Statistica Sinica **35** (2025), 919-937
doi:https://doi.org/10.5705/ss.202023.0022

REGULARIZED ADAPTIVE HUBER MATRIX REGRESSION AND DISTRIBUTED LEARNING

Yue Wang[1], Wenqi Lu[2], Lei Wang[2], Zhongyi Zhu[3], Hongmei Lin*[4] and Heng Lian[1,5]

[1]*City University of Hong Kong,* [2]*Nankai University,* [3]*Fudan University,* [4]*Shanghai University of International Business and Economics and* [5]*City University of Hong Kong Shenzhen Research Institute*

Abstract: Matrix regression provides a powerful technique for analyzing matrix-type data, as exemplified by many contemporary applications. Despite the rapid advance, distributed learning for robust matrix regression to deal with heavy-tailed noises in the big data regime still remains untouched. In this paper, we first consider adaptive Huber matrix regression with a nuclear norm penalty, which enjoys insensitivity to heavy-tailed noises without losing the statistical accuracy. To further enhance the scalability in massive data applications, we employ the communication-efficient surrogate likelihood framework to develop distributed robust matrix regression, which can be efficiently implemented through the ADMM algorithms. Under only bounded $(1+\delta)$-th moment on the noise for some $\delta \in (0, 1]$, we provide upper bounds for the estimation error of the central estimator and the distributed estimator, and prove they can achieve the same rate as established with sub-Gaussian tails when only the second moment of noise exists. Numerical studies verify the advantage of the proposed method over existing methods in heavy-tailed noise settings.

Key words and phrases: Big data, communication-efficient, Huber loss, nuclear norm, robust matrix regression.

1. Introduction

Advances of modern technologies have made matrix-type data increasingly frequent in various applications, including image processing in computer vision, microarray gene study in medicine and asset allocation in economics (Rohde and Tsybakov, 2011; Senneret et al., 2016; Yang et al., 2016; Fan, Wang and Zhu, 2021). Although one intuitive idea is to reshape the matrix into a vector and apply popular vector-based regression methods, this may incur ultrahigh dimensionality and also destroy the inherent structure of matrix data such as the correlation between rows and columns. When considering matrix estimation, the rank plays an important part in constraining the model complexity, and the nuclear norm is a convex surrogate for rank (Candès and Tao, 2010). Indeed, the

*Corresponding author. E-mail: hongmeilin66@outlook.com

idea of imposing the nuclear norm penalty has been widely used in the literature. For example, Negahban and Wainwright (2011) and Koltchinskii, Lounici and Tsybakov (2011) studied the least squares matrix regression with nuclear norm penalty and derived the convergence rates under sub-Gaussian tails. However, the above methods are sensitive to outliers and the tails of the noise distribution due to the nature of least squares loss.

To tackle the issue of robustness, Huber (1973) proposed Huber loss which worked as a robust alternative to least squares loss. Subsequently, a variety of methods based on Huber loss were developed, including Lambert-Lacroix and Zwald (2011), Naseem, Togneri and Bennamoun (2012), and Loh (2017). These works share a common characteristic that the robustification parameter is treated as a constant based on the 95% asymptotic efficiency criterion. More recently, Sun, Zhou and Fan (2020) proposed the adaptive Huber regression that adapts the magnitude of the robustification parameter according to the sample size, dimension and moments of noises. It is worth mentioning that, although robust statistical tools like median/quantile regression are also frequently employed to cope with heavy-tailed noises, they differ from Huber-type methods in that the latter have a specific emphasis on robust *mean* regression. When the error is heterogeneous, the mean and the median can be considerably different and replacing mean regression with median regression incurs significant bias if our interest is on the conditional mean as we are focusing on in the current work. For more discussions on Huber-type methods, please refer to Fan, Li and Wang (2017), Chen and Zhou (2020), and Wang et al. (2021). While these methods achieve favorable results in the existing literature, they only work on vector regression and may be inefficient once handling matrix data.

On another direction of research development, with the availability of large-scale data, storing all data on a single machine is impracticable due to privacy issues, limited storage or communication costs. For example, different hospitals gather their own information individually and these original patient data cannot be shared to safeguard privacy. In the distributed learning, the full data are partitioned across multiple machines and each local machine only needs to store and process local data. Thus both storage and computation costs are functions of the local sample size n rather than functions of the total sample size N. Motivated by data parallelism, the divide-and-conquer strategy has been employed. The main idea is to calculate local estimates on local machines in parallel and then take the average to obtain the final estimate. Despite its low communication cost, the calculation may be expensive and some helpful structures may be sacrificed. For instance, Lee et al. (2017) and Battey et al. (2018) investigated the averaged debiased Lasso where the debiasing step is acknowledged to be computationally intensive and the resulting estimator is no longer sparse. To overcome such barrier, Jordan, Lee and Yang (2018) introduced a communication-efficient surrogate likelihood (CSL) method. Although it is easy to implement

and enjoys appealing statistical properties, the success of CSL method depends delicately on the smooth loss function (the original theoretical development requires the loss is thrice differentiable). Moreover, Chen et al. (2020) and Wang and Lian (2020) adopted different strategies to relax this smoothness condition, which were developed for vector regression. There are still few works concerning the distributed learning for matrix regression, let alone the robust matrix regression.

