Robust Regression via Online Feature Selection under Adversarial Data Corruption

Xuchao Zhang[†], Shuo Lei[†], Liang Zhao[‡], Arnold P. Boedihardjo[§], Chang-Tien Lu[†]
[†]Virginia Tech, Falls Church, VA, USA
[‡]George Mason University, Fairfax, VA, USA
[§]U. S. Army Corps of Engineers, Alexandria, VA, USA
[†]{xuczhang, slei, ctlu}@vt.edu, [‡]lzhao9@gmu.edu, [§]arnold.p.boedihardjo@usace.army.mil

***Abstract*—The presence of data corruption in user-generated streaming data, such as social media, motivates a new fundamental problem that learns reliable regression coefficient when features are not accessible entirely at one time. Until now, several important challenges still cannot be handled concurrently: 1) corrupted data estimation when only partial features are accessible; 2) online feature selection when data contains adversarial corruption; and 3) scaling to a massive dataset. This paper proposes a novel RObust regression algorithm via Online Feature Selection (*RoOFS*) that concurrently addresses all the above challenges. Specifically, the algorithm iteratively updates the regression coefficients and the uncorrupted set via a robust online feature substitution method. Extensive empirical experiments in both synthetic and real-world data sets demonstrated that the effectiveness of our new method is superior to that of existing methods in the recovery of both feature selection and regression coefficients, with very competitive efficiency.**

I. Introduction

The presence of noise and data corruption in real-world data can be inevitably caused by various reasons such as experimental errors, accidental outliers, or even adversarial data attacks. In traditional robust regression problem, reliable regression coefficients are learned in the presence of adversarial data corruptions in its response vector. A commonly adopted model from existing methods assumes that the observed response is obtained from the generative model $\boldsymbol{y} = X^T\boldsymbol{\beta}^* + \boldsymbol{u}$, where $\boldsymbol{\beta}^*$ is the true regression coefficients we wish to recover and $\boldsymbol{u}$ is the corruption vector with adversarial values. In the problem setting, the data matrix X is assumed to contain all the features that can be accessed at any time and by arbitrarily many times. However, the assumption is no longer suitable to the following scenarios in the applications that contain exponentially increasing user-generated contents: 1) *features are too many to be loaded entirely*. Features grows dramatically fast and becomes extremely large in. For instance, over 4.7 million movies and televisions with 8.3 million reviews in IMDb[1] online movie and television review website, which makes it hard to load all the features entirely for any machine learning models using the movies as features. 2) *features are generated dynamically*. For example, people create and use new terms and hashtags all the time in Twitter, and the "Likes" [1] on newly-generated articles in Facebook can be considered as new features describing the interestingness of the user. 3) Therefore, it is necessary to address online features in traditional robust regression as a new fundamental problem; however, current methods either focus on robust regression or online feature learning separately. Specifically, existing robust learning methods typically focus on modeling the entire dataset with all the features at once; however, they may meet the bottleneck in terms of computation and memory as more and more data sets are becoming available in the era of data explosion.

To the best of our knowledge, our proposed approach is the first robust regression algorithm that can handle the online features with adversarial data corruptions. It is nontrivial to consider online features and adversarial data corruption simultaneously in robust regression because 1) robust methods usually estimate data corruption based on the entire data, but online features make the data can only be partially accessible at one time; and 2) online feature selection methods can only select features based on uncorrupted data. Simply using robust regression and online feature selection methods sequentially makes the recovery result of coefficients worse, which is presented in our experiments in Section V. To address the above challenges, we proposed a new robust regression algorithm via online feature selection (*RoOFS*). The main contributions of our study are summarized as follows: 1) *design of an efficient algorithm to simultaneously address the problem of data corruption and online feature*. The algorithm *RoOFS* is proposed to recover the regression coefficients and uncorrupted set efficiently. Unlike using entire features, our approach alternately estimates the data corruption and selects the feature set via a robust online feature substitution method. 2) *theoretical analysis of the algorithm*. We prove that our method yields a solution with a restricted error bound compared to ground truth coefficients under the Subset Restricted Strong Convexity (*SRSC*) property. 3) *demonstration of empirical effectiveness and efficiency*. Our proposed algorithm was evaluated with 6 competing methods in both robust regression and online feature selection literatures. The results showed that our approach consistently outperforms existing methods in coefficients recovery and uncorrupted set estimation, delivering a competitive running time.

The reminder of this paper is organized as follows. Section II reviews the related work in robust regression model and

[1]https://www.imdb.com/

978-1-5386-9159-5/18/$31.00 ©2018 IEEE
DOI 10.1109/ICDM.2018.00199

online feature selection categories. Section III gives a formal problem formulation. The proposed *RoOFS* algorithm is presented in Section IV. In Section V, the experimental results are analyzed and the paper concludes with a summary of our work in Section VI.

II. Related Work

The work related to this paper is summarized in the categories of robust regression model and online feature selection as below.

