ReLU Regression with Massart Noise

Anonymous Author(s)
Affiliation
Address
email

Abstract

We study the fundamental problem of ReLU regression, where the goal is to fit Rectified Linear units (ReLUs) to data. This supervised learning task is efficiently solvable in the realizable setting, but is known to be computationally hard in the agnostic setting. In this work, we focus on ReLU regression in the Massart noise model, a natural and well-studied semi-random noise model. In this model, the label of every point is generated according to a function in the class, but an adversary is allowed to change this value arbitrarily with some probability, which is *at most* $\eta < 1/2$. We develop an efficient algorithm that achieves exact parameter recovery in this model under mild anti-concentration assumptions on the underlying distribution. Such assumptions are necessary for exact recovery to be information-theoretically possible. We demonstrate that our algorithm significantly outperforms naive applications of ℓ_1 and ℓ_2 regression on both synthetic and real data.

1 Introduction

Learning in the presence of noisy data is a key challenge in machine learning. In addition to its fundamental nature, the broad algorithmic problem of designing efficient noise-tolerant learners is of critical importance in several high-dimensional data analysis applications, including machine learning security (aka data poisoning attacks) [2, 6, 36] and extracting useful insights from real datasets with natural outliers, e.g., in biological settings [33, 31, 28]. The goal in such settings is to design computationally efficient estimators that can tolerate a constant fraction of outliers, independent of the dimensionality of the data.

A standard framework to study learning with noisy data comes from robust statistics [20, 22]. Classical work, starting with Tukey and Huber in the 1960s, developed statistically optimal robust estimators for a number of settings. These early methods lead to exponential-time algorithms, even for the most basic high-dimensional task of mean estimation. A recent line of work in computer science, starting with [11, 27], developed the first computationally efficient robust learning algorithms for various high-dimensional tasks. Since these early works, there has been significant progress in algorithmic robust high-dimensional statistics by several communities. The reader is referred to [13] for a recent survey.

The aforementioned line of work focuses on a contamination model in which an adversary is allowed to corrupt an *arbitrary* $\eta < 1/2$ fraction of the dataset. A limitation of these algorithmic results is the assumption that the inliers comes from a "nice' distribution, e.g., a Gaussian or log-concave distribution. Without such strong distributional assumptions, robust learning with fully adversarial data quickly becomes computationally intractable [3, 19, 8]. This suggests the following natural question:

Are there realistic semi-random noise models in which efficient robust learning is possible without (strong) assumptions on the distribution of inliers?

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

Here we focus on robust *supervised* learning. Specifically, we study regression problems in the presence of Massart (bounded) noise [29]. The Massart model is a classical noise model originally defined in the context of *binary classification*. In this model, an adversary has control over a random $\eta < 1/2$ fraction of the labels. Recent work [10] gave the first efficient learning algorithm for linear separators in the Massart model *without distributional assumptions*. In this work, we ask to what extent such algorithmic results are possible for learning real-valued functions. To state our contributions, we formally define the noise model in our context.

Definition 1.1 (Learning Real-valued Functions with Massart Noise)**.** *Given $0 \leq \eta < 1/2$, a distribution $\mathcal{D}_{\mathbf{x}}$ on $\mathbb{R}^d$, and a function $f : \mathbb{R}^d \to \mathbb{R}$, an adversary operates as follows. The algorithm specifies m and the adversary generates $(\mathbf{x}_i, f(\mathbf{x}_i))_{i=1}^m$ i.i.d. samples from $\mathcal{D}_{\mathbf{x}}$. For each sample, the adversary is allowed to change the label $f(\mathbf{x}_i)$ with probability $\eta(\mathbf{x}_i) \leq \eta$ and after inspecting which samples can be corrupted, it may change the label to an arbitrary value.*

1.1 Main Results

To build up to the more difficult case of ReLUs, we first focus on the simpler case of linear regression, which is an interesting problem itself. Linear regression is a prototypical problem in statistics with a range of applications in signal processing (e.g., face recognition, time series analysis) and various other data analysis tasks. The reader is referred to [34, 5] and references therein.

Our goal is to identify a linear relation $y = \mathbf{w}^* \cdot \mathbf{x}$ that the true samples $(\mathbf{x}, y)$ (inliers) satisfy. We show that, under the mild yet necessary assumption that the distribution is not fully concentrated on any subspace, the problem is identifiable and we can efficiently and exactly recover $\mathbf{w}^*$.

Theorem 1.2 (Exact Recovery of Linear Functions)**.** *Let $\mathcal{D}_{\mathbf{x}}$ be a distribution on $\mathbb{R}^d$ that has zero measure on any linear subspace and let $\eta < 1/2$ be the upper bound on the Massart noise rate. Denote $\mathbf{w}^*$ the vector representing the true linear function. There is an algorithm that draws $\tilde{O}(\frac{d^3}{(1-2\eta)^2})$ samples, runs in $\mathrm{poly}(d, b, (1-2\eta)^{-1})$ time, where b is an upper bound on the bit complexity of the samples and parameters, and outputs $\mathbf{w}^*$ with probability at least $9/10$.*

We provide extended results that relax the density assumption on $\mathcal{D}_{\mathbf{x}}$ in Theorem C.2 of Appendix C so that the only assumption on $\mathcal{D}_{\mathbf{x}}$ is that it is full-dimensional in $\mathbb{R}^d$. It is important to note that if the distribution was concentrated entirely on a linear subspace, it would be information-theoretically impossible to determine the orthogonal component of $\mathbf{w}^*$ to that subspace. That is, our anti-concentration assumption on the distribution is necessary for identifiability to be possible. Even when these assumptions are violated and the problem is non-identifiable, we provide a (weaker) PAC learning guarantee for the linear case in Theorem D.1 of Appendix D.

Our main algorithmic result is for the problem of ReLU regression, where all the inliers satisfy $y = \mathrm{ReLU}(\mathbf{w}^* \cdot \mathbf{x})$ when an η fraction of the labels are corrupted according to the Massart noise model of Definition 1.1. . ReLU regression is a fundamental primitive in the theory of neural networks that has been extensively studied in recent years, see, e.g., [18, 9] and references therein. Even in this more challenging case, we show it is possible to efficiently identify the true parameters $\mathbf{w}^*$, as long as every homogeneous halfspace contains a non-negligible fraction of the sample points.

Theorem 1.3 (Exact Recovery of ReLUs)**.** *Let $\mathcal{D}_{\mathbf{x}}$ be a distribution on $\mathbb{R}^d$ that has zero measure on any linear subspace and such that $\Pr_{\mathbf{x}\sim\mathcal{D}_{\mathbf{x}}}[\mathbf{w} \cdot \mathbf{x} \geq 0] \geq \lambda$ for any $\mathbf{w} \in \mathbb{R}^d$. Let $\eta < 1/2$ be the upper bound on the Massart noise rate. Denote $\mathbf{w}^*$ the parameter vector of the target ReLU. There is an algorithm that draws $\tilde{O}(\frac{d^3}{\lambda^2(1-2\eta)^2})$ samples, runs in time $\mathrm{poly}(d, b, \lambda^{-1}, (1-2\eta)^{-1})$, and outputs $\mathbf{w}^*$ with probability at least $9/10$.*

We provide extended results that relax the density assumption on $\mathcal{D}_{\mathbf{x}}$ in Theorem C.3 of Appendix C.