In this paper, we extend the idea of adaptive Huber vector regression to matrix regression, which can handle matrix covariates without loss of structural information. Inherited from the merits of adaptive Huber loss, our method is less sensitive to heavy-tailed errors and the adaptivity of the robustification parameter leads to the optimal trade-off between bias and robustness. To enhance the scalability of large-scale applications, we apply the communication-efficient distributed framework to the proposed robust matrix regression and developed an efficient implementation through ADMM based algorithms. Theoretically, we provide upper bounds for the estimation errors in terms of both Frobenius norm and nuclear norm in the presence of heavy-tailed noises. Specially, when $\delta = 1$, we show that the convergence rate of the central estimator using the full data and the distributed estimator can achieve the same rate established for matrix regression with light tails. In other words, the proposed method can enjoy the (approximate) unbiasness and robustness simultaneously. Last, it is rigorously proven that the the regularized estimators possess the low-rankness.

We note that matrix regression are closely related to multi-task learning, matrix completion and compressed sensing (Fan, Gong and Zhu, 2019). Therefore, the proposed robust matrix regression and associated distributed learning can be applied to these special models after some adjustments, which shows broad applications in practice.

The rest of the paper is organized as follows. Section 2 presents the proposed regularized adaptive Huber matrix regression. The associated algorithm and theoretical guarantees are developed. Section 3 introduces the distributed estimation for robust matrix regression, which can be solved by an ADMM based algorithm. The convergence rates for the distributed estimator are also provided. Section 4 shows an application to Beijing Air-Quality data. Section 5 concludes with some discussions. All the proofs and additional technical details are deferred to the Supplementary Material, as well as two simulation studies.

Notations. For a vector $\mathbf{v} = (v_1, \ldots, v_n)^\top$, $\|\mathbf{v}\|_q = (\sum_{i=1}^n |v_i|^q)^{1/q}$ denotes the l_q norm for $q \in [1, \infty)$. For a matrix $\mathbf{A}$, $\sigma_1(\mathbf{A}) \geq \sigma_2(\mathbf{A}) \geq \cdots \geq \sigma_m(\mathbf{A})$ denote the ordered singular values, $\|\mathbf{A}\|_{op} = \sigma_1(\mathbf{A})$ denotes the operator norm, $\|\mathbf{A}\|_F = \sqrt{\sum_{j=1}^m \sigma_j^2(\mathbf{A})}$ denotes the Frobenius norm and $\|\mathbf{A}\|_* = \sum_{j=1}^m \sigma_j(\mathbf{A})$ denotes the nuclear norm. $\mathrm{tr}(\mathbf{A})$ denotes the trace, $\mathrm{vec}(\mathbf{A})$ denotes the vectorization of $\mathbf{A}$, and $\lambda_{\min}(\mathbf{A})$ and $\lambda_{\max}(\mathbf{A})$ denote the minimum and maximum eigenvalues of $\mathbf{A}$,

respectively. Throughout the paper, C denotes a generic constant whose value may change even on the same line.

2. Regularized Adaptive Huber Matrix Regression

2.1. Model setting

Consider the matrix regression (also called trace regression) model

$$Y = \langle \mathbf{X}, \mathbf{\Theta}_0 \rangle + e,$$

where $Y \in \mathbb{R}$ is the response, $\mathbf{X} \in \mathbb{R}^{p\times q}$ is the covariate, $\mathbf{\Theta}_0 \in \mathbb{R}^{p\times q}$ is the unknown regression coefficient matrix, $\langle \mathbf{X}, \mathbf{\Theta}_0 \rangle = \mathrm{tr}(\mathbf{X}^\top \mathbf{\Theta}_0)$ is the inner product between matrices, and $e \in \mathbb{R}$ is the noise term. For ease of presentation, we omit the intercept and it can be added with some easy modifications that only involves more notational burden. Assume we have independent and identically distributed (i.i.d.) $\mathbf{X}_i$ and e_i for $i = 1, \ldots, n$. Define the empirical Huber loss as $L(\mathbf{\Theta}) = (1/n)\sum_{i=1}^n \ell_\tau(Y_i - \langle \mathbf{X}_i, \mathbf{\Theta}\rangle)$, where $\ell_\tau(\cdot)$ is the Huber loss of the form

$$\ell_\tau(u) = \begin{cases} u^2/2, & \text{if } |u| \le \tau, \\ \tau|u| - \tau^2/2, & \text{if } |u| > \tau, \end{cases}$$

with $\tau > 0$ being the robustification parameter. Intuitively, a larger τ leads to less bias but reduced robustness at the same time. Conversely, estimates with a small τ are typically more robust yet deviate more from the mean estimation. When τ goes to infinity, the Huber loss reduces to the least squares loss which possesses unbiasness at the expense of losing robustness. The regularized estimator of $\mathbf{\Theta}_0$ is defined as

$$\widehat{\mathbf{\Theta}} := \underset{\mathbf{\Theta}\in\mathbb{R}^{p\times q}}{\mathrm{argmin}}\, L(\mathbf{\Theta}) + \lambda\|\mathbf{\Theta}\|_*, \tag{2.1}$$

where $\lambda > 0$ is a regularization parameter. Here we adopt nuclear norm penalty to encourage a low rank estimate due to the fact that nuclear norm can be regarded as the convex surrogate of the rank, and enjoys desirable theoretical properties and tractable calculation.