A. Robust Regression Model

A large body of literature on robust regression and anomaly detection [2]–[6] has been established over the last few decades. Most of studies focus on handling stochastic noise in small amounts [7]–[10]; however, these methods cannot be applied to data that may exhibit malicious corruption [11]. To recover regression coefficients with adversarial data corruption, Chen et al. [11] proposed a robust algorithm based on trimmed inner product. McWilliams et al. [12] proposed a subsampling algorithm for large-scale corrupted linear regression, but their theoretical recovery boundaries are not close to the ground truth [13]. Some L_1 penalty based methods [14], [15] pursue strong recovery results for robust regression problem, but these methods depend on severe restrictions of the data distribution such as row-sampling from an incoherent orthogonal matrix [15]. Zhang et al. [16] proposed a distributed robust algorithm to handle the large-scale data set under adversarial data corruption.

Most research in this area requires the corruption ratio parameter, which is difficult to estimate under the assumption that the dataset can be adversarially attacked. For instance, She and Owen [17] rely on a regularization parameter to determine the size of the uncorrupted set based on soft-thresholding. Chen et al. [11] require the upper bound of the outliers number, which is also difficult to estimate when the data contain the adversarial data corruption. Bhatia et al. [13] proposed a hard-thresholding algorithm with a strong guarantee of coefficient recovery under mild assumption on input data. However, the corruption ratio parameter is required by the algorithm and its recovery error can be more than doubled in size if the parameter is far from the true value. Recently, Zhang et al. [18] proposed a heuristic hard-thresholding based methods that learns the optimal uncorrupted set. However, all these approaches are based on batch feature selection under the assumption that all features can be accessed entirely at any time, which is infeasible to apply in massive and fast growing feature set.

B. Online Feature Selection

Online feature selection methods [19]–[21] relaxes the requirement of batch selection and fit the scenarios that feature cannot be accessed entirely at one time. Statistical online feature selection algorithms [22]–[24] select features via certain statistical quantity such as mutual information, but these methods lack of specific objectives and usually have suboptimal solutions for some certain tasks. Optimization based approaches [25] use target oriented objective functions solved by some specific optimization techniques. These methods usually require the regression coefficient $\boldsymbol{\beta}$ be sparse, i.e., $\|\boldsymbol{\beta}\|_0 \leq \mu$. Grafting [26] and its variation [25] relax the hard constraint of feature set into L_1 penalty, which makes it a convex problem. However, the parameter of L_1 norm [27] is difficult to determine because the usual cross validation strategy is unavailable for the online feature selection scenario [28]. Yang et al. [29] proposed a limited-memory substitution algorithm based on the L_0 norm constraint. Although the hard constraint leads to an NP-hard problem, a theoretical guarantee for the error bound of their local optimal solution is provided. However, none of these online feature methods can handle the adversarial data corruption.

TABLE I: Math Notations

Notations	Explanations
$p, n \in \mathbb{R}$	number of entire features and data samples
$p_t \in \mathbb{R}$	number of features in t^{th} time interval
$\mu \in \mathbb{R}$	ratio of feature sparsity, where $\|\boldsymbol{\beta}\|_0 = \mu$
$X_t \in \mathbb{R}^{p_t \times n}$	data samples containing features in the t^{th} time interval
$X \in \mathbb{R}^{p \times n}$	data samples containing the entire features
$\boldsymbol{\beta}, \boldsymbol{\beta}^* \in \mathbb{R}^{p \times 1}$	estimated and ground truth regression coefficient
$\boldsymbol{u} \in \mathbb{R}^{n \times 1}$	corruption vector with adversarial values
$\boldsymbol{\varepsilon} \in \mathbb{R}^{n \times 1}$	dense noise vector, where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$
$\boldsymbol{y} \in \mathbb{R}^{n \times 1}$	response vector, where $\boldsymbol{y} = X^T\boldsymbol{\beta}^* + \boldsymbol{u} + \boldsymbol{\varepsilon}$
$\boldsymbol{r} \in \mathbb{R}^{n \times 1}$	residual vector, where $\boldsymbol{r} = \|\boldsymbol{y} - X^T\boldsymbol{\beta}\|$
$S \subseteq [n]$	estimated uncorrupted set
$S_* \subseteq [n]$	ground truth uncorrupted set, where $S_* = \overline{supp(\boldsymbol{u})}$
$\Psi, \Psi_* \subseteq [\mu]$	estimated and ground truth feature set