We note that the assumption on the mass of the halfspaces is necessary for identifiability. Indeed, if there was a halfspace parameterized by $\mathbf{w} \in \mathbb{R}^d$ such that $\Pr_{\mathbf{x}\sim\mathcal{D}_{\mathbf{x}}}[\mathbf{w} \cdot \mathbf{x} \geq 0] = 0$, it would be impossible to distinguish between the functions $\mathrm{ReLU}(\mathbf{w} \cdot \mathbf{x})$ and $\mathrm{ReLU}(2\mathbf{w} \cdot \mathbf{x})$ (even without noise), as all points would have 0 label. It remains an interesting open problem whether similar PAC learning guarantees can be obtained for the case of ReLU regression. We suspect that this problem is computationally hard in full generality.

1.2 Technical Overview

As explained in the introduction, the focus of our work is on the problem of robust regression in the presence of outliers. When $\eta < \frac{1}{2}$ fraction of the data is arbitrarily corrupted, the goal is to compute a function that maximizes the number of inliers. Given a sufficient number of samples from a full-dimensional distribution, this function is unique for the class of ReLUs and matches the true function with high probability. However, even in the simpler case of linear functions, the corresponding computational problem of ℓ_0-minimization is computationally hard, as it is an instance of robust subspace recovery [21, 25].

Our positive results are driven by relaxing the assumption that an *arbitrary* $\eta < \frac{1}{2}$ fraction of the points are corrupted. Instead, we consider a more restricted adversary that is presented with a uniformly random η fraction of the points, which can be corrupted arbitrarily at will. We view this as an instance of Massart noise, where with probability $\eta(\mathbf{x}) \leq \eta$, the adversary is allowed to corrupt a point arbitrarily (see also Definition 1.1).

ℓ_0 to ℓ_1 minimization Given this milder corruption model, we propose novel algorithms for efficient exact recovery of the underlying function. We obtain our algorithms by replacing the ℓ_0-minimization with ℓ_1-minimization, which can be shown to converge to the true function *in the limit* and is efficient to optimize in the linear regression case. For intuition, consider a single-point distribution that always outputs samples $(\mathbf{x}, y)$, where $\mathbf{x}$ is always the same but y may be different. The Massart assumption indicates that the value of y is correct more than half of the time, so the estimate that maximizes the number of correct samples (ℓ_0-minimizer) recovers the underlying function. However, if one considers the ℓ_1-minimizer, i.e., the value v that minimizes $\mathbb{E}[|y - v|]$, this corresponds to the median value of y which is also correct if more than half samples are correct.

We generalize this intuition and propose a natural and tight condition under which simple ℓ_1-minimization results in the true ℓ_0-minimizer (see Lemma 2.2). While this condition holds under Massart noise for arbitrary distributions, it can fail to hold with high probability when considering only a finite set of samples from the distribution. In particular, the sample complexity would depend on the concentration properties of the distribution on $\mathbf{x}$.

Transforming the points via radial-isotropy The main technical idea behind obtaining sample efficient algorithms that run in polynomial time is to transform the original point set into an equivalent one that satisfies the required properties with high probability, as it becomes sufficiently concentrated. In particular, performing a linear transformation mapping every point $\mathbf{x}$ to $\mathbf{A}\mathbf{x}$, while keeping the corresponding label y, is without loss of generality, as we are interested in identifying the true (generalized) linear function that depends only on the inner product of every point with a parameter vector $\mathbf{w}$. Finding such a vector $\mathbf{w}'$ in the transformed space $\mathbf{A}\mathbf{x}$ results in the equivalent vector $\mathbf{w} = \mathbf{A}\mathbf{w}'$ in the original space. Moreover, an additional operation we can perform is to take a single sample $(\mathbf{x}, y)$ and multiply it by a positive scalar $\lambda > 0$ to replace it with the sample $(\lambda\mathbf{x}, \lambda y)$. For both the linear and ReLU cases, any sample that is an inlier for the true function, remains an inlier after this transformation.

We can use these two operations to bring our point set in radial-isotropic position, i.e., so that all the $\mathbf{x}$'s in the dataset are unit-norm and the variance in any direction is identical:

Definition 1.4 (Radial Isotropy)**.** *Given $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathcal{S}^{d-1}$, $\mathbf{A} : \mathbb{R}^d \to \mathbb{R}^d$ is a radial-isotropic transformation if $\sum_{i=1}^n \frac{(\mathbf{A}\mathbf{x}_i)(\mathbf{A}\mathbf{x}_i)^T}{\|\mathbf{A}\mathbf{x}_i\|^2} = \frac{n}{d} I$. For $0 < \gamma < 1$, we say that the points are in γ-approximate radial-isotropic position, if for all $\mathbf{v} \in \mathcal{S}^{d-1}$, it holds that $\sum_{i=1}^n \frac{d}{n}(\mathbf{x}_i \cdot \mathbf{v})^2 \geq 1 - \gamma$.*

In such a normalized position, we can argue that with high probability the weight of all inliers in every direction is more than the weight of the outliers, which guarantees that the ℓ_1-minimizer will converge to the true function.

Learning ReLUs We build on these ideas from learning linear functions to learning ReLUs. Since even the ℓ_1 minimization problem is non-convex (unlike the case for linear functions), we instead show that any suboptimal guess for the parameter vector $\mathbf{w}$ can be improved by moving along the gradient direction of the ℓ_1-loss function. However, for this result to hold with a small number of samples, we need to again bring to radial-isotropic position the points that fall in the linear (positive) part of ReLU for the current guess vector $\mathbf{w}$. In contrast to the linear case, though, where this

transformation was applied once, in this case it needs to be applied again with every new guess. This results in a function that changes at every step, which is not suitable for direct optimization.

We show however that identifying such a direction of improvement yields a separating hyperplane for the current guess and the optimal parameter vector. We exploit this to efficiently identify the optimal parameter vector $\mathbf{w}^*$ by running the Ellipsoid method using our separation oracle.

Using these ideas, our algorithms can efficiently recover the underlying function exactly using very few samples. Our algorithms make mild genericity assumptions about the position of the points, requiring that the points are not concentrated on a lower-dimensional subspace or, for the case of ReLUs, do not lie entirely in an origin-centered half-space. As already mentioned, such assumptions are necessary for the purposes of identifiability.

1.3 Related Work

Given the extensive literature on robust regression, we highlight and compare our results with the most relevant prior work.

ReLU regression For agnostic learning, where there is no assumption made regarding the covariates and the labels, [18, 14] give hardness results for achieving error arbitrarily close to the optimal even when the underlying distribution is standard Gaussian. Despite this, [9] provides algorithms for constant-factor approximations for agnostic ReLU learning.