2.2. Algorithm

We employ an ADMM algorithm to solve the above optimization problem. Specifically, by introducing auxiliary variables $\{r_i\}_{i=1}^n$ and $\mathbf{B}$, we can rewrite (2.1) equivalently as

$$\begin{aligned} &\min_{\mathbf{\Theta},\mathbf{B},r_i} \frac{1}{n}\sum_{i=1}^n \ell_\tau(r_i) + \lambda\|\mathbf{B}\|_* \\ &\text{subject to } \; r_i = Y_i - \langle \mathbf{X}_i, \mathbf{\Theta}\rangle, i = 1, \ldots, n, \text{ and } \mathbf{B} = \mathbf{\Theta}. \end{aligned} \tag{2.2}$$

To write it in matrix form, denote by $\mathbf{r} = (r_1, \ldots, r_n)^\top$, $\mathbf{y} = (Y_1, \ldots, Y_n)^\top$, $\bar{\mathbf{X}} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top$ with $\mathbf{x}_i = \mathrm{vec}(\mathbf{X}_i)$, and $\boldsymbol{\theta} = \mathrm{vec}(\boldsymbol{\Theta})$. Let $\ell_\tau(\mathbf{r})$ represent that $\ell_\tau(\cdot)$ is applied to each entry of $\mathbf{r}$ and then takes the sum. The augmented Lagrangian function is

$$L_\rho(\mathbf{r}, \mathbf{B}, \boldsymbol{\theta}; \mathbf{u}, \mathbf{V}) = \frac{1}{n}\ell_\tau(\mathbf{r}) + \lambda\|\mathbf{B}\|_* + \mathbf{u}^\top(\mathbf{r} - \mathbf{y} + \bar{\mathbf{X}}\boldsymbol{\theta}) + \langle \mathbf{V}, \mathbf{B} - \boldsymbol{\Theta}\rangle + \frac{\rho}{2}\|\mathbf{r} - \mathbf{y} + \bar{\mathbf{X}}\boldsymbol{\theta}\|_2^2 + \frac{\rho}{2}\|\mathbf{B} - \boldsymbol{\Theta}\|_F^2,$$

where $\mathbf{u} \in \mathbb{R}^n, \mathbf{V} \in \mathbb{R}^{p\times q}$ are dual variables associated with the constraints in (2.2) and $\rho > 0$ is the augmentation parameter. Then we have the following updates

$$\begin{aligned}
\mathbf{r}^{k+1} &= \operatorname*{argmin}_{\mathbf{r}} \frac{1}{n}\ell_\tau(\mathbf{r}) + \frac{\rho}{2}\|\mathbf{r} - \mathbf{y} + \bar{\mathbf{X}}\boldsymbol{\theta}^k + \mathbf{u}^k\rho^{-1}\|_2^2,\\
\mathbf{B}^{k+1} &= \operatorname*{argmin}_{\mathbf{B}} \frac{\rho}{2}\|\mathbf{B} - \boldsymbol{\Theta}^k + \mathbf{V}^k\rho^{-1}\|_F^2 + \lambda\|\mathbf{B}\|_*,\\
\boldsymbol{\theta}^{k+1} &= \operatorname*{argmin}_{\boldsymbol{\theta}} \frac{\rho}{2}\|\mathbf{r}^{k+1} - \mathbf{y} + \bar{\mathbf{X}}\boldsymbol{\theta} + \mathbf{u}^k\rho^{-1}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{\theta} - \mathbf{b}^{k+1} - \mathbf{v}^k\rho^{-1}\|_2^2,\\
\mathbf{u}^{k+1} &= \mathbf{u}^k + \rho(\mathbf{r}^{k+1} - \mathbf{y} + \bar{\mathbf{X}}\boldsymbol{\theta}^{k+1}),\\
\mathbf{V}^{k+1} &= \mathbf{V}^k + \rho(\mathbf{B}^{k+1} - \boldsymbol{\Theta}^{k+1}),
\end{aligned} \tag{2.3}$$

where $\mathbf{b} = \mathrm{vec}(\mathbf{B})$ and $\mathbf{v} = \mathrm{vec}(\mathbf{V})$. The algorithm is summarized in Algorithm 1 and the detailed derivation is deferred to the Supplementary Material. Note that the update of $\mathbf{B}$ does not depend on $\mathbf{r}$ and vice versa. Thus this ADMM algorithm indeed has two blocks, one is the update of $(\mathbf{r}, \mathbf{B})$ and the other is the update of $\boldsymbol{\theta}$. Consequently, we have global convergence to the minimizer of (2.1). We present the following convergence result from Boyd, Parikh and Chu (2011) for completeness.

Proposition 1. *The iterates $\boldsymbol{\Theta}^K$ produced by Algorithm 1 converges to the minimizer of (2.1) $\widehat{\boldsymbol{\Theta}}$ as $K \to \infty$.*

2.3. Theoretical properties

To establish an upper bound for the error of $\widehat{\boldsymbol{\Theta}}$ in (2.1), we impose the following assumptions.

(A1) The true regression coefficient matrix $\boldsymbol{\Theta}_0$ has rank $r \le \min(p, q)$.