III. Problem Formulation

In this study, we consider the problem of robust regression with adversarial data corruption in the feature selection scenario in which only a few features are accessible at each time. Given data matrix $X_t \in \mathbb{R}^{p_t \times n}$ where p_t is the number of features available in the t^{th} time interval, and n are the number of data samples. The data matrix for all the time intervals is represented as $X = \{X_t\}_{t=1}^{T}$. We assume the corresponding response vector $\boldsymbol{y} \in \mathbb{R}^{n \times 1}$ is generated using the following model:

$$\boldsymbol{y} = X^T\boldsymbol{\beta}^* + \boldsymbol{u} + \boldsymbol{\varepsilon} \tag{1}$$

where $\boldsymbol{\beta}^*$ represents the μ-sparse ground truth coefficients of the regression model i.e., $\|\boldsymbol{\beta}^*\|_0 \leq \mu$ and $\boldsymbol{u}$ is the unbounded corruption vector introduced by adversarial data attacks. $\boldsymbol{\varepsilon} \in \mathbb{R}^{n \times 1}$ represents the additive dense noise, where $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$. Different from the corruption vector $\boldsymbol{u}$ that can be arbitrarily distributed, the dense noise ε_i follows normal distribution with zero mean and a relatively small variance σ. The notations used in this paper is summarized in Table I.

The goal of our problem is to learn a new robust regression problem with online feature selection, which is to recover the regression coefficients $\boldsymbol{\beta}^*$ and simultaneously determine the uncorrupted point set $\hat{S}$ with sequentially accessible features. The problem is formally defined as follows:

$$\begin{aligned} \hat{\boldsymbol{\beta}}, \hat{S} &= \arg\min_{\boldsymbol{\beta}, S} \|\boldsymbol{y}_S - X_S^T\boldsymbol{\beta}\|_2^2 \\ s.t. \;\; & S \subset [n], \; |S| \geq \mathcal{G}(\boldsymbol{\beta}), \; \|\boldsymbol{\beta}\|_0 \leq \mu \end{aligned} \tag{2}$$

Given a subset $S \subset [n]$, $\boldsymbol{y}_S$ restricts the row of $\boldsymbol{y}$ to indices in S and X_S signifies that the columns of X are restricted to indices in S. Therefore, we have $\boldsymbol{y}_S \in \mathbb{R}^{|S|\times 1}$ and $X_S \in \mathbb{R}^{p\times|S|}$. We use the notation $S_* = supp(\boldsymbol{u})$ to denote the ground truth set of uncorrupted points. Also, for any vector $\boldsymbol{v} \in \mathbb{R}^n$, the notation $\boldsymbol{v}_S$ represents the $|S|$-dimensional vector containing the components in S. The notation $\Psi = supp(\boldsymbol{\beta})$ is used to represent the set of selected features, resulting in $|\Psi| \leq \mu$. Similarly, we use X_Ψ to signify the rows of X are restricted to indices in Ψ and $X_{\Psi,S}$ to restrict both the rows and columns in set Ψ and S. The function $\mathcal{G}(\cdot)$ determines the size of uncorrupted data according to the regression coefficients $\boldsymbol{\beta}$, which is explained in Section IV. It is worth mentioning that the features of data matrix X in Equation (2) cannot be loaded entirely, but they can be accessed partially for each time interval. Therefore, the joint optimization of $\boldsymbol{\beta}$ and S in our problem are very challenging because it amounts to a non-convex discrete optimization problem under the assumption that data matrix X cannot be access entirely at one time.

IV. The Proposed Methodology

To solve the problem in Equation (2) efficiently with the guarantee on the strong recovery of regression coefficients, we propose a novel robust regression algorithm with online feature selection, *RoOFS*. The algorithm is only allowed to access part of features at each time, which are defined as the newly incoming features in our problem. One naive solution to handle the sequentially incoming features is to retain all the features in the memory and then apply traditional robust feature selection methods. However, the solution has two major drawbacks: 1) the feature set can be too large to be retained in the memory, and 2) the algorithm becomes slower and slower when the feature set increases. Therefore, we proposed a new "robust online substitution" method to decide the retained feature set based on an adaptively estimated corrupted set. The procedure of robust online substitution is defined as follows:

- Update coefficients of retained features Ψ based on the estimated uncorrupted set S as follows: $\boldsymbol{\beta}_\Psi := \boldsymbol{\beta}_\Psi - \eta\, X_{\Psi,S}^T(X_{\Psi,S}^T\boldsymbol{\beta}_\Psi - \boldsymbol{y}_S)$, where η is the step length.
- Retain the top μ largest (in magnitude) elements in $\boldsymbol{\beta}$ and set the rest to zero. Then all the non-zero features will be kept in the retained feature set Ψ.
- Compute the residual vector $\boldsymbol{r} \in \mathbb{R}^{n\times 1}$ with the updated coefficients $\boldsymbol{\beta}$, then estimate the uncorrupted feature set S via a thresholding operator $\mathcal{H}_\tau(\boldsymbol{r})$, where τ is the estimated size of uncorrupted set.