Learning ReLUs in the realizable setting is also well-motivated and actively studied. [35, 23] show that gradient descent efficiently learns ReLUs under a standard Gaussian in the realizable setting, whereas [38] relaxes the distributional assumption to a milder one. Concurrently, [24] also studies ReLU under Massart noise in the realizable setting, albeit with the adversarial noise bounded by a constant. Moreover, their stochastic gradient descent algorithm for ReLUs does not get arbitrarily close to the optimal as their bounds depend on the concentration properties of the marginal distribution.

Comparison of noise models In the fully adversarial setting, [5] show efficient algorithms for robust linear regression in a corruption model that allows adversarial corruptions to the responses (but not the covariates). Their deterministic condition on the covariates amounts to strong concentration bounds on the underlying distribution and their algorithms can only tolerate a constant but small breakdown point for corruptions ($\eta < 1/50$). [26, 15, 12] give robust algorithms in Huber's contamination model (where both the covariates and their labels can be corrupted). These results require strong distributional assumptions about the inliers and can only guarantee approximate recovery.

[7] also considers a Massart-like adversarial noise model, in a more general regression setting. For the realizable case that we consider in this work, their results yield significantly weaker guarantees. First, they do not achieve exact recovery and their guarantee depends on concentration properties of the underlying distribution. Their algorithm also incurs a polynomial dependence of the magnitude of the true parameter in the sample complexity. In essence, these results are comparable to the guarantees of directly running gradient descent on the ℓ_1-loss under our problem setting. Finally, our results work up to information-theoretic limit $\eta < 1/2$, while their algorithm only works up to $\eta < 1/3$.

Finally, there is a line of work [4, 37, 16, 32] that studies robust linear regression under oblivious noise where the adversary can corrupt the labels with additive noise that is independent of $\mathbf{x}$. More precisely, the oblivious adversary corrupts the vector of labels $\mathbf{y} \in \mathbb{R}^m$ by adding a ηm-sparse corruption vector $\mathbf{b}$. Since $\mathbf{b}$ is independent of the covariates, oblivious noise can be viewed as corrupting a sample with probability η with a random non-zero entry of $\mathbf{b}$. Thus, oblivious noise can be viewed as a special case of Massart noise. We formally compare these two noise models in more detail in Appendix E.

2 Warmup: Linear Regression with Massart Noise

In this section, we establish structural conditions under which we can perform efficient ℓ_0-minimization for linear functions and ReLUs under Massart noise. It is imperative that we find the ℓ_0-minimizer with respect to $\mathbf{w}$ since, with a sufficient number of samples, the ℓ_0-minimizer is the

true function we wish to recover. We then show that appropriately transforming the data via radial-isotropic transformationand then solving for the empirical ℓ_1-loss $\arg\min_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{m}\sum_{i=1}^m |\tilde{y}_i - \mathbf{w}\cdot\tilde{\mathbf{x}}_i|$ can efficiently recover the true parameter $\mathbf{w}^*$. We describe the algorithm for recovering linear functions below.

Algorithm 1 Linear function recovery via radial isotropy

Draw $m = \tilde{O}(\frac{d^3}{(1-2\eta)^2})$ samples $(\mathbf{x}_i, y_i)_{i=1}^m$ with η-Massart noise
Compute $\mathbf{A}$ that puts $(\mathbf{x}_i, y_i)_{i=1}^m$ in $1/2$-approximate radial-isotropic position
$\hat{\mathbf{w}} \leftarrow \arg\min_{\mathbf{w}\in\mathbb{R}^d} \sum_{i=1}^m |\frac{y_i}{\|\mathbf{A}\mathbf{x}_i\|_2} - \mathbf{w}\cdot\frac{\mathbf{A}\mathbf{x}_i}{\|\mathbf{A}\mathbf{x}_i\|_2}|$ by solving the LP
return $\mathbf{A}\hat{\mathbf{w}}$

In fact, for the first step, there is no need to compute the exact radial-isotropic transformation as an approximate one will suffice. As a relaxation to (exact) radial-isotropic position, we state the computational complexity of computing an approximate transformation in Lemma 2.1, which we prove in the Appendix.

Lemma 2.1. *Given $S \subset \mathbb{R}^d$ in general position, there is an algorithm that computes an invertible linear transformation $\mathbf{A}$ such that $\left\{\frac{\mathbf{A}\mathbf{x}}{\|\mathbf{A}\mathbf{x}\|} : \mathbf{x}\in S\right\}$ is in γ-approximate radial-isotropic position in $\mathrm{poly}(n, d, b, \gamma^{-1})$ time where b is an upper bound on the bit complexity of the parameters and samples in S. Morever, the ratio between the largest and smallest eigenvalue of $\mathbf{A}$ is at most $2^{\mathrm{poly}(n,d,b)}$.*

Given that computing such approximate transformation $\mathbf{A}$ and solving a linear program (LP) are efficient, Algorithm 1 gives us the polynomial runtime for Theorem 1.2.

The proof of Theorem 1.2 relies on two key ideas. First, we can minimize the empirical ℓ_1-loss with respect to $\mathbf{w}$ instead of the ℓ_0-loss because the two are equivalent under the structural condition of Lemma 2.2. Once the following sufficient condition is satisfied, we can efficiently solve for ℓ_1-minimization as this is efficient in the case of linear functions by solving an LP.

Lemma 2.2 (Structural Condition for Recovery)**.** *Let $f : \mathbb{R}\to\mathbb{R}$ and $\mathcal{H}$ be a hypothesis class such that $\mathcal{H} = \{h_{\mathbf{w}} : h_{\mathbf{w}}(\mathbf{x}) = f(\mathbf{w}\cdot\mathbf{x}), \mathbf{w}\in\mathbb{R}^d\}$. Given m samples $(\mathbf{x}_i, y_i)_{i=1}^m$, let the ℓ_0-minimizer $h_{\mathbf{w}^*} = \arg\min_{h\in\mathcal{H}} \frac{1}{m}\sum_{i=1}^m \|y_i - h(\mathbf{x}_i)\|_0$ be unique. If*

$$\sum_{y_i = h_{\mathbf{w}^*}(\mathbf{x}_i)} |h^-(\mathbf{x}_i)| > \sum_{y_i \neq h_{\mathbf{w}^*}(\mathbf{x}_i)} |h^-(\mathbf{x}_i)| \tag{$\star$}$$

for all $h^- \in \{h_1 - h_2 : h_1, h_2 \in \mathcal{H}, h_1 \neq h_2\}$, then $h_{\mathbf{w}^*}$ is also the ℓ_1-minimizer $\arg\min_{h\in\mathcal{H}} \frac{1}{m}\sum_{i=1}^m |y_i - h(\mathbf{x}_i)|$.