(A2) The vectorized covariate $\mathbf{x} = \mathrm{vec}(\mathbf{X})$ is sub-Gaussian. That is, for any $\boldsymbol{\alpha} \in \mathbb{R}^{pq}$ and $t > 0$, there exits some positive constant c_0 such that $\mathbb{P}(|\boldsymbol{\alpha}^\top\mathbf{x}| \ge t) \le 2\exp(-t^2\|\boldsymbol{\alpha}\|_2^2/c_0^2)$.

(A3) Let $\boldsymbol{\Sigma} = \mathbb{E}(\mathbf{x}\mathbf{x}^\top)$. There exist positive constants c_l and c_u such that $c_l \le \lambda_{\min}(\boldsymbol{\Sigma}) \le \lambda_{\max}(\boldsymbol{\Sigma}) \le c_u$.

Algorithm 1 ADMM-based algorithm for solving the regularized adaptive Huber matrix regression.

Require: $\{(Y_i, \mathbf{X}_i)\}_{i=1}^n$.

Initialize $(\mathbf{r}^0, \mathbf{B}^0, \boldsymbol{\theta}^0, \mathbf{u}^0, \mathbf{V}^0)$.

for $k = 0, 1, 2, \ldots, K-1$ **do**

Calculate $h_i = (\mathbf{y} - \bar{\mathbf{X}}\boldsymbol{\theta}^k - \mathbf{u}^k/\rho)_i$;

Update

$$r_i^{k+1} = \begin{cases} n\rho h_i/(1+n\rho), & \text{if } |h_i| \le \tau(1+n\rho)/(n\rho), \\ \{h_i - \tau/(n\rho)\}_+ - \{-h_i - \tau/(n\rho)\}_+, & \text{otherwise;} \end{cases}$$

Calculate SVD[1]: $\boldsymbol{\Theta}^k - \mathbf{V}^k/\rho = \sum_{j=1}^{\min(p,q)} \omega_j \mathbf{a}_j \mathbf{c}_j^{\mathrm{T}}$;

Update $\mathbf{B}^{k+1} = \sum_{j=1}^{\min(p,q)} (\omega_j - \lambda/\rho)_+ \mathbf{a}_j \mathbf{c}_j^{\mathrm{T}}$;

Update

$$\boldsymbol{\theta}^{k+1} = (\bar{\mathbf{X}}^\top \bar{\mathbf{X}} + \mathbf{I}_{pq})^{-1} \left\{ \bar{\mathbf{X}}^\top \left(-\mathbf{r}^{k+1} + \mathbf{y} - \frac{\mathbf{u}^k}{\rho} \right) + \mathbf{b}^{k+1} + \frac{\mathbf{v}^k}{\rho} \right\};$$

Update $\mathbf{u}^{k+1} = \mathbf{u}^k + \rho(\mathbf{r}^{k+1} - \mathbf{y} + \bar{\mathbf{X}}\boldsymbol{\theta}^{k+1})$;

Update $\mathbf{V}^{k+1} = \mathbf{V}^k + \rho(\mathbf{B}^{k+1} - \boldsymbol{\Theta}^{k+1})$;

end for

return $\boldsymbol{\theta}^K$.

[1]SVD denotes the singular value decomposition.

(A4) The noise e satisfies $\mathbb{E}(e|\mathbf{X}) = 0$ and $\mathbb{E}(|e|^{1+\delta}|\mathbf{X}) \le \sigma_\delta$ almost surely for some $0 < \delta \le 1$.

(A1) is imposed to ensure the true coefficient matrix has a low-rank structure. Our upper bound to follow becomes smaller with a decrease of r as expected. (A2) is used to bound some operator norms involving the covariates and derive exponential-type concentration inequalities. (A3) assumes the eigenvalues of $\boldsymbol{\Sigma}$ are bounded, which is a mild assumption even typically used in high-dimensional linear regression. (A4) allows for the heavy-tailed noise with only finite $(1+\delta)$-th moment for some $\delta \in (0, 1]$, which is milder than common sub-Gaussian assumptions. It also allows conditional heteroscedastic models, where e can depend on $\mathbf{X}$. Similar assumptions appeared in Sun, Zhou and Fan (2020) and Chen and Zhou (2020).

To facilitate the theoretical analysis, we introduce some additional notations. Let $\boldsymbol{\Theta}_0 = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ be the singular value decomposition (SVD) of $\boldsymbol{\Theta}_0$, where $\mathbf{U} \in \mathbb{R}^{p\times r}$ and $\mathbf{V} \in \mathbb{R}^{q\times r}$ are orthogonal matrices, and $\mathbf{D}$ is a diagonal matrix. We define two subspaces of $\mathbb{R}^{p\times q}$ as follows:

$$\mathcal{M} = \{\mathbf{A} \in \mathbb{R}^{p\times q} : \mathrm{row}(\mathbf{A}) \subseteq \mathrm{col}(\mathbf{V}), \mathrm{col}(\mathbf{A}) \subseteq \mathrm{col}(\mathbf{U})\},$$
$$\mathcal{N} = \{\mathbf{A} \in \mathbb{R}^{p\times q} : \mathrm{row}(\mathbf{A}) \perp \mathrm{col}(\mathbf{V}), \mathrm{col}(\mathbf{A}) \perp \mathrm{col}(\mathbf{U})\},$$