The procedure will be repeatedly executed until the residual vector $\boldsymbol{r}$ converges. The thresholding operator $\mathcal{H}_\tau(\cdot)$ is formally defined as follows:

Definition 1 (Thresholding Operator). *Defining $\varphi_{\boldsymbol{v}}^{-1}(i)$ as the position of the i^{th} element in input vector $\boldsymbol{v}$'s ascending order of magnitude and τ as the threshold parameter, the thresholding operator of $\boldsymbol{v}$ is defined as*

$$\mathcal{H}_\tau(\boldsymbol{v}) = \{i \in [n] : \varphi_{\boldsymbol{v}}^{-1}(i) \leq \tau\} \tag{3}$$

Algorithm 1: RoOFS Algorithm

Input: Corrupted training data $\{\boldsymbol{x}_i, y_i\}$, i = 1...n, feature ratio μ, tolerance ϵ
Output: solution $\hat{\boldsymbol{\beta}}$

$\boldsymbol{\beta}^0 \leftarrow \mathbf{0}$, $\Psi = \emptyset$, S_0 = [n], $t \leftarrow 0$, $k \leftarrow 0$
repeat
  Receive features X_{Ψ^k} from the pool $\bar{\Psi}$ with index set Ψ^k
  $\Psi = \Psi \cup \Psi^k$
  repeat
    $\boldsymbol{\beta}_\Psi^{t+1} \leftarrow \boldsymbol{\beta}_\Psi^t - \eta X_{\Psi,S_t}^T(X_{\Psi,S_t}^T\boldsymbol{\beta}_\Psi^t - \boldsymbol{y}_{S_t})$
    if $|\Psi| > \mu$ **then**
      $\Omega = \arg\min_{\Omega\in\Psi}\|\boldsymbol{\beta}_\Omega\|_1 \quad s.t.\ |\Omega| = |\Psi| - \mu$
      $\boldsymbol{\beta}_\Omega = 0$
      $\Psi = \Psi \setminus \Omega$
    $\boldsymbol{r} = |\boldsymbol{y} - X^T\boldsymbol{\beta}|$
    $S_{t+1} \leftarrow \mathcal{H}_\tau(\boldsymbol{r}^{t+1})$, where τ is the estimated uncorrupted size.
    $t \leftarrow t + 1$
  until $\|\boldsymbol{r}_{S_{t+1}}^{t+1} - \boldsymbol{r}_{S_t}^t\|_2 < \epsilon n$
  $k \leftarrow k + 1$
until *No more features;*
return $\boldsymbol{\beta}^{t+1}$, S_{t+1}

To estimate the uncorrupted set S, the thresholding operator $\mathcal{H}_\tau(\cdot)$ generally requires two inputs: residual vector $\boldsymbol{r}$ and the size of uncorrupted set τ. The residual vector $\boldsymbol{r}$ can be computed with coefficients $\boldsymbol{\beta}$ as follows:

$$\boldsymbol{r} = |\boldsymbol{y} - X^T\boldsymbol{\beta}| \tag{4}$$

For the size of uncorrupted set, two general cases are discussed. The first case is that the size can be estimated by users based on their prior knowledge on the data. For instance, if we know the data corruption happens rarely, then we can estimate the uncorrupted size as 95% of the entire data. However, it is hard to obtain prior knowledge on the data in the real-world. Thus, in the second case, we propose a method to adaptively estimate the uncorrupted size based on the residual vector $\boldsymbol{r}$. The method follows an intuition that when the coefficient $\boldsymbol{\beta}$ is close to $\boldsymbol{\beta}^*$, the residuals of uncorrupted samples are smaller than those of corrupted samples in strong possiblity. The intuition can be explained by the generative model in Equation (1), where the corrupted samples have the residual $\boldsymbol{r} \approx \boldsymbol{u} + \boldsymbol{\varepsilon}$, but the residual of uncorrupted samples only contains the white noise $\boldsymbol{\varepsilon}$.

The estimation of uncorrupted size can be formalized to solve the following problem:

$$\hat{\tau} := \underset{\lceil n/2\rceil<\tau\leq n}{\arg\max}\ \tau \quad s.t.\ \ r_{\varphi(\tau)} \leq \frac{2\tau r_{\varphi(\tau_o)}}{\tau_o}, \tau \in \mathbb{Z}^+ \tag{5}$$

where $r_{\varphi(k)}$ represents the k^{th} elements of residual vector $\boldsymbol{r}$ in ascending order of magnitude. The variable τ_o in the constraint is defined as an intermediate variable whose $r^2_{\varphi(\tau_o)}$ has the closest value to $\frac{\|\boldsymbol{r}_{\mathcal{H}_{\tau'}(\boldsymbol{r})}\|_2^2}{\tau'}$, where $\tau' = \tau - \lceil n/2\rceil$ and $\mathcal{H}_{\tau'}(\boldsymbol{r})$

represent the position set containing the smallest τ' elements in residual $\boldsymbol{r}$. The problem in Equation (5) can be solved by searching from n to $\lceil n/2 \rceil + 1$ and return the first value $\hat{\tau}$ which satisfies the constraint. It is important to note that the estimation method in Equation (5) requires the coefficients $\boldsymbol{\beta}$ to be close to $\boldsymbol{\beta}^*$. Thus, we optimize the uncorrupted set S along with coefficient $\boldsymbol{\beta}$ until both of them converge.