Proof. Let $\mathbf{w}^*$ be the parameter corresponding to the minimizer $h_{\mathbf{w}^*}$. Define $\hat{L}(\mathbf{w}) = \frac{1}{m}\sum_{i=1}^m |y_i - f(\mathbf{w}\cdot\mathbf{x}_i)|$. Given the strict inequality in ($\star$), for non-zero $r\in\mathbb{R}^d$, we have that

$$\begin{aligned}
m(\hat{L}(\mathbf{w}^*+\mathbf{r}) - \hat{L}(\mathbf{w}^*)) &= \sum_{i=1}^m |f((\mathbf{w}^*+\mathbf{r})\cdot\mathbf{x}_i) - y_i| - \sum_{i=1}^m |f(\mathbf{w}^*\cdot\mathbf{x}_i) - y_i| \\
&= \sum_{y_i = h_{\mathbf{w}^*}(\mathbf{x}_i)} |f((\mathbf{w}^*+\mathbf{r})\cdot\mathbf{x}_i) - f(\mathbf{w}^*\cdot\mathbf{x}_i)| + \sum_{y_i \neq h_{\mathbf{w}^*}(\mathbf{x}_i)} |f((\mathbf{w}^*+\mathbf{r})\cdot\mathbf{x}_i) - y_i| - \sum_{y_i \neq h_{\mathbf{w}^*}(\mathbf{x}_i)} |f(\mathbf{w}^*\cdot\mathbf{x}_i) - y_i| \\
&\geq \sum_{y_i = h_{\mathbf{w}^*}(\mathbf{x}_i)} |h_{\mathbf{w}^*+\mathbf{r}}(\mathbf{x}_i) - h_{\mathbf{w}^*}(\mathbf{x}_i)| - \sum_{y_i \neq h_{\mathbf{w}^*}(\mathbf{x}_i)} |h_{\mathbf{w}^*+\mathbf{r}}(\mathbf{x}_i) - h_{\mathbf{w}^*}(\mathbf{x}_i)| > 0.
\end{aligned}$$

Therefore, the ℓ_0-minimizer $h_{\mathbf{w}^*}$ is also the ℓ_1-minimizer $\arg\min_{h\in\mathcal{H}} \frac{1}{m}\sum_{i=1}^m |y_i - h(\mathbf{x}_i)|$. □

The structural condition ($\star$) for linear functions reduces to having the sum of $|\mathbf{r}\cdot\mathbf{x}_i|$ for the "good" points be greater than the sum of $|\mathbf{r}\cdot\mathbf{x}_i|$ for the "bad" points in every direction $\mathbf{r}$. However, this implies that if one sample is much greater in norm than the others in some direction, this point can have undue influence and may easily dominate the ℓ_1-loss. Therefore, without any preprocessing or transformation to the data, one has to rely on naively increasing the sample complexity until there are enough points in this direction to satisfy condition ($\star$). Instead, we minimize the dominating effects of such outlier points and reduce the sample complexity through transforming the dataset with radial isotropy. We defer the proof of Theorem 1.2 to Appendix A.

3 ReLU Regression with Massart Noise

In this subsection, we study the problem of exact recovery for ReLUs in the presence of Massart noise. For the case of ReLUs, we can still use the structural condition of Lemma 2.2 to do ℓ_1-minimization $\arg\min_{\mathbf{w}} \frac{1}{m}\sum_{i=1}^m |y_i - \mathrm{ReLU}(\mathbf{w}\cdot\mathbf{x}_i)|$. However, minimizing this objective is no longer straightforward, because the objective function is non-convex. Despite this fact, it is possible to exactly recover a ReLU under mild anti-concentration assumptions on the underlying distribution.

The key idea behind Theorem 1.3 is establishing the condition under which we can compute an efficient separation oracle between the query $\mathbf{w}$ and the true parameter $\mathbf{w}^*$. Once we obtain a separation oracle, we can use the Ellipsoid method to recover $\mathbf{w}^*$ exactly. In turn, similarly to Lemma 2.2, we identify a sufficient structural condition on the dataset, which allows us to efficiently compute a separating hyperplane between $\mathbf{w}$ and $\mathbf{w}^*$ if $\mathbf{w} \neq \mathbf{w}^*$, and then use radial-isotropic transformations such that this condition is satisfied. We state this separation condition in the following lemma.

Lemma 3.1 (Separation Condition for Recovery)**.** *Let $\mathcal{H}$ be an hypothesis class such that $\mathcal{H} = \{h_{\mathbf{w}} : h_{\mathbf{w}}(\mathbf{x}) = f(\mathbf{w}\cdot\mathbf{x}), \mathbf{w}\in\mathbb{R}^d\}$ where $f:\mathbb{R}\to\mathbb{R}$ is monotonically non-decreasing. Given a set of m samples $(\mathbf{x}_i, y_i)_{i=1}^m$, let $\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathbb{R}^d}\frac{1}{m}\sum_{i=1}^m \|y_i - f(\mathbf{w}\cdot\mathbf{x}_i)\|_0$ be unique. Let $\Delta > 0$ and denote $\mathcal{B}(\mathbf{w},\Delta)$ to be the open ball of radius Δ centered at $\mathbf{w}$. Given a query $\mathbf{w}\notin\mathcal{B}(\mathbf{w}^*,\Delta)$, if*

$$(1/m)\sum_{y_i = f(\mathbf{w}^*\cdot\mathbf{x}_i)} |(\mathbf{w}-\mathbf{w}^*)\cdot\mathbf{x}_i| f'(\mathbf{w}\cdot\mathbf{x}_i) - (1/m)\sum_{y_i\neq f(\mathbf{w}^*\cdot\mathbf{x}_i)} |(\mathbf{w}-\mathbf{w}^*)\cdot\mathbf{x}_i| f'(\mathbf{w}\cdot\mathbf{x}_i) \geq \Delta \quad (\dagger)$$

then we can efficiently compute a separating hyperplane between $\mathbf{w}$ and $\mathcal{B}(\mathbf{w}^*,\Delta/2)$.

In particular, the gradient of the empirical ℓ_1-loss gives us the separating hyperplane above. Other than the fact that only the points in the non-negative side of the halfspace of $\mathbf{w}$ are considered in the separation condition ($\dagger$), the condition resembles much of the structural condition used for linear functions. Analogously, we apply a radial-isotropic transformation to the points of $\mathbf{w}\cdot\mathbf{x}_i \geq 0$. Thus, we have the following sub-procedure of the Ellipsoid method.

Algorithm 2 Separation oracle sub-procedure

Input: $(\mathbf{x}_i, y_i)_{i=1}^m$ with Massart noise, query $\mathbf{w}_0$
Output: Unless $\mathbf{w}_0 \in \mathcal{B}(\mathbf{w}^*,\Delta)$, a separating hyperplane between $\mathbf{w}_0$ and $\mathcal{B}(\mathbf{w}^*,\Delta/2)$
if $\mathrm{ReLU}(\mathbf{w}_0\cdot\mathbf{x})$ fits at least $\frac{m}{2}$ points **then**
  return $\mathbf{w}_0$ as true parameter $\mathbf{w}^*$
$S \leftarrow \{(\mathbf{x}_i, y_i) : \mathbf{w}_0\cdot\mathbf{x}_i \geq 0 \text{ for } i\in[m]\}$
Compute $\mathbf{A}$ that puts $S_{\mathbf{x}}$ into $1/2$-approximate radial-isotropic position
$\mathbf{r} \leftarrow \frac{1}{|S|}\sum_{(\mathbf{x}_i,y_i)\in S} \frac{\mathbf{A}\mathbf{x}_i}{\|\mathbf{A}\mathbf{x}_i\|_2}\cdot\mathrm{sgn}(\mathbf{w}_0\cdot\mathbf{x}_i - y_i)$
return separating hyperplane $\mathbf{r}\cdot(\mathbf{A}^{-1}\mathbf{w}) < \mathbf{r}\cdot(\mathbf{A}^{-1}\mathbf{w}_0)$

The main difference between the algorithm for ReLUs and linear functions is that here we must apply a different radial-isotropic transformation to every new subset of points in every iteration, depending on the query $\mathbf{w}_0$. In turn, the algorithm transforms the space according a new transformation $\mathbf{A}$ and computes a separating hyperplane and transforms the hyperplane back into the original space. Due to these repeated transformations, the proof of Theorem 1.3 requires a more intricate argument to make the Ellipsoid method work correctly. We now prove Theorem 1.3.