where row$(\cdot)$ and col$(\cdot)$ denote the row space and column space, respectively. As stated in Negahban et al. (2012), the nuclear norm is decomposable with respect to a pair $(\mathcal{M}, \mathcal{N})$ which means that $\|\mathbf{A}_1+\mathbf{A}_2\|_* = \|\mathbf{A}_1\|_*+\|\mathbf{A}_2\|_*$ for any $\mathbf{A}_1 \in \mathcal{M}$ and $\mathbf{A}_2 \in \mathcal{N}$. Let $\mathcal{P}_{\mathcal{N}} : \mathbb{R}^{p\times q} \to \mathbb{R}^{p\times q}$ denote the projection onto the subspace $\mathcal{N}$, $\mathbf{\Delta}_{r^c} = \mathcal{P}_{\mathcal{N}}\mathbf{\Delta}$ and $\mathbf{\Delta}_r = \mathbf{\Delta} - \mathbf{\Delta}_{r^c}$ for any matrix $\mathbf{\Delta} \in \mathbb{R}^{p\times q}$. We consider the restricted set

$$\mathbb{C} := \{\mathbf{\Delta} \in \mathbb{R}^{p\times q} : \|\mathbf{\Delta}_{r^c}\|_* \le 3\|\mathbf{\Delta}_r\|_*\}.$$

In fact, as long as we assume that $\lambda \ge 2\|\nabla L(\mathbf{\Theta}_0)\|_{op}$ holds, $\widehat{\mathbf{\Delta}} := \widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0$ will fall into the above nuclear norm cone $\mathbb{C}$; see Lemma 6 in the Supplementary Material. Then we can establish the restricted strong convexity property (Lem. 5 in the Supplementary Material) over $\mathbb{C} \cap \{\|\mathbf{\Delta}\|_F \le \gamma\}$ for some $\gamma > 0$, which plays a pivotal role in deriving the error bounds of the regularized estimator. In the statement of Theorem 1 below, for notational simplicity, we define $v_\delta = (\sigma_\delta)^{1/(1+\delta)}$, where σ_δ is the constant as given in Assumption (A4).

Before stating the theorem, let us make it clear that in our theoretical study r, p, q are allowed to diverge with n (one can think of these quantities as functions of n such that when we say $n \to \infty$, we also mean we possibly have $p \to \infty$ at the same time, for example). Several other quantities that appeared in the assumptions, including c_0, c_l, c_u, are treated as constants. In the statements and proofs of our theoretical results, C denotes a generic positive constants that can depends on c_0, c_l, c_u, but not on r, p, q.

Theorem 1. *Assume conditions* (A1)–(A4) *hold,* $n \ge Cr(p+q)\log n$, $\tau = Cv_\delta[n/\{(p+q)\log n\}]^{1/(1+\delta)}$ *and* $\lambda \ge Cv_\delta\{(p+q)\log n/n\}^{\delta/(1+\delta)}$ *for some sufficiently large constant* C. *Then with probability at least* $1-n^{-C}$ *for some* $C > 0$, *we have*

$$\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_F \le C\sqrt{r}\lambda, \quad \|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_* \le Cr\lambda.$$

Theorem 1 provides upper bounds for the estimation error in terms of both Frobenius norm and nuclear norm. The error bounds depend on $r(p+q)$ rather than the ambient parameter pq. Specifically, when the noises have a finite second moment ($\delta = 1$), the proposed estimator can achieve the convergence rate established in Negahban and Wainwright (2011) for matrix regression models with sub-Gaussian tails up to a $\log n$ factor ($\log n$ factor can be removed as discussed in Rmk. 1 below). When the noises are more heavy-tailed ($0 < \delta < 1$), the proposed estimator achieves a slower rate than that with $\delta = 1$. Our result thus provides a theoretical justification of using Huber's loss instead of squared loss when the noise is heavy-tailed.

We also note that τ adapts to the sample size, dimension and moments of the noise and reveals an optimal tradeoff between bias and robustness. As described in Section 2.1, τ controls the blending of least squares loss and absolute value loss. A large τ reduces the bias but compromises the robustness. Specifically, the convergence rate is determined by the order of $\lambda \ge C\|\nabla L(\mathbf{\Theta}_0)\|_{op}$ and Lemma 6

in the Supplementary Material reveals that

$$\|\nabla L(\mathbf{\Theta}_0)\|_{op} \leq C\sqrt{\frac{\sigma_\delta \tau^{1-\delta}(p+q)\log n}{n}} + C\frac{\tau(p+q)\log n}{n} + C\sigma_\delta \tau^{-\delta}.$$

If we assume $\tau = Cv_\delta[n/\{(p+q)\log n\}]^{1/(1+\delta)}$, the above three terms have the same order and thus yield the upper bound

$$\|\nabla L(\mathbf{\Theta}_0)\|_{op} \leq Cv_\delta \left\{\frac{(p+q)\log n}{n}\right\}^{\delta/(1+\delta)},$$

which implies the bound in Theorem 1. When $\delta = 1$, this is consistent with the rate $\sqrt{r(p+q)/n}$ established in Negahban and Wainwright (2011) up to a $\log n$ factor ($\log n$ factor can be removed as discussed in Rmk. 1 below).