The details of *RoOFS* algorithm are presented in Algorithm 1. In Line 3, the algorithm receives data matrix X_{Ψ^k} with the incoming feature set Ψ^k at time k. The new feature set Ψ^k is combined into the retained feature set Ψ in Line 4. For each incoming feature set, the algorithm iteratively optimizes the regression coefficients $\boldsymbol{\beta}$ and the uncorrupted set S until the value of residual vector $\boldsymbol{r}^t_{S_t}$ is converged in Line 14. Specifically, in Line 6, regression coefficients $\boldsymbol{\beta}$ are updated to a better fit for the current estimated feature set Ψ and uncorrupted set S_t. In Line 8, feature set Ω that contains features with $|\Psi| - \mu$ smallest weights in $\boldsymbol{\beta}$ is selected. Then features in Ω are removed from the retained feature set Ψ and the weights in $\boldsymbol{\beta}_\Omega$ are reset to zero in Lines 9 and 10. The residual vector $\boldsymbol{r}$ is updated in Line 11, while the uncorrupted set S_{t+1} is estimated in Line 12 by the thresholding operator. Finally, both coefficients $\boldsymbol{\beta}$ and uncorrupted set S are returned in Line 17.

V. Experimental Results

In this section, we report the extensive experimental evaluation performed to verify the robustness, effectiveness of feature selection, and efficiency of the proposed method. All the experiments were conducted on a 64-bit machine with Intel(R) core(TM) quad-core processor (i7CPU@3.6GHz) and 32.0GB memory. Details of both the source code and sample data used in the experiment can be downloaded here[2].

A. Datasets and Metrics

To demonstrate the performance of our proposed method, comprehensive experiments are performed in synthetic datasets whose simulation samples were randomly generated according to the model in Equation (1). Specifically, we sample the regression coefficients $\boldsymbol{\beta}^* \in \mathbb{R}^p$ as a random unit norm vector with feature ratio constraint $\|\boldsymbol{\beta}\|_0 = \mu$. The data matrix X was drawn independently and identically distributed from $\boldsymbol{x}_i \sim \mathcal{N}(\mathbf{0}, I_p)$ and the uncorrupted response variables were generated as $y_i^* = \boldsymbol{x}_i^T \boldsymbol{\beta}^*$. The set of uncorrupted samples S was selected as a uniformly random τ_*-sized subset of $[n]$. The response vector $\boldsymbol{y}$ containing corrupted samples was generated as $\boldsymbol{y} = \boldsymbol{y}^* + \boldsymbol{u} + \boldsymbol{\varepsilon}$, where the corruption vector $\boldsymbol{u}$ was sampled from the uniform distribution $\big[-5\|\boldsymbol{y}^*\|_\infty, 5\|\boldsymbol{y}^*\|_\infty\big]$ and the additive dense noise was $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$. For the real-world data set, we applied our methods on the IMDb reviews data set for the review score prediction. The data set contains 50,000 popular movie reviews with the review score from 1 to 10 provided by the IMDb website. The adversarial data corruption vector $\boldsymbol{u}$ was appended to its original review score, where $\boldsymbol{u}$ was also sampled from the range $\big[-5\|\boldsymbol{y}^*\|_\infty, 5\|\boldsymbol{y}^*\|_\infty\big]$ randomly.

[2]https://goo.gl/C4HQjo

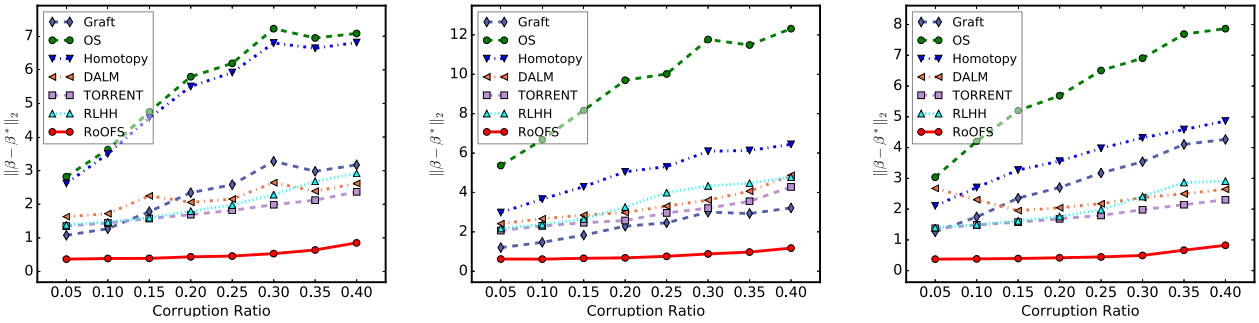
Following the setting in [13] [18], we measured the performance of the regression coefficients recovery using the standard L_2 error $e = \|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_2$, where $\hat{\boldsymbol{\beta}}$ represents the recovered coefficients for each method and $\boldsymbol{\beta}^*$ is the true regression coefficients. To validate the performance for corrupted set discovery, the F1 score is measured by comparing the discovered corrupted sets with the actual ones. Similarly, the F1 score is also used to measure the effectiveness of feature selection by comparing the selected feature set with actual ones. To compare the scalability of each method, the CPU running time for each of the competing methods was also measured.