Proof of Theorem 1.3. Assume two points $\mathbf{u},\mathbf{v}$ that are seperated by the hyperplane $\mathbf{w}\cdot\mathbf{x} = b$ such that $\mathbf{w}\cdot\mathbf{u} < b < \mathbf{w}\cdot\mathbf{v}$. Then for linear transformation $\mathbf{A}$, the seperating hyperplane for $\mathbf{A}^{-1}\mathbf{u}, \mathbf{A}^{-1}\mathbf{v}$ is $(\mathbf{A}\mathbf{w})\cdot\mathbf{x} = b$. Assume the Separation Condition from Lemma 3.1 holds for a set of points $(\mathbf{A}\mathbf{x}_i/\|\mathbf{A}\mathbf{x}_i\|_2, y_i/\|\mathbf{A}\mathbf{x}_i\|_2)$. Let $\mathbf{w}_0$ be the original query. Then for the query $\mathbf{A}^{-1}\mathbf{w}_0$, the separation is $\mathbf{r}\cdot\mathbf{w}_{out} < \mathbf{r}\cdot(\mathbf{A}^{-1}\mathbf{w}_0)$ where

$$\begin{aligned}\mathbf{r} &= \frac{1}{|S|}\sum_{(\mathbf{x}_i,y_i)\in S}\frac{\mathbf{A}\mathbf{x}_i}{\|\mathbf{A}\mathbf{x}_i\|_2}\cdot\mathrm{sgn}(\mathbf{A}^{-1}\mathbf{w}_0\cdot\frac{\mathbf{A}\mathbf{x}_i}{\|\mathbf{A}\mathbf{x}_i\|_2} - \frac{y_i}{\|\mathbf{A}\mathbf{x}_i\|_2}) \\ &= \frac{1}{|S|}\sum_{(\mathbf{x}_i,y_i)\in S}\tilde{\mathbf{x}}_i\cdot\mathrm{sgn}(\mathbf{w}_0\cdot\mathbf{x}_i - y_i).\end{aligned}$$

Thus the separation for $\mathbf{w}_0$ and $\mathbf{w}^*$ is $\mathbf{r}\cdot(\mathbf{A}^{-1}\mathbf{w}) < \mathbf{r}\cdot(\mathbf{A}^{-1}\mathbf{w}_0)$.

Now it remains to check the sample complexity necessary for the Separation Condition from Lemma 3.1. Each unique set of $\{(\mathbf{x}_i, y_i) : \mathbf{w}_0 \cdot \mathbf{x}_i \geq 0 \text{ for } i \in [m]\}$ determines a radial isotropic transformation but there can only be at most m^{d+1} unique sets by the VC dimension of halfspaces. So there are only at most m^{d+1} radial isotropic transformations we have to consider. Let $\mathbf{A}$ be the linear transformation of the radial isotropic transformation applied to points of $\mathbf{w}_0 \cdot \mathbf{x}_i \geq 0$. Denote $(\tilde{\mathbf{x}}, \tilde{y}) = (\frac{\mathbf{A}\mathbf{x}}{\|\mathbf{A}\mathbf{x}\|_2}, \frac{y}{\|\mathbf{A}\mathbf{x}\|_2})$, $\tilde{\mathbf{w}} = \mathbf{A}^{-1}\mathbf{w}$, and let $\tilde{\mathcal{D}}$ be $\mathcal{D}|_{\{\mathbf{w}_0 \cdot \mathbf{x} \geq 0\}}$ transformed by $\mathbf{A}$ then normalized so that $\tilde{\mathcal{D}}$ lies on $\mathcal{S}^{d-1}$. Then, for all m^{d+1} transformations, we have the following VC inequality using $m = \tilde{O}(\frac{d}{\epsilon^2})$ with high probability:

$$\sup_{\mathbf{w} \in \mathbb{R}^d} \Big| \Pr_{\tilde{\mathbf{x}} \sim \tilde{\mathcal{D}}_{\mathbf{x}}} [|(\mathbf{w} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}| \mathbb{1}\{\mathbf{w} \cdot \tilde{\mathbf{x}} \geq 0,\ y = \mathrm{ReLU}(\mathbf{w}^* \cdot \mathbf{x})\} > t] - \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\{|(\mathbf{w} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}_i| > t,\ \mathbf{w} \cdot \tilde{\mathbf{x}}_i \geq 0,\ y_i = \mathrm{ReLU}(\mathbf{w}^* \cdot \mathbf{x}_i)\}\Big| \leq \epsilon.$$

Let $S = \{(\tilde{\mathbf{x}}_i, \tilde{y}_i) : \mathbf{w} \cdot \mathbf{x}_i \geq 0\}$. Similarly to the proof of Theorem 1.2, we have that $\max_{(\tilde{\mathbf{x}}, \tilde{y}) \in S} |\mathbf{r} \cdot \tilde{\mathbf{x}}| \leq \frac{(1-\gamma)d}{1-\gamma-\epsilon d} \mathbb{E}_{\tilde{\mathcal{D}}_{\mathbf{x}}}[|\mathbf{r} \cdot \tilde{\mathbf{x}}|]$. Then, we can write

$$\begin{aligned}
&(1/|S|) \sum_{(\tilde{\mathbf{x}}, \tilde{y}) \in S} |(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}| \mathbb{1}\{y = \mathrm{ReLU}(\mathbf{w}^* \cdot \mathbf{x})\} \\
&= (m/|S|) \int_0^\infty \big((1/m) \sum_{i=1}^{m} \mathbb{1}\{|(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}_i| > t,\ \tilde{\mathbf{w}} \cdot \tilde{\mathbf{x}}_i \geq 0,\ y_i = \mathrm{ReLU}(\mathbf{w}^* \cdot \mathbf{x}_i\}\big) dt \\
&\geq \mathbb{E}_{\tilde{\mathcal{D}}_{\mathbf{x}}}[|(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}| \mathbb{1}\{y = \mathrm{ReLU}(\mathbf{w}^* \cdot \mathbf{x})\}] - (\epsilon m/|S|) \max_{(\tilde{\mathbf{x}}, \tilde{y}) \in S} |(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}| \\
&\geq (1 - \eta)\, \mathbb{E}_{\tilde{\mathcal{D}}_{\mathbf{x}}}[|(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}|] - (\epsilon m d/(|S|(1 - \epsilon d))) \mathbb{E}_{\tilde{\mathcal{D}}_{\mathbf{x}}}[|(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}|]
\end{aligned}$$