Remark 1. In Theorem 1, λ should be set to have at least the same order as $\|\nabla L(\mathbf{\Theta}_0)\|_{op}$. Moving to the proof for the upper bound of $\|\nabla L(\mathbf{\Theta}_0)\|_{op}$ (Lem. 6 in the Supplementary Material), if we take $z = C(p+q)$ in equation (S2.15) rather than $z = C(p+q)\log n$, we can further get the following result.

Assume conditions (A1)–(A4) hold, $n \geq Cr(p+q)$, $\tau = Cv_\delta\{n/(p+q)\}^{1/(1+\delta)}$ and $\lambda \geq Cv_\delta\{(p+q)/n\}^{\delta/(1+\delta)}$ for some sufficiently large constant C. Then with probability at least $1 - e^{-C(p+q)}$, we have

$$\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_F \leq C\sqrt{r}\lambda, \quad \|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_* \leq Cr\lambda.$$

That is, the above bounds hold with probability approaching one provided that p, q diverge to infinity. We keep the $\log n$ term for technical convenience to cover the settings of fixed p, q. Similar arguments also apply to other theorems below.

The following result shows that the nuclear norm-regularized estimator (2.1) has a rank that is of the same order as the unknown true rank.

Theorem 2. *Assume the same conditions as in Theorem* 1. *Then with probability approaching one as n goes to infinity, we have*

$$\text{rank}(\widehat{\mathbf{\Theta}}) \leq Cr.$$

Theorem 2 points out that the rank of the regularized estimator is of the order $O(r)$ with an overwhelming probability. Although $\widehat{\mathbf{\Theta}}$ is often low-rank in practice as long as the regularization parameter λ is sufficiently large, we provide strict theoretical guarantee on the low-rankness for regularized robust matrix estimator.

2.4. Comparison with existing works

Past work by Negahban and Wainwright (2011) (NW11) studied the matrix estimation problem with least squares loss and nuclear norm penalty and derived

Table 1. Summary of comparison with existing works.

	Loss	Convergence rate		Order of rank
		Frobenius norm	Nuclear norm	
NW11	quadratic	$\sqrt{\frac{r(p+q)}{n}}$	$\times$	$\times$
ZL14	quadratic	$\times$	$\times$	$\times$
EV18	median/Huber	$\sqrt{r}pq\sqrt{\frac{\log(p+q)}{n(p\wedge q)}}$	$rpq\sqrt{\frac{\log(p+q)}{n(p\wedge q)}}$	$\times$
Ours	Huber	$\sqrt{r}\left(\frac{p+q}{n}\right)^{\delta/(1+\delta)}$	$r\left(\frac{p+q}{n}\right)^{\delta/(1+\delta)}$	$O(r)$

the upper bound $\sqrt{r(p+q)/n}$ in terms of the Frobenius norm under sub-Gaussian tails and restricted strong convexity. Besides, Zhou and Li (2014) (ZL14) also covered this setting but did not prove theoretical guarantee in terms of the convergence rate. Due to the nature of quadratic loss, their methods are sensitive to outliers and the tails of the noise distribution. Moreover, they did not prove the low-rankness of the regularized estimator.

Elsener and van de Geer (2018) (EV18) considered robust nuclear norm penalized estimators through using the absolute deviation/median loss and the Huber loss. However, for the Huber loss, they treat the robustification parameter τ as fixed which does not trade off bias versus robustness. Their assumption that the distribution of noises is symmetric around zero is quite stringent. If this condition is violated, the estimator using median loss can only estimate the conditional median rather than the conditional mean. Unlike EV18, we allow τ to diverge, thereby waiving the requirement of symmetric noise distribution. Moreover, if $p \asymp q$ and $p(\log p)^{(1+\delta)/(3+\delta)} \gg n^{(1-\delta)/(3+\delta)}$, their convergence rates $\sqrt{r}pq\sqrt{\log(p+q)/\{n(p\wedge q)\}}$ in Frobenius norm and $rpq\sqrt{\log(p+q)/\{n(p\wedge q)\}}$ in nuclear norm are slower than ours. In other words, our rate is better if the dimension of the matrix is sufficiently large. In particular, if $\delta = 1$, our rate is always better, although requiring a stronger assumption of finite second moment for error.

We summarize the above comparison in Table 1 and also applicability of several methods under different error distribution assumptions in Table 2. Theoretically, Negahban and Wainwright (2011) assumed the error distribution is Gaussian or sub-Gaussian. We assumed finite moment assumption up to the second moment in this work. Elsener and van de Geer (2018) does not require any moment assumption, but in order for the estimation target to be the conditional mean, the error distribution needs to be symmetric.

Table 2. Applicability of different methods for conditional mean regression.