B. Comparison Methods

The following methods are included in the performance comparison presented here: *Grafting* [26]. The *Grafting* method is an online version of L_1 regularization approach to selects features. *Online Substitution* (*OS*) [29] is a parameter-free online feature selection algorithm with limited-memory. Both *Grafting* and *OS* cannot handle the adversarial data corruption and train models without considering data corruption. We also compared our method to the robust regression methods [14] [15]. *Homotopy* and *DALM* are two L_1 based solvers that outperform other L_1 methods both in terms of recovery properties and running time [30]. A hard thresholding method, *TORRENT (abbreviated "TORR")* [13], developed for robust regression was also compared to our method. As the method requires a parameter for the corruption ratio, which is difficult to estimate in practice, we chose two versions of parameter settings: *TORR** and *TORR25*. *TORR** uses the true corruption ratio as its parameter, and *TORR25* applies parameter that is uniformly distributed across the range of $\pm 25\%$ off the true value. Another recently proposed heuristic hard thresholding method, *RLHH* [18], is also compared in our experiment. The method is a parameter-free approach, where the data corruption is estimated by a heuristic hard thresholding method. As all these robust methods are not designed for online feature selection, we run them individually in different feature batches and select features with largest μ weights in regression coefficients when $\|\boldsymbol{\beta}\|_0 = \mu$.

C. Recovery of regression coefficients

We selected 6 competing methods with which to evaluate the recovery performance of regression coefficients $\boldsymbol{\beta}$: *Grafting*, *OS*, *Homotopy*, *DALM*, *TORR*, *RLHH*. Figures 1(a) and 1(b) show the recovery performance for different feature numbers when the data size is fixed. The results show that 1) the proposed method, *RoOFS*, outperforms all the competing methods in all the setting of corruption ratios, and 2) The performance of *RoOFS* is very resistant to the corruption data because the error of *RoOFS* method increases much more slowly than others when corruption ratio increases from 5% to 40%. Figures 1(b) and 1(c) show that when data size increases, we have similar conclusion on the performance except the overall error is decreased since more data is applied.



(a) p=2K, n=1K, μ/p=20%, dense (b) p=4K, n=1K, μ/p=20%, dense (c) p=4K, n=2K, μ/p=20%, dense

Fig. 1: Performance on regression coefficients recovery for different corruption ratios in uniform distribution.

TABLE II: F1 Score on Performance of Feature Selection (cr=30%).

	p=10K, n=10K, dense						**p=20K, n=10K, dense**					
	10%	20%	30%	40%	50%	60%	10%	20%	30%	40%	50%	60%
Grafting	0.130	0.201	0.302	0.543	0.759	0.789	0.111	0.399	0.642	0.773	0.844	0.895
OS	0.706	0.649	0.626	0.643	0.810	0.678	0.611	0.611	0.606	0.689	0.836	**0.975**
Homotopy	0.116	0.217	0.304	0.406	0.498	0.667	0.109	0.202	0.417	0.625	0.750	0.833
DALM	0.130	0.219	0.309	0.418	0.516	0.597	0.114	0.215	0.390	0.404	0.502	0.603
TORR	0.275	0.297	0.368	0.446	0.525	0.647	0.320	0.338	0.395	0.458	0.535	0.623
RLHH	0.193	0.261	0.338	0.416	0.510	0.647	0.322	0.336	0.390	0.461	0.536	0.628
RoOFS	**0.911**	**0.910**	**0.876**	**0.870**	**0.895**	**0.891**	**0.876**	**0.842**	**0.832**	**0.848**	**0.881**	0.906

D. Performance of Feature Selection

We selected all the six competing methods to evaluate the performance of feature selection in different settings including data sizes, feature numbers, and dense noises. For each data setting, we chose different ratios of feature sparsity (also known as μ/p) ranging from 10% to 60%. Table II shows the following: 1) the F1 scores of *RoOFS* method is up to 69.2% better than other methods, especially when the feature ratio is less than 40%. 2) Although the F1 scores of most methods such as *Grafting* and *TORR* are above 0.6 when the ratio is larger than 50%, the performance degraded significantly when the ratio decreased to 10%. However, the F1 score of *RoOFS* method is constantly higher than 0.85 in all the ratios of features. 3) *OS* method is very competitive in the task of feature selection; however, it still has lower F1 scores in all the settings when the ratio is less than 50%.

