastic Sub-Quantile Minimization Optimization Algorithm

Input: Training iterations T , Quantile p , Corruption Percentage ϵ , Input Parameters d , Batch Size m

Output: Trained Parameters, θ

Data: Inliers: $y|x \sim \mathcal{N}(x^2 - x + 2, 0.01)$, Outliers: $y|x \sim \mathcal{N}(-x^2 + x + 4, 0.01)$

```
1:  $\theta_1 \leftarrow \mathcal{N}(0, \sigma)^d$ 
2: for  $k \in 1, 2, \dots, T$  do
3:    $I \subseteq [n]$  of size  $m$ 
4:    $\nu = (X_I \theta_k - y_I)^2$ 
5:    $\hat{\nu} = \text{sorted}(\nu)$ 
6:    $t_{k+1} = \hat{\nu}_{mp}$ 
7:    $t_{k+1} = \frac{1}{mp} \sum_{i=1}^{mp} \nu_i$ 
8:    $L := \sum_{i=1}^{mp} \mathbf{x}_i^\top \mathbf{x}_i$ 
9:    $\alpha := \frac{1}{2L}$ 
10:   $\theta_{k+1} = \theta_k - \alpha \nabla_{\theta_k} g(t_{k+1}, \theta_k)$ 
11: end
12: return  $\theta_T$ 
```

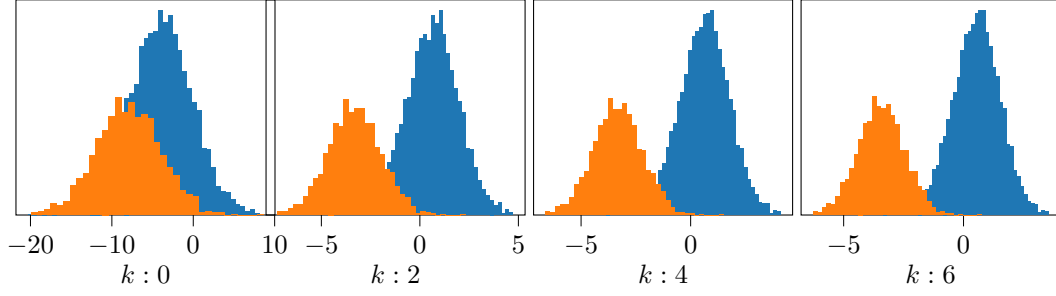


Figure 4: Residuals with respect to \mathbb{P} and \mathbb{Q}

D THEORY FOR LINEAR CORRUPTION

In this section, we provide rigorous theory for why Sub-Quantile Minimization works so well in the case of corruption of the form $\beta_Q^\top \mu = y_P + \epsilon_Q$.

Assumption 3. *The residuals of θ_k are normally distributed with respect to \mathbb{P} and \mathbb{Q} . In other words, $\theta_k \mathbf{p} - y_P$ and $\theta_k \mathbf{q} - y_Q$ are normally distributed.*

Assumption 3 can be visually verified in figure 4. Even after multiple iteration steps the residuals with respect to \mathbb{P} and \mathbb{Q} are still normal.

D.1 PROOF OF THEOREM 3

Proof. By Assumptions 1 and 2 we can do the following manipulations. Furthermore, please note \mathbf{P} is a random matrix, θ, β_P are non-random vectors, where $\beta^\top \mathbf{P}$ is a random vector-matrix product and $\mathbb{E}[\beta_P^\top \mathbf{p}] = y_P$ and $\text{Var}(\beta_P^\top \mathbf{p}) = \epsilon_P$. In this theoretical analysis, we assume no feature corruption, i.e., $\mathbb{E}[\mathbf{p}_i] = \mathbb{E}[\mathbf{q}_i] = \mu$. We will now analytically calculate the expected least squares error:

Recalculate expected value by using $\mathbb{E}[X^2] = \mathbb{E}[X]^2 + \text{Var}(X)$

$$\begin{aligned}
 \mathbb{E}[\theta^\top \mu - y_P] &= \mathbb{E}[\theta^\top \mu - (\beta_P^\top \mu + \epsilon_P)] \\
 &= \mathbb{E}[(\theta^\top - \beta_P^\top) \mu - \epsilon_P] \\
 &= \mathbb{E}[(\theta^\top - \beta_P^\top) \mu] + \mathbb{E}[-\epsilon_P] \\
 &= \mathbb{E}[\theta^\top - \beta_P^\top] \mathbb{E}[\mu] \\
 &= (\theta^\top - \beta_P^\top) \mu \\
 \text{Var}(\theta^\top \mu - y_P) &= \text{Var}(\theta^\top \mu - (\beta_P^\top \mu + \epsilon_P)) \\
 &= \text{Var}((\theta^\top - \beta_P^\top) \mu + \epsilon_P) \\
 &= \text{Var}((\theta^\top - \beta_P^\top) \mu) + \text{Var}(\epsilon_P) + \text{Cov}((\theta^\top - \beta_P^\top) \mu, \epsilon_P) \\
 &= (\theta^\top - \beta_P^\top) \mu (\theta - \beta_P) + \text{Var}(\epsilon_P)
 \end{aligned}$$

It thus follows:

$$\begin{aligned}
 \mathbb{E}[(\theta^\top \mu - y_P)^2] &= \mathbb{E}[(\theta^\top \mu - y_P)^2] + \text{Var}(\theta^\top \mu - y_P) \\
 &= ((\theta^\top - \beta_P^\top) \mu)^2 + (\theta^\top - \beta_P^\top) \mu (\theta - \beta_P) + \text{Var}(\epsilon_P)
 \end{aligned}$$

This result is accurate as it shows when $\theta = \beta_P$ the expectation is equal to the variance in β_P .

$$\Xi = \Xi$$

Therefore we can find the expectation over all of the data with:

$$\mathbb{E}[\theta^\top \mu - y] = (1 - \varepsilon)(\theta^\top - \beta_P^\top) \mu + \varepsilon(\theta^\top - \beta_Q^\top) \mu$$

Furthermore, we can find the variance over all of the data with:

$$\begin{aligned}
 \text{Var}(\theta^\top \mu - y) &= (1 - \varepsilon) \left(((\theta^\top - \beta_P^\top) \mu)^2 + (\theta^\top - \beta_P^\top) \mu (\theta - \beta_P) + \text{Var}(\epsilon_P) \right) \\
 &\quad + \varepsilon \left(((\theta^\top - \beta_Q^\top) \mu)^2 + (\theta^\top - \beta_Q^\top) \mu (\theta - \beta_Q) + \text{Var}(\epsilon_Q) \right)
 \end{aligned}$$

Furthermore, this is empirically shown to be the correct derivation by optimal error in Figure 3 converging to 0.1 which we see in Appendix F is equal to ϵ_P . Therefore we can similarly say,

$$\mathbb{E} \left[\|\boldsymbol{\theta}^\top \boldsymbol{\mu} - y_Q\|_2^2 \right] = ((\boldsymbol{\theta}^\top \boldsymbol{\mu}) - (\boldsymbol{\beta}_Q^\top \boldsymbol{\mu}))^2 + \boldsymbol{\theta}^\top \boldsymbol{\Sigma} \boldsymbol{\theta} - 2\boldsymbol{\theta}^\top \boldsymbol{\Sigma} \boldsymbol{\beta}_Q + \boldsymbol{\beta}_Q^\top \boldsymbol{\Sigma} \boldsymbol{\beta}_Q + \text{Var}(\epsilon_Q) \quad (87)$$

By assumption 3 we assume normal error, thus we are now interested in calculating the variance. Therefore we assume the distribution of squared errors is a non-central chi-squared distribution

Now that we have the Expected Value and Variance of Squared Error we now want to calculate the expected number of points in the p sub-quantile. This is equivalent to:

$$n(1 - \varepsilon)\mathbb{P} \left[\|\boldsymbol{\theta}^\top \boldsymbol{\mu} - y_P\|_2^2 \leq \mathcal{Q}_p \right] \quad (88)$$