Setting $\epsilon = O(\lambda(1 - 2\eta)/d)$, we can bound $m/|S|$ be at most a constant times λ^{-1} for all m^{d+1} possible subsets S. Then we have

$$(1/|S|) \sum_{(\tilde{\mathbf{x}}, \tilde{y}) \in S} |(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}| \mathbb{1}\{\tilde{y} = \mathrm{ReLU}(\tilde{\mathbf{w}}^* \cdot \tilde{\mathbf{x}})\} \geq ((1/2) + (1 - 2\eta)/4) \mathbb{E}_{\tilde{\mathcal{D}}_{\mathbf{x}}}[|(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}|].$$

We can apply the same to the corrupted points in S with $\leq (1/2 - (1 - 2\eta)/4) \mathbb{E}_{\tilde{\mathcal{D}}_{\mathbf{x}}}[|(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}|]$. Thus, for points $(\tilde{\mathbf{x}}, \tilde{y}) \in S$, we have the condition

$$\begin{aligned}
(1/|S|)\Big(\sum_{\tilde{y} = \mathrm{ReLU}(\tilde{\mathbf{w}}^* \cdot \tilde{\mathbf{x}})} |(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}| - \sum_{\tilde{y} \neq \mathrm{ReLU}(\tilde{\mathbf{w}}^* \cdot \tilde{\mathbf{x}})} |(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}| \Big) &\geq (1/2 - \eta) \mathbb{E}_{\tilde{\mathcal{D}}_{\mathbf{x}}}[|(\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*) \cdot \tilde{\mathbf{x}}|] \\
&\geq (1/2 - \eta) \|\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*\|_2 / d\,.
\end{aligned}$$

By the Separation Condition of Lemma 3.1, the inequality above implies that we can find a hyperplane that separates $\tilde{\mathbf{w}}$ and $\mathcal{B}(\tilde{\mathbf{w}}^*, \Delta/2)$ where $\Delta = \frac{1-2\eta}{2d} \|\tilde{\mathbf{w}} - \tilde{\mathbf{w}}^*\|_2$. In the original space where we multiply $\mathbf{A}$ to the hyperplane and all points, we have that the transformed hyperplane seperates $\mathbf{w}$ and $\mathcal{B}(\mathbf{w}^*, \Delta'/2)$ where $\Delta' = \frac{1-2\eta}{2d} \cdot \frac{\lambda_{\min}(\mathbf{A})}{\lambda_{\max}(\mathbf{A})} \|\mathbf{w} - \mathbf{w}^*\|_2$. This follows by $\Delta \geq \frac{1-2\eta}{2d\lambda_{\max}(\mathbf{A})} \|\mathbf{w} - \mathbf{w}^*\|_2$ and transforming by $\mathbf{A}$, the minimum distance from $\mathbf{w}^*$ to the new hyperplane is at least $\Delta \cdot \lambda_{\min}(\mathbf{A})$.

If $\mathbf{w} \neq \mathbf{w}^*$, by bounded bit complexity b, we have that ellipsoid method will decrease the volume of the ellipsoid at every step but the ball with radius $\frac{(1-2\eta)\lambda_{\min}(\mathbf{A})}{4d\lambda_{\max}(\mathbf{A})}$ will always be contained in the ellipsoid. Thus the algorithm will terminate in $\mathrm{poly}(d, b, 1/(1 - 2\eta))$ iterations since $\log\left(\frac{\lambda_{\min}(\mathbf{A})}{\lambda_{\max}(\mathbf{A})}\right) = \mathrm{poly}(d, b, 1/(1 - 2\eta))$. □

4 Experiments

In this section, we apply our algorithms that are based on radial-isotropic transformations to both synthetic and real datasets and compare robustness in regression with other baseline methods of ℓ_1 and ℓ_2-regression. Our experiments demonstrate the efficacy of radial-isotropic transformations in robust regression and how our algorithms outperform simple regression methods.

All experiments were done on a laptop computer with a 2.3 GHz Dual-Core Intel Core i5 CPU and 8 GB of RAM. We ran CVXPY's linear program solver for ℓ_1-regression for linear functions.

Recovering Linear Functions We first show how our algorithm based on radial-isotropic position (Algorithm 1) compares to naive ℓ_1 regression in exact recovery using an LP solver. As another baseline, we also ran ℓ_1-regression with a normalization preprocessing step, where we normalize all points $(\mathbf{x}, y)$ to $(\frac{\mathbf{x}}{\|\mathbf{x}\|}, \frac{y}{\|x\|})$. We did not run regression with an isotropic-transformation preprocessing step because this yields identical results as naive regression with no preprocessing.

We evaluated different transformations to the data on the following synthetic distribution. Define a mixture of Gaussians $\mathcal{D}_{\mathbf{x}} = \frac{1}{2}\mathcal{N}(\mathbf{e}_1, \frac{1}{d^2}I_d) + \frac{1}{2d}\sum_{i=1}^{d}\mathcal{N}(d\mathbf{e}_i, \frac{1}{d^2}I_d)$ where $\mathbf{e}_i$ denotes the i-th standard basis vector and $d = 30$. Let $\mathbf{w}^* = 9\mathbf{e}_2 + \sum_{i=1}^{d}\mathbf{e}_i$. For various noise levels η, consider the following η-Massart adversary: the labels for all $\mathbf{x}$ for which any coordinate is greater than $\frac{d}{2}$ are flipped to $-\mathbf{w}^* \cdot \mathbf{x}$ with probability η, and the labels for all other points are not flipped. Essentially, only the points *not* from $\mathcal{N}(\mathbf{e}_1, \frac{1}{d^2}I_d)$ are affected by Massart noise.

We measured the exact parameter recovery rate, which captures how often the algorithm solves for $\mathbf{w}^*$ exactly. We varied the noise rate η while running the methods with 120 samples from $\mathcal{D}_{\mathbf{x}}$. We also varied the sample size while keeping the noise level $\eta = 0.25$. We ran 200 iterations for each measurement of exact recovery rate and the error bars represent two standard deviations away from the mean.

Recovering ReLUs For ReLUs, we used the same distribution as the experiments for linear functions to generate samples. We ran and compared constant-step-sized gradient descent on the empirical ℓ_1-loss $\frac{1}{m}\sum_{i=1}^{m}|y_i - \mathrm{ReLU}(\mathbf{w}^* \cdot \mathbf{x}_i)|$ with different transformations to the data. We ran gradient descent since our separation oracle for the Ellipsoid method bears similarities with gradient descent. As seen in Lemma 3.1, this is due to the fact that our separating hyperplane is based on the gradient of the empirical ℓ_1-loss of a subset of points.