		Quadratic (NW11, ZL14)	Median (EV18)	Huber (Ours)
Light tails	Symmetric	✓	✓	✓(large τ)
	Asymmetric	✓	×	✓(large τ)
Heavy tails	Symmetric	×	✓	✓(small τ)
	Asymmetric	×	×	✓(small τ)

3. Distributed Estimation for Regularized Adaptive Huber Matrix Regression

In this section, we apply the communication-efficient surrogate likelihood (CSL) method (Jordan, Lee and Yang, 2018) to the robust matrix regression studied in Section 2, to enhance the scalability of dealing with large-scale data. Let $\{Z_{ij} : i = 1, \ldots, n, j = 1, \ldots, m\}$ denote $N = nm$ samples that are stored on m machines (in the regression setting $Z_{ij} = (Y_{ij}, \mathbf{X}_{ij})$), where we indeed assume each local machine $\mathcal{M}_j$ stores an equal number of subsamples n for simplicity. We briefly review CSL and then illustrate the distributed Huber matrix regression.

3.1. Communication-efficient distributed framework

We adopt CSL to develop distributed robust matrix regression, owing to the appealing property that CSL can achieve the tradeoff between communication cost and statistical efficiency, and is easy to implement. Assume θ_0 is the unknown parameter of interest, loss function ℓ is differentiable and $L_j(\theta) = (1/n)\sum_{i=1}^{n} \ell(\theta, Z_{ij})$ is the local loss on machine $\mathcal{M}_j$. Given an initial estimator $\widehat{\theta}$ of θ_0 that is typically computed on a small part of the entire data, Jordan, Lee and Yang (2018) proposed to estimate the global loss $L(\theta) = (1/N)\sum_{i=1}^{n}\sum_{j=1}^{m} \ell(\theta; Z_{ij})$ by

$$L(\theta) \approx L_1(\theta) - \theta^\top\{\nabla L_1(\widehat{\theta}) - \nabla L(\widehat{\theta})\} + \text{terms independent of } \theta, \tag{3.1}$$

which is essentially motivated by second-order Taylor's expansion. We note that $\nabla L(\widehat{\theta}) = (1/m)\sum_{j=1}^{m} \nabla L_j(\widehat{\theta})$ is the only term that involves data other than those on $\mathcal{M}_1$. The distributed learning procedure is implemented as follows.

- First, $\mathcal{M}_1$ transmits the initial estimator to local machines, and each $\mathcal{M}_j$ calculates the local gradient $\nabla L_j(\widehat{\theta})$ and sends it back to $\mathcal{M}_1$;
- Second, $\mathcal{M}_1$ computes the updated θ by using the surrogate loss $L_1(\theta) - \theta^\top\{\nabla L_1(\widehat{\theta}) - \nabla L(\widehat{\theta})\}$.

As discussed in Jordan, Lee and Yang (2018), the communication cost is $O(m\widetilde{p})$ with $\widetilde{p}$ denoting the dimension of θ. Compared with collecting all data on a single machine with communication cost $O(mn\widetilde{p})$, CSL significantly reduces the communication cost. Technically, Jordan, Lee and Yang (2018) requires ℓ is at

least thrice differentiable and thus does not apply to Huber's loss directly. We need to construct a proof that heavily relies on empirical process techniques making it sufficiently different.

3.2. Distributed adaptive Huber matrix regression

We integrate the idea of CSL with the adaptive Huber matrix regression in the following. Given samples $\{(Y_{ij}, \mathbf{X}_{ij})\}_{i=1}^{n}{}_{j=1}^{m}$, the local and global Huber loss functions are of the form

$$L_j(\boldsymbol{\Theta}) = \frac{1}{n}\sum_{i=1}^{n} \ell_\tau(Y_{ij} - \langle \mathbf{X}_{ij}, \boldsymbol{\Theta}\rangle), \quad j = 1, \ldots, m,$$
$$L(\boldsymbol{\Theta}) = \frac{1}{m}\sum_{j=1}^{m} L_j(\boldsymbol{\Theta}) = \frac{1}{N}\sum_{i=1}^{n}\sum_{j=1}^{m} \ell_\tau(Y_{ij} - \langle \mathbf{X}_{ij}, \boldsymbol{\Theta}\rangle).$$

Adapted from (3.1), we can construct the surrogate loss function as

$$\widetilde{L}(\boldsymbol{\Theta}) = L_1(\boldsymbol{\Theta}) - \langle \boldsymbol{\Theta}, \nabla L_1(\widehat{\boldsymbol{\Theta}}) - \nabla L(\widehat{\boldsymbol{\Theta}})\rangle,$$

where $\widehat{\boldsymbol{\Theta}}$ is an initial estimator of the true coefficient matrix $\boldsymbol{\Theta}_0$, $\nabla L_1(\widehat{\boldsymbol{\Theta}}) = -(1/n)\sum_{i=1}^{n} \ell_\tau'(Y_{i1} - \langle \mathbf{X}_{i1}, \widehat{\boldsymbol{\Theta}}\rangle)\mathbf{X}_{i1}$ and $\nabla L(\widehat{\boldsymbol{\Theta}}) = -(1/N)\sum_{i=1}^{n}\sum_{j=1}^{m} \ell_\tau'(Y_{ij} - \langle \mathbf{X}_{ij}, \widehat{\boldsymbol{\Theta}}\rangle)\mathbf{X}_{ij}$ with $\ell_\tau'(u) = \operatorname{sign}(u)\min(|u|, \tau)$. The distributed estimator is defined as

$$\widetilde{\boldsymbol{\Theta}} := \operatorname*{argmin}_{\boldsymbol{\Theta}\in\mathbb{R}^{p\times q}} \widetilde{L}(\boldsymbol{\Theta}) + \lambda\|\boldsymbol{\Theta}\|_*. \tag{3.2}$$