We thus want to calculate the p -quantile of the Joint Distribution. We will define $\mathbb{Z} = (1 + \varepsilon)\mathbb{P} + \varepsilon\mathbb{Q}$. It thus follows:

$$\begin{aligned} \mathbb{E}[\mathbb{Z}] &= (1 - \varepsilon)\mathbb{E}[\mathbb{P}] + \varepsilon\mathbb{E}[\mathbb{Q}] \\ \text{Var}(\mathbb{Z}) &= (1 - \varepsilon)^2\mathbb{E}[\mathbb{P}] + \varepsilon^2\mathbb{E}[\mathbb{Q}] \end{aligned}$$

By assumption 3 we will assume \mathbb{Z} follows a chi-squared distribution. We can estimate the parameter k by the following:

$$k = \frac{\mathbb{E}[\mathbb{Z}]}{2\text{Var}(\mathbb{Z})} \quad (89)$$

It thus follows by the definition of the Chi-Squared Distribution:

$$\mathcal{Q}_p = \inf \left\{ x \in \mathbb{R} : p \leq \frac{1}{\Gamma(k/2)} \gamma \left(\frac{k}{2}, \frac{x}{2} \right) \right\} \quad (90)$$

This is efficiently calculated with computational solvers.

Since we assume \mathbb{P} and \mathbb{Q} are also chi-squared distribution.

$$\mathbb{P} \left[\|\boldsymbol{\theta}^\top \boldsymbol{\mu} - y_P\|_2^2 \leq \mathcal{Q}_p \right] \leq \mathcal{Q}_p \quad (91)$$

By definition of the chi-squared distribution, this is equivalent to the CDF of chi-squared with

$$k = \frac{\mathbb{E}[\mathbb{P}]}{2\text{Var}(\mathbb{P})} \quad (92)$$

$$\left(\mathbb{P} \left[\|\boldsymbol{\theta}^\top \boldsymbol{\mu} - y_P\|_2^2 \leq \mathcal{Q}_p \right] \leq \mathcal{Q}_p \right) = \frac{1}{\Gamma(k/2)} \gamma \left(\frac{k}{2}, \frac{\mathcal{Q}_p}{2} \right) \quad (93)$$

We test our theoretical results in figure ... □

D.2 PROOF OF THEOREM 4

Proof. Note Assumption 2 tells us the whole distribution is sampled from the same distribution. Let us define two functions for the empirical loss on \mathbb{P} and \mathbb{Q}

$$\phi(\boldsymbol{\theta}) = \frac{1}{np} \sum_{i=1}^m (\boldsymbol{\theta}^\top \mathbf{p}_i - y_i)^2 \quad (94)$$

$$\psi(\boldsymbol{\theta}) = \frac{1}{np} \sum_{i=1}^l (\boldsymbol{\theta}^\top \mathbf{q}_i - y_i)^2 \quad (95)$$

These two functions hold nice properties.

$$\nabla_{\boldsymbol{\theta}} \phi(\boldsymbol{\theta}) = \frac{1}{np} \sum_{i=1}^m 2\mathbf{p}_i (\boldsymbol{\theta}^\top \mathbf{p}_i - y_i) \quad (96)$$

$$\nabla_{\boldsymbol{\theta}} \psi(\boldsymbol{\theta}) = \frac{1}{np} \sum_{i=1}^l 2\mathbf{q}_i (\boldsymbol{\theta}^\top \mathbf{q}_i - y_i) \quad (97)$$

Here we note that the summation of these derivatives is equal to the theta update

$$\nabla_{\boldsymbol{\theta}_k} g(t_{k+1}, \boldsymbol{\theta}_k) = \nabla_{\boldsymbol{\theta}_k} \phi(\boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}_k} \psi(\boldsymbol{\theta}) \quad (98)$$

For a given optimization step, we will calculate the expected derivatives.

Recalculate expectation of derivative, should have the covariance matrix of \mathbf{p}

$$\mathbb{E}[\nabla_{\theta}\phi(\theta)] = \frac{1}{np} \sum_{i=1}^{np(1-\varepsilon)} 2\mathbb{E}[\mathbf{p}_i(\theta^{\top}\mathbf{p}_i - y_i)] \quad (99)$$

$$= \frac{2}{np} \sum_{i=1}^{np(1-\varepsilon)} \mathbb{E}[\mathbf{p}_i\mathbf{p}_i^{\top}\theta - \mathbf{p}_i(\beta_P^{\top}\mathbf{p}_i + \epsilon_P)] \quad (100)$$

$$= \frac{2}{np} \sum_{i=1}^{np(1-\varepsilon)} (\mu\mu^{\top} + \Sigma_P)\theta - (\mu\mu^{\top} + \Sigma_P)\beta_P \quad (101)$$

$$= 2(1-\varepsilon)(\mu\mu^{\top} + \Sigma_P)(\theta - \beta_P) \quad (102)$$

It thus similarly follows

$$\mathbb{E}[\nabla_{\theta}\psi(\theta)] = 2(\varepsilon)(\mu\mu^{\top} + \Sigma_Q)(\theta - \beta_Q) \quad (103)$$

Since we assume both \mathbb{P} and \mathbb{Q} are sampled from the same distribution. It follows $\Sigma_P = \Sigma_Q$, therefore we can calculate the expected derivative with respect to theta as:

$$\mathbb{E}[\nabla_{\theta}g(t_{k+1}, \theta_k)] = 2(\mu\mu^{\top} + \Sigma)(2\theta - (1-\varepsilon)\beta_P - \varepsilon\beta_Q)$$

Let us now calculate the expected change in θ in each iteration. We will start with the θ -update as described in Equation 11.

$$\mathbb{E}[\theta_{k+1}^{\top}] = \theta_k^{\top} - \frac{1}{2L}\mathbb{E}[\nabla_{\theta}g(t_{k+1}, \theta_k)] \quad (104)$$

We can now use the expected derivative in equation 102

$$= \theta_k^{\top} - \frac{1}{2L}(2\mu\mu^{\top}(\theta_k - (1-\varepsilon)\beta_P - \varepsilon\beta_Q)) \quad (105)$$

$$= \theta_k^{\top} - \frac{1}{L}(\mu\mu^{\top}(\theta_k - (1-\varepsilon)\beta_P - \varepsilon\beta_Q)) \quad (106)$$

$$= \left(\mathbf{I} - \frac{1}{L}\mu\mu^{\top}\right)\theta_k + \frac{1-\varepsilon}{L}\mu\mu^{\top}\beta_P + \frac{\varepsilon}{L}\mu\mu^{\top}\beta_Q \quad (107)$$

Let us now calculate $\mathbb{E}[\theta_{k+1}\mu]$

$$\mathbb{E}[\theta_{k+1}^{\top}\mu] = \left(\left(\mathbf{I} - \frac{1}{L}\mu\mu^{\top}\right)\theta_k + \frac{1-\varepsilon}{L}\mu\mu^{\top}\beta_P + \frac{\varepsilon}{L}\mu\mu^{\top}\beta_Q\right)^{\top}\mu \quad (108)$$

$$= \left(\theta_k^{\top}\left(\mathbf{I} - \frac{1}{L}\mu\mu^{\top}\right) + \frac{1-\varepsilon}{L}\beta_P^{\top}\mu\mu^{\top} + \frac{\varepsilon}{L}\beta_Q^{\top}\mu\mu^{\top}\right)\mu \quad (109)$$

$$= \left(\theta_k^{\top}\mu - \frac{1}{L}\theta_k^{\top}\mu\mu^{\top}\mu + \frac{1-\varepsilon}{L}\beta_P^{\top}\mu\mu^{\top}\mu + \frac{\varepsilon}{L}\beta_Q^{\top}\mu\mu^{\top}\mu\right) \quad (110)$$

Note $\mathbb{E}[\mu^{\top}\mu] = \text{Var}[\mu] + \mu^2 = C$ for simplicity

$$= \theta_k^{\top}\mu - \frac{C}{L}\theta_k^{\top}\mu + \frac{C(1-\varepsilon)}{L}\beta_P^{\top}\mu + \frac{C\varepsilon}{L}\beta_Q^{\top}\mu \quad (111)$$

$$\theta_{k+1}^{\top}\mu - \theta_k^{\top}\mu = \left(-\frac{C}{L}\theta_k^{\top} + \frac{C(1-\varepsilon)}{L}\beta_P^{\top} + \frac{C\varepsilon}{L}\beta_Q^{\top}\right)\mu \quad (112)$$

It thus follows nicely

$$\theta_{k+1} - \theta_k = -\frac{C}{L}\theta_k + \frac{C(1-\varepsilon)}{L}\beta_P + \frac{C\varepsilon}{L}\beta_Q \quad (113)$$

This gives us insight into how θ_{k+1} changes. We can now calculate a telescopic sum to see the change in θ after T iterations. We will first assume ε does not change. Furthermore, let us note we

start at $\theta_0 = \mathbf{0}$. Let us display a couple of iterations to show how θ_k changes

$$\theta_0 = \mathbf{0} \quad (114)$$

$$\theta_1 = \frac{C(1-\varepsilon)}{L}\beta_P + \frac{C\varepsilon}{L}\beta_Q \quad (115)$$

$$\theta_2 = -\frac{C}{L} \left(\frac{C(1-\varepsilon)}{L}\beta_P + \frac{C\varepsilon}{L}\beta_Q \right) + \frac{C(1-\varepsilon)}{L}\beta_P + \frac{C\varepsilon}{L}\beta_Q \quad (116)$$

$$= \frac{CL(1-\varepsilon)}{L^2}\beta_P + \frac{CL\varepsilon}{L^2}\beta_Q - \frac{C^2(1-\varepsilon)}{L^2}\beta_P - \frac{C^2\varepsilon}{L^2}\beta_Q \quad (117)$$

$$= \frac{(CL - C^2)(1-\varepsilon)}{L^2}\beta_P + \frac{(CL - C^2)\varepsilon}{L^2}\beta_Q \quad (118)$$

This only converges to β_P when ε is decreasing. Otherwise it will simply converge to the OLS optimal solution we proved in Theorem 1.

We are thus interested in $\mathbb{P}[\varepsilon \rightarrow 0]$. The fact that $\theta_0 = \mathbf{0}$ is vital to the convergence analysis. Furthermore, the coefficients of β_P and β_Q are also important for this analysis. This is why for the first iteration, it is vital not to randomize θ_0 , but to randomize the first np elements trained on. As initial θ_0 is deterministic of the final distribution.

Definition 8. A function f is L smooth if

$$\|f(\mathbf{x}) - f(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y}$$

We will now find the expected change in loss with respect to the distributions of \mathbb{P} and \mathbb{Q} in each iteration. Furthermore, let us note $g(t, \theta)$ is L -smooth with respect to θ . Thus we can use the properties of Definition 8.

$$\mathbb{E}[g(t_{k+1}, \theta_{k+1})] \leq \mathbb{E}[g(t, \theta_k)] + \langle \nabla_{\theta} \mathbb{E}[g(t_k, \theta_k)], \theta_{k+1} - \theta_k \rangle + \frac{L}{2} \|\theta_{k+1} - \theta_k\|_2^2 \quad (119)$$

$$= \mathbb{E}[g(t, \theta_k)] + \langle \nabla_{\theta} \phi(\theta) + \nabla_{\theta} \psi(\theta), \theta_{k+1} - \theta_k \rangle + \frac{L}{2} \|\theta_{k+1} - \theta_k\|_2^2 \quad (120)$$

We can use the θ update rule described in Equation 11

$$\begin{aligned} &= \mathbb{E}[g(t, \theta_k)] + \langle \nabla_{\theta} \phi(\theta) + \nabla_{\theta} \psi(\theta), -\frac{1}{L} (\nabla_{\theta} \phi(\theta_k) + \nabla_{\theta} \psi(\theta_k)) \rangle \\ &\quad + \frac{L}{2} \|\phi(\theta) + \nabla_{\theta} \psi(\theta)\|_2^2 \end{aligned} \quad (121)$$

We can now use the expected derivative described in Equation 102

$$\begin{aligned} &= \mathbb{E}[g(t, \theta_k)] + \langle 2\mu\mu^\top (\theta^\top - (1-\epsilon)\beta_P^\top - \epsilon\beta_Q^\top), -\frac{1}{L} 2\mu\mu^\top (\theta^\top - (1-\epsilon)\beta_P^\top - \epsilon\beta_Q^\top) \rangle \\ &\quad + \frac{1}{2L} (2\mu\mu^\top (\theta^\top - (1-\epsilon)\beta_P^\top - \epsilon\beta_Q^\top))^\top (2\mu\mu^\top (\theta^\top - (1-\epsilon)\beta_P^\top - \epsilon\beta_Q^\top)) \end{aligned} \quad (122)$$

$$= \mathbb{E}[g(t, \theta_k)] - \frac{1}{2L} (2\mu\mu^\top (\theta^\top - (1-\epsilon)\beta_P^\top - \epsilon\beta_Q^\top))^\top (2\mu\mu^\top (\theta^\top - (1-\epsilon)\beta_P^\top - \epsilon\beta_Q^\top)) \quad (123)$$

□

D.3 PROOF OF THEOREM 5

Proof. To converge,

□

E ADDITIONAL EXPERIMENTS

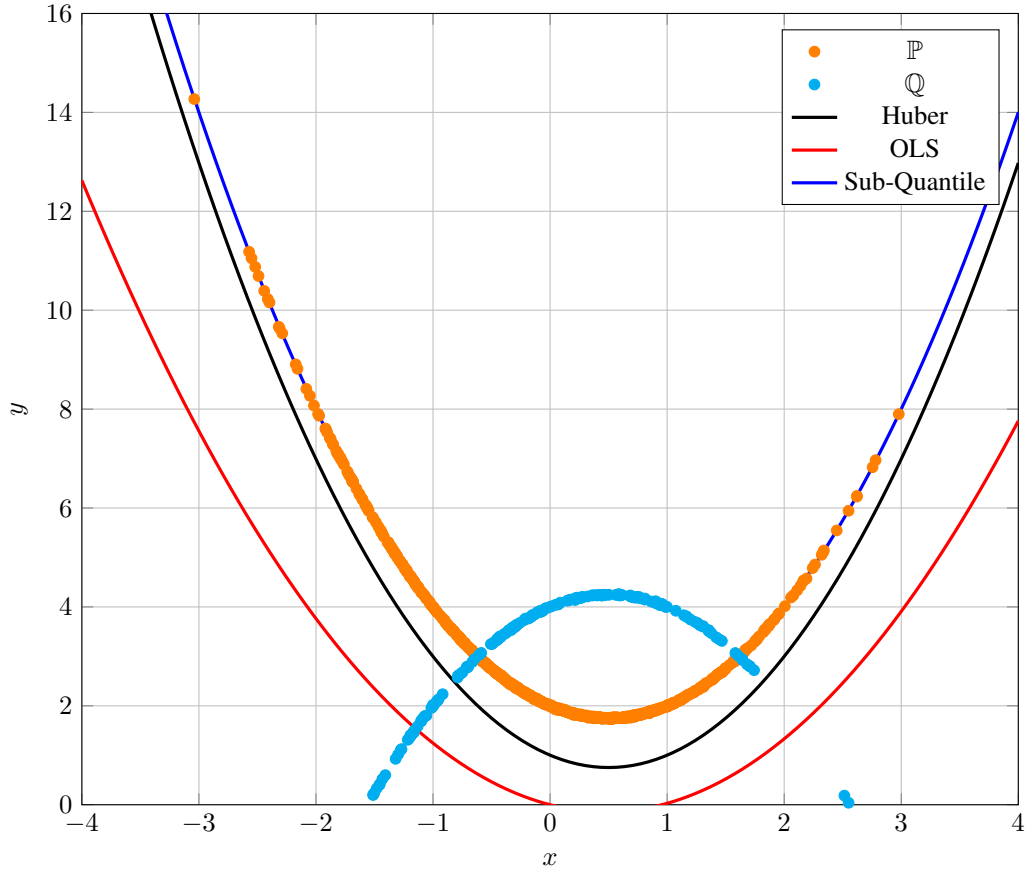


Figure 5: Quadratic Regression $n = 1000$ and $\epsilon = 0.2$

F EXPERIMENTAL DETAILS

F.1 STRUCTURED LINEAR REGRESSION DATASET

We will describe \mathbb{P} and \mathbb{Q} in the Linear Regression Dataset.

$$\mathbf{x} \sim \mathcal{N}(0, 4)^{500}$$

$$\mathbf{m} \sim \mathcal{N}(0, 4)^{500}$$

$$b \sim \mathcal{N}(0, 4)$$

$$\mathbf{m}' \sim \mathcal{N}(4, 4)^{500}$$

$$b' \sim \mathcal{N}(4, 4)$$

$$n_{\text{train}} = 2\text{e}3$$

$$\mathbb{P} : y|\mathbf{x} \sim \mathcal{N}(\mathbf{m}^\top \mathbf{x} + b, 0.1)$$

$$\mathbb{Q} : y|\mathbf{x} \sim \mathcal{N}(\mathbf{m}'^\top \mathbf{x} + b', 0.1)$$

Please note \mathbf{m} , b , \mathbf{m}' , b' , are all sampled independently.

F.2 NOISY LINEAR REGRESSION DATASET

We will describe \mathbb{P} and \mathbb{Q} in the Linear Regression Dataset.

$$\mathbf{x} \sim \mathcal{N}(0, 4)^{500}$$

$$\mathbf{m} \sim \mathcal{N}(0, 4)^{500}$$

$$b \sim \mathcal{N}(0, 4)$$

$$\mathbf{m}' \sim \mathcal{N}(4, 4)^{500}$$

$$b' \sim \mathcal{N}(4, 4)$$

$$n_{\text{train}} = 2\text{e}3$$

$$\mathbb{P} : y|\mathbf{x} \sim \mathcal{N}(\mathbf{m}^\top \mathbf{x} + b, 0.1)$$

$$\mathbb{Q} : y|\mathbf{x} \sim \mathcal{N}(\mathbf{m}'^\top \mathbf{x} + b', 0.1)$$

Please note \mathbf{m} , b , \mathbf{m}' , b' , are all sampled independently.

F.3 QUADRATIC REGRESSION DATASET

We will describe \mathbb{P} and \mathbb{Q} in the Quadratic Regression dataset.

$$x \sim \mathcal{N}(0, 1)$$

$$n_{\text{train}} = 10\text{e}4$$

$$\mathbb{P} : y|x \sim \mathcal{N}(x^2 - x + 2, 0.01)$$

$$\mathbb{Q} : y|x \sim \mathcal{N}(-x^2 + x + 4, 0.01)$$

F.4 DRUG DISCOVERY DATASET

This dataset is downloaded from ?. We utilize the same noise procedure as in ?.

\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

F.5 BASELINE METHODS IN SECTION 5

Here we will describe the objective functions used in the synthetic data experiments.

Ordinary Least Squares (OLS) can be solved utilizing the Moore Penrose Inverse.

$$\mathbf{X}^* = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y} \quad (124)$$

Huber Regression is solved with the following objective function.

$$L_\delta(y, f(\mathbf{x})) = \begin{cases} \frac{1}{2}(y - f(\mathbf{x}))^2 \\ \delta \cdot (|y - f(\mathbf{x})| - \frac{1}{2}\delta) & \text{otherwise} \end{cases} \quad (125)$$

RANSAC