The experiment is set up with $\eta = 0.4$, $\mathbf{w}^* = 9\mathbf{e}_2 + \sum_{i=1}^{d}\mathbf{e}_i$, $\mathbf{w}_0 = 0$, 240 samples from $\mathcal{D}_{\mathbf{x}}$, and gradient descent step size of 1. For 'Original', we use a step size of $1/465$ to keep the magnitude of the points $\mathbf{x}_i$ comparable to that of the transformed points $\tilde{\mathbf{x}}_i$.

In Figure 2(a), 'Original' corresponds to naive gradient descent, while 'Normalized' has a normalization preprocessing step. The transformations of 'Isotropic' and 'Radial-isotropic' follow our algorithm for ReLUs from Section 3 where the transformation is only applied to the positive-side points of $\mathbf{w} \cdot \mathbf{x}_i \geq 0$ for the current hypothesis $\mathbf{w}$. The gradient is then calculated with the transformed points $\tilde{\mathbf{x}}_i$ and appropriately transformed back to the original space in order to update $\mathbf{w}$. The gradient descent updates under transformation $\mathbf{A}$ and step size α is the following:

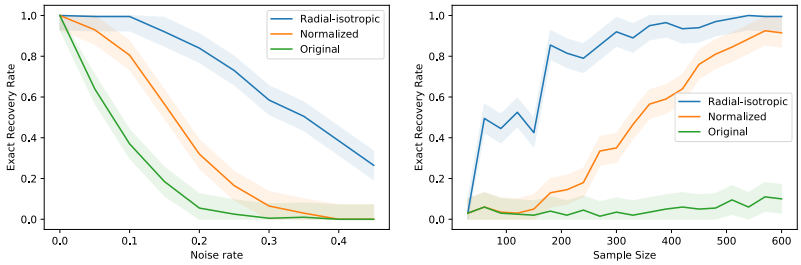
$$\mathbf{w}' \leftarrow \mathbf{A}^{-1}\mathbf{w} \quad \text{then} \quad \mathbf{w} \leftarrow \mathbf{w} - \alpha \cdot (\mathbf{A}\nabla_{\mathbf{w}'}L')$$

where L' denotes the empirical ℓ_1-loss for the transformed subset of points. This update method is directly adapted from our Ellipsoid method.

Drug Discovery Dataset The drug discovery dataset was originally curated by [30] and we used the same dataset as the one used in [12]. The dataset has a training and test set of 3084 and 1000 points of 410 dimensions. The η-Massart noise adversary corrupts the training data $(\mathbf{x}_i, y_i)$ so that all points are corrupted to flip labels to $-100y_i$ with probability η.

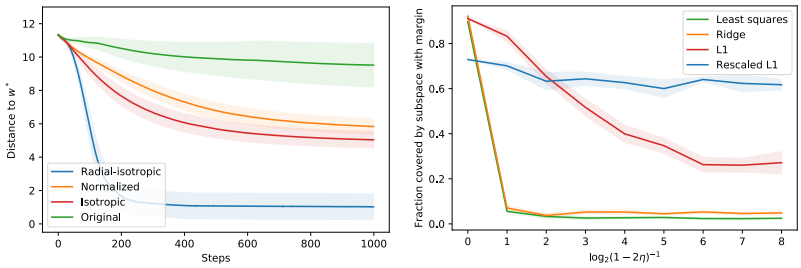
We compared ℓ_1-regression with radial-isotropic transformation ('Rescaled L1') to other baseline methods, such as least squares and naive ℓ_1-regression. For ridge regression, we optimized the regularization coefficient based on the uncorrupted data. We measure performance by computing the fraction of the test set that lies within the subspace generated by the output vector with a margin of 2.

Results In Figure 1, our algorithm with radial-isotropic transformation outperforms other baseline methods in robustness with respect to the noise level and in efficiency with respect to the sample size. This is in line with the results of Theorem 1.2. In fact, our experiments on ReLUs also empirically demonstrate that radial isotropy significantly improves ReLU regression via gradient descent by making the dataset more robust to noise at each iteration. Interestingly for the drug discovery dataset, although ℓ_1-regression with radial isotropy performs slightly worse than naive ℓ_1-regression when there is minimal noise, it significantly outperforms the baseline methods at regimes of high noise levels.



(a) Recovering Linear (vs. noise rate) (b) Recovering Linear (vs. sample size)

Figure 1: Experiments for exact parameter recovery of linear functions on synthetic data. Exact recovery rate (y-axis) measures how often the algorithm outputs the true parameter out of 200 trials. We compare Algorithm 1 with naive ℓ_1-minimization and ℓ_1-minimization with normalized data. Error bars cover two standard deviations from the mean.



(a) Gradient Descent on ReLU (b) Drug Discovery Dataset

Figure 2: On the left, we compare the performance of gradient descent with different transformations applied to synthetic data. Transformatioms are applied in a similar way to Algorithm 4. At each iteration, we measure the ℓ_2-distance to the optimal solution. On the right, we compare different regression methods to real data where the labels are artificially corrupted with η-Massart noise. Rescaled L1 represents ℓ_1-minimization on the data transformed by radial-isotropic transformation. We report the fraction of points that lie in the subspace generated by its output with margin.

5 Conclusion

In this paper, we have proposed a Massart (bounded) noise model on real-valued functions, generalizing the original noise model for binary classification into a noise model for regression. Although this adversarial model is slightly more restricted than arbitrary corruptions, we have presented algorithms that exactly recover its true parameter for classes of linear functions and ReLUs with only mild assumptions on the underlying distribution. In contrast to works of robust regression that require strong distributional assumptions, our framework and results may be seen as an intricate balance between slightly weakening the adversarial framework yet affording generality in the underlying distribution.

In the process, we have provided structural conditions on which ℓ_0-minimization is efficiently computable for both functions. Our algorithms rely on transforming the dataset, using radial-isotropic transformations and spectral outlier removal, so that we can satisfy these conditions with fewer samples. Furthermore, we have shown an efficient PAC learning algorithm for linear functions in the presence of Massart noise. We also empirically validate our results through our experiments on both synthetic and real data.

References

[1] Artstein-Avidan, S., Kaplan, H., and Sharir, M. On radial isotropic position: Theory and algorithms. *arXiv preprint arXiv:2005.04918* (2020).

[2] BARRENO, M., NELSON, B., JOSEPH, A. D., AND TYGAR, J. D. The security of machine learning. *Machine Learning 81*, 2 (2010), 121–148.

[3] BERNHOLT, T. Robust estimators are hard to compute. Tech. rep., University of Dortmund, Germany, 2006.

[4] BHATIA, K., JAIN, P., KAMALARUBAN, P., AND KAR, P. Consistent robust regression. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017* (2017), pp. 2107–2116.

[5] BHATIA, K., JAIN, P., AND KAR, P. Robust regression via hard thresholding. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015* (2015), pp. 721–729.

[6] BIGGIO, B., NELSON, B., AND LASKOV, P. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012* (2012).

[7] CHEN, S., KOEHLER, F., MOITRA, A., AND YAU, M. Online and distribution-free robustness: Regression and contextual bandits with huber contamination. *arXiv preprint arXiv:2010.04157* (2020).