E. Performance in real-world data

To evaluate the robustness of our proposed methods in a real-world dataset, we compared the performance of sentiment prediction in different corruption settings, ranging from 5% to 40%. The dataset was first proposed by Maas et al. [31] as a benchmark for sentiment analysis. It consists of movie reviews taken from IMDB. One key aspect of this dataset is that each movie review has several sentences. The 100,000 movie reviews are divided into three datasets: 25,000 labeled training instances, 25,000 labeled test instances and 50,000 unlabeled training instances. The unlabeled data were designed as the additional corruption to the dataset: the score of sentiment were random number between one to ten. Table III shows the mean absolute error of sentiment prediction in the IMDB datasets. From the result, we can conclude: 1) *RoOFS* method outperform all the other methods in different corruption settings. 2) Although the absolute error of the other methods such as *Homotopy* and *DALM* are above 4, the performance varied significantly when the ratio changed because these methods highly dependent on the parameters and it's hard to estimate the feature sparsity ratio and true corruption ratio in the real-world data. However, the performance of *RoOFS* method is constantly above 3.10 in all the ratios of corruption. 3) It is true that *OS* has a very competitive performance in all the corruption settings because the deviation of corruption is small, which is less than 50% from the labeled data. But the running time of *OS* is too high to train the data which has 10k features. 4) When increasing the corruption ratio, the absolute error of *RoOFS* method decreased.

VI. Conclusion

In this paper, a novel robust regression algorithm via online feature selection, *RoOFS*, is proposed to recover the regression coefficients and the uncorrupted set under the assumption that features cannot be accessed entirely at one time. To achieve this, we designed a robust online substitution method to alternately estimate the optimal uncorrupted set and substitute the retained feature set with newly updated features. Extensive

TABLE III: Mean Absolute Error of Sentiment Prediction.

	p=10K, n=10K					
	5%	10%	20%	30%	40%	Avg
Grafting	3.162	3.162	3.162	3.162	3.162	3.162
OS	3.111	3.078	3.078	3.113	3.108	3.098
Homotopy	4.157	3.925	3.894	3.828	3.540	3.869
DALM	3.573	3.311	3.523	3.459	3.252	3.424
TORR	5.090	3.928	4.596	5.147	4.218	4.596
RLHH	5.448	4.198	3.716	4.269	4.626	4.451
RoOFS	**3.074**	**3.073**	**3.072**	**3.070**	**3.066**	**3.071**

experiments on massive simulation data demonstrated that the proposed algorithm outperforms other competing methods in both effectiveness and efficiency.

REFERENCES

[1] R. W. Naylor, C. P. Lamberton, and P. M. West, "Beyond the like button: The impact of mere virtual presence on brand evaluations and purchase intentions in social media settings," *Journal of Marketing*, vol. 76, no. 6, pp. 105–120, 2012.

[2] C. Huang, D. Wang, and S. Zhu, "Where are you from: Home location profiling of crowd sensors from noisy and sparse crowdsourcing data," in *INFOCOM 2017-IEEE Conference on Computer Communications, IEEE*. IEEE, 2017, pp. 1–9.

[3] X. Teng, Y.-R. Lin, and X. Wen, "Anomaly detection in dynamic networks using multi-view time-series hypersphere learning," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, ser. CIKM '17. New York, NY, USA: ACM, 2017, pp. 827–836. [Online]. Available: http://doi.acm.org/10.1145/3132847.3132964

[4] X. Zhang, L. Zhao, A. P. Boedihardjo, C.-T. Lu, and N. Ramakrishnan, "Spatiotemporal event forecasting from incomplete hyper-local price data," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 2017, pp. 507–516.

[5] C. Huang, X. Wu, and D. Wang, "Crowdsourcing-based urban anomaly prediction system for smart cities," in *Proceedings of the 25th ACM international on conference on information and knowledge management*. ACM, 2016, pp. 1969–1972.

[6] B. Wang, X. Zhang, C.-T. Lu, and F. Chen, "Water disaggregation via shape features based bayesian discriminative sparse coding," *arXiv preprint arXiv:1808.08951*, 2018.

[7] P.-L. Loh and M. J. Wainwright, "High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity," in *Advances in Neural Information Processing Systems*, 2011, pp. 2726–2734.

[8] X. Wu, Y. Dong, C. Huang, J. Xu, D. Wang, and N. V. Chawla, "Uapd: Predicting urban anomalies from spatial-temporal data," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2017, pp. 622–638.