We again adapt the ADMM algorithm to solve the distributed estimator $\widetilde{\boldsymbol{\Theta}}$. By introducing auxiliary variables $\{r_i\}_{i=1}^{n}$ and $\mathbf{B}$, (3.2) can be reformulated as

$$\min_{\boldsymbol{\Theta},\mathbf{B},r_i} \frac{1}{n}\sum_{i=1}^{n} \ell_\tau(r_i) - \langle \boldsymbol{\Theta}, \nabla L_1(\widehat{\boldsymbol{\Theta}}) - \nabla L(\widehat{\boldsymbol{\Theta}})\rangle + \lambda\|\mathbf{B}\|_*$$
$$\text{subject to } r_i = Y_{i1} - \langle \mathbf{X}_{i1}, \boldsymbol{\Theta}\rangle, i = 1, \ldots, n, \text{ and } \mathbf{B} = \boldsymbol{\Theta}.$$

Denote by $\mathbf{r} = (r_1, \ldots, r_n)^\top$, $\mathbf{y}_1 = (Y_{11}, \ldots, Y_{n1})^\top$, $\bar{\mathbf{X}}_1 = (\mathbf{x}_{11}, \ldots, \mathbf{x}_{n1})^\top$ with $\mathbf{x}_{i1} = \operatorname{vec}(\mathbf{X}_{i1})$, $\boldsymbol{\theta} = \operatorname{vec}(\boldsymbol{\Theta})$ and $\mathbf{g} = \operatorname{vec}(\nabla L_1(\widehat{\boldsymbol{\Theta}}) - \nabla L(\widehat{\boldsymbol{\Theta}}))$. The augmented Lagrangian function is

$$L_\rho(\mathbf{r}, \mathbf{B}, \boldsymbol{\theta}; \mathbf{u}, \mathbf{V}) = \frac{1}{n}\ell_\tau(\mathbf{r}) - \boldsymbol{\theta}^\top\mathbf{g} + \lambda\|\mathbf{B}\|_* + \mathbf{u}^\top(\mathbf{r} - \mathbf{y}_1 + \bar{\mathbf{X}}_1\boldsymbol{\theta})$$
$$+\langle \mathbf{V}, \mathbf{B} - \boldsymbol{\Theta}\rangle + \frac{\rho}{2}\|\mathbf{r} - \mathbf{y}_1 + \bar{\mathbf{X}}_1\boldsymbol{\theta}\|_2^2 + \frac{\rho}{2}\|\mathbf{B} - \boldsymbol{\Theta}\|_F^2.$$

Then we have the following updates:

$$\mathbf{r}^{k+1} = \operatorname*{argmin}_{\mathbf{r}} \frac{1}{n}\ell_\tau(\mathbf{r}) + \frac{\rho}{2}\|\mathbf{r} - \mathbf{y}_1 + \bar{\mathbf{X}}_1\boldsymbol{\theta}^k + \mathbf{u}^k\rho^{-1}\|_2^2,$$

$$\mathbf{B}^{k+1} = \underset{\mathbf{B}}{\operatorname{argmin}} \frac{\rho}{2}\|\mathbf{B} - \mathbf{\Theta}^k + \mathbf{V}^k\rho^{-1}\|_F^2 + \lambda\|\mathbf{B}\|_*,$$
$$\boldsymbol{\theta}^{k+1} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \frac{\rho}{2}\|\mathbf{r}^{k+1} - \mathbf{y}_1 + \bar{\mathbf{X}}_1\boldsymbol{\theta} + \mathbf{u}^k\rho^{-1}\|_2^2 + \frac{\rho}{2}\|\boldsymbol{\theta} - \mathbf{b}^{k+1} - \mathbf{v}^k\rho^{-1}\|_2^2 - \boldsymbol{\theta}^\top\mathbf{g},$$
$$\mathbf{u}^{k+1} = \mathbf{u}^k + \rho(\mathbf{r}^{k+1} - \mathbf{y}_1 + \bar{\mathbf{X}}_1\boldsymbol{\theta}^{k+1}),$$
$$\mathbf{V}^{k+1} = \mathbf{V}^k + \rho(\mathbf{B}^{k+1} - \mathbf{\Theta}^{k+1}),$$

where $\mathbf{b} = \operatorname{vec}(\mathbf{B})$ and $\mathbf{v} = \operatorname{vec}(\mathbf{V})$. We note that all updates have closed-form solutions and can be solved efficiently. Indeed, the above updates naturally reduce to (2.3) when $\mathbf{g} = \mathbf{0}$. The detailed implementation of the distributed learning procedure is summarized in Algorithm 2. This algorithm is also a two-block ADMM algorithm and thus it also has the global convergence property as in Proposition 1.

Next we proceed with the theoretical analysis. As mentioned above, since ℓ_τ is not thrice differentiable and the matrix covariate has a more complex structure, the proof of Jordan, Lee and Yang (2018) cannot be applied directly. Although the surrogate loss involves the local loss L_1, we need to derive error bounds that correspond to that of the central estimator using the global loss L, which seems quite challenging. The key techniques used