[8] DANIELY, A. Complexity theoretic limitations on learning halfspaces. In *Proceedings of the 48th Annual Symposium on Theory of Computing, STOC 2016* (2016), pp. 105–117.

[9] DIAKONIKOLAS, I., GOEL, S., KARMALKAR, S., KLIVANS, A. R., AND SOLTANOLKOTABI, M. Approximation schemes for 'ssion. In *Conference on Learning Theory, COLT 2020* (2020), J. D. Abernethy and S. Agarwal, Eds., vol. 125 of *Proceedings of Machine Learning Research*, PMLR, pp. 1452–1485.

[10] DIAKONIKOLAS, I., GOULEAKIS, T., AND TZAMOS, C. Distribution-independent pac learning of halfspaces with massart noise. In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 4751–4762.

[11] DIAKONIKOLAS, I., KAMATH, G., KANE, D. M., LI, J., MOITRA, A., AND STEWART, A. Robust estimators in high dimensions without the computational intractability. In *Proc. 57th IEEE Symposium on Foundations of Computer Science (FOCS)* (2016), pp. 655–664.

[12] DIAKONIKOLAS, I., KAMATH, G., KANE, D. M., LI, J., STEINHARDT, J., AND STEWART, A. SEVER: A robust meta-algorithm for stochastic optimization. In *Proc. 36th International Conference on Machine Learning (ICML)* (2019), pp. 1596–1606.

[13] DIAKONIKOLAS, I., AND KANE, D. M. Recent advances in algorithmic high-dimensional robust statistics. *CoRR abs/1911.05911* (2019).

[14] DIAKONIKOLAS, I., KANE, D. M., AND ZARIFIS, N. Near-optimal sq lower bounds for agnostically learning halfspaces and relus under gaussian marginals. *arXiv preprint arXiv:2006.16200* (2020).

[15] DIAKONIKOLAS, I., KONG, W., AND STEWART, A. Efficient algorithms and lower bounds for robust linear regression. In *Proc. 30th Annual Symposium on Discrete Algorithms (SODA)* (2019), pp. 2745–2754.

[16] D'ORSI, T., NOVIKOV, G., AND STEURER, D. Regress consistently when oblivious outliers overwhelm. *arXiv preprint arXiv:2009.14774* (2020).

[17] DUNAGAN, J., AND VEMPALA, S. Optimal outlier removal in high-dimensional spaces. *J. Computer & System Sciences 68*, 2 (2004), 335–373.

[18] GOEL, S., KARMALKAR, S., AND KLIVANS, A. R. Time/accuracy tradeoffs for learning a relu with respect to gaussian marginals. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019* (2019), pp. 8582–8591.

[19] GURUSWAMI, V., AND RAGHAVENDRA, P. Hardness of learning halfspaces with noise. In *Proc. 47th IEEE Symposium on Foundations of Computer Science (FOCS)* (2006), IEEE Computer Society, pp. 543–552.

[20] HAMPEL, F. R., RONCHETTI, E. M., ROUSSEEUW, P. J., AND STAHEL, W. A. *Robust statistics. The approach based on influence functions*. Wiley New York, 1986.

[21] HARDT, M., AND MOITRA, A. Algorithms and hardness for robust subspace recovery. In *Proc. 26th Annual Conference on Learning Theory (COLT)* (2013), pp. 354–375.

[22] HUBER, P. J., AND RONCHETTI, E. M. *Robust statistics*. Wiley New York, 2009.

[23] KALAN, S. M. M., SOLTANOLKOTABI, M., AND AVESTIMEHR, A. S. Fitting relus via sgd and quantized sgd. In *2019 IEEE International Symposium on Information Theory (ISIT)* (2019), IEEE, pp. 2469–2473.

[24] KARMAKAR, S., MUKHERJEE, A., AND MUTHUKUMAR, R. A study of neural training with iterative non-gradient methods. *arXiv e-prints* (2020), arXiv–2005.

[25] KHACHIYAN, L. On the complexity of approximating extremal determinants in matrices. *Journal of Complexity 11*, 1 (1995), 138–153.

[26] KLIVANS, A., KOTHARI, P., AND MEKA, R. Efficient algorithms for outlier-robust regression. In *Proc. 31st Annual Conference on Learning Theory (COLT)* (2018), pp. 1420–1430.

[27] LAI, K. A., RAO, A. B., AND VEMPALA, S. Agnostic estimation of mean and covariance. In *Proc. 57th IEEE Symposium on Foundations of Computer Science (FOCS)* (2016), pp. 665–674.

[28] LI, J., ABSHER, D., TANG, H., SOUTHWICK, A., CASTO, A., RAMACHANDRAN, S., CANN, H., BARSH, G., FELDMAN, M., CAVALLI-SFORZA, L., AND MYERS, R. Worldwide human relationships inferred from genome-wide patterns of variation. *Science 319* (2008), 1100–1104.

[29] MASSART, P., AND NEDELEC, E. Risk bounds for statistical learning. *Ann. Statist. 34*, 5 (10 2006), 2326–2366.

[30] OLIER, I., SADAWI, N., BICKERTON, G. R., VANSCHOREN, J., GROSAN, C., SOLDATOVA, L., AND KING, R. D. Meta-qsar: a large-scale application of meta-learning to drug design and discovery. *Machine Learning 107*, 1 (2018), 285–311.

[31] PASCHOU, P., LEWIS, J., JAVED, A., AND DRINEAS, P. Ancestry informative markers for fine-scale individual assignment to worldwide populations. *Journal of Medical Genetics 47* (2010), 835–847.

[32] PESME, S., AND FLAMMARION, N. Online robust regression via sgd on the l1 loss. *arXiv preprint arXiv:2007.00399* (2020).

[33] ROSENBERG, N., PRITCHARD, J., WEBER, J., CANN, H., KIDD, K., ZHIVOTOVSKY, L., AND FELDMAN, M. Genetic structure of human populations. *Science 298* (2002), 2381–2385.

[34] ROUSSEEUW, P. J., AND LEROY, A. M. *Robust Regression and Outlier Detection*. John Wiley & Sons, Inc., New York, NY, USA, 1987.

[35] SOLTANOLKOTABI, M. Learning relus via gradient descent. In *Advances in neural information processing systems* (2017), pp. 2007–2017.

[36] STEINHARDT, J., KOH, P. W., AND LIANG, P. S. Certified defenses for data poisoning attacks. In *Advances in Neural Information Processing Systems 30* (2017), pp. 3520–3532.

[37] SUGGALA, A. S., BHATIA, K., RAVIKUMAR, P., AND JAIN, P. Adaptive hard thresholding for near-optimal consistent robust regression. In *Conference on Learning Theory, COLT 2019* (2019), pp. 2892–2897.

[38] YEHUDAI, G., AND OHAD, S. Learning a single neuron with gradient methods. In *Conference on Learning Theory* (2020), PMLR, pp. 3756–3786.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section 2
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? **[TODO]**
   (c) Did you discuss any potential negative societal impacts of your work? **[TODO]**
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Section 4
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4 about the drug discovery dataset
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]