[9] X. Teng, M. Yan, A. M. Ertugrul, and Y.-R. Lin, "Deep into hypersphere: Robust and unsupervised anomaly discovery in dynamic networks," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*. International Joint Conferences on Artificial Intelligence Organization, 7 2018, pp. 2724–2730. [Online]. Available: https://doi.org/10.24963/ijcai.2018/378

[10] X. Zhang, L. Zhao, Z. Chen, and C. Lu, "Distributed self-paced learning in alternating direction method of multipliers," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, 2018, pp. 3148–3154. [Online]. Available: https://doi.org/10.24963/ijcai.2018/437

[11] Y. Chen, C. Caramanis, and S. Mannor, "Robust sparse regression under adversarial corruption," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, S. Dasgupta and D. Mcallester, Eds., vol. 28, no. 3. JMLR Workshop and Conference Proceedings, May 2013, pp. 774–782. [Online]. Available: http://jmlr.org/proceedings/papers/v28/chen13h.pdf

[12] B. McWilliams, G. Krummenacher, M. Lucic, and J. M. Buhmann, "Fast and robust least squares estimation in corrupted linear models," in *Advances in Neural Information Processing Systems*, 2014, pp. 415–423.

[13] K. Bhatia, P. Jain, and P. Kar, "Robust regression via hard thresholding," in *Advances in Neural Information Processing Systems*, 2015, pp. 721–729.

[14] J. Wright and Y. Ma, "Dense error correction via l1-minimization," *IEEE Trans. Inf. Theor.*, vol. 56, no. 7, pp. 3540–3560, Jul. 2010. [Online]. Available: http://dx.doi.org/10.1109/TIT.2010.2048473

[15] N. H. Nguyen and T. D. Tran, "Exact recoverability from dense corrupted observations via l1-minimization," *IEEE transactions on information theory*, vol. 59, no. 4, pp. 2017–2035, 2013.

[16] X. Zhang, L. Zhao, A. P. Boedihardjo, and C. T. Lu, "Online and distributed robust regressions under adversarial data corruption," in *2017 IEEE International Conference on Data Mining (ICDM)*, Nov 2017, pp. 625–634.

[17] Y. She and A. B. Owen, "Outlier detection using nonconvex penalized regression," *Journal of the American Statistical Association*, vol. 106, no. 494, pp. 626–639, 2011. [Online]. Available: http://www.jstor.org/stable/41416397

[18] X. Zhang, L. Zhao, A. P. Boedihardjo, and C.-T. Lu, "Robust regression via heuristic hard thresholding," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 2017, pp. 3434–3440. [Online]. Available: https://doi.org/10.24963/ijcai.2017/480

[19] W. Jiang, G. Er, Q. Dai, and J. Gu, "Similarity-based online feature selection in content-based image retrieval," *IEEE Transactions on Image Processing*, vol. 15, no. 3, pp. 702–712, 2006.

[20] J. Wang, P. Zhao, S. C. Hoi, and R. Jin, "Online feature selection and its applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 3, pp. 698–710, 2014.

[21] K. Yu, X. Wu, W. Ding, and J. Pei, "Scalable and accurate online feature selection for big data," *ACM Trans. Knowl. Discov. Data*, vol. 11, no. 2, pp. 16:1–16:39, Dec. 2016. [Online]. Available: http://doi.acm.org/10.1145/2976744

[22] J. Zhou, D. P. Foster, R. A. Stine, and L. H. Ungar, "Streamwise feature selection," *Journal of Machine Learning Research*, vol. 7, no. Sep, pp. 1861–1885, 2006.

[23] X. Wu, K. Yu, H. Wang, and W. Ding, "Online streaming feature selection," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 1159–1166.

[24] K. Yu, X. Wu, W. Ding, and J. Pei, "Towards scalable and accurate online feature selection for big data," in *Data Mining (ICDM), 2014 IEEE International Conference on*. IEEE, 2014, pp. 660–669.

[25] J. Zhu, N. Lao, and E. P. Xing, "Grafting-light: fast, incremental feature selection and structure learning of markov random fields," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 303–312.

[26] S. Perkins and J. Theiler, "Online feature selection using grafting," in *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ser. ICML'03. AAAI Press, 2003, pp. 592–599. [Online]. Available: http://dl.acm.org/citation.cfm?id=3041838.3041913

[27] S. Ryali and V. Menon, "Feature selection and classification of fmri data using logistic regression with l1 norm regularization," *NeuroImage*, vol. 47, p. S57, 2009.

[28] J. Wang, M. Wang, P. Li, L. Liu, Z. Zhao, X. Hu, and X. Wu, "Online feature selection with group structure analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 11, pp. 3029–3041, 2015.

[29] H. Yang, R. Fujimaki, Y. Kusumura, and J. Liu, "Online feature selection: A limited-memory substitution algorithm and its asynchronous parallel variation," in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 1945–1954. [Online]. Available: http://doi.acm.org/10.1145/2939672.2939881

[30] A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Fast l1-minimization algorithms and an application in robust face recognition: A review," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2010-13, Feb 2010. [Online]. Available: http://www2.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-13.html

[31] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011, pp. 142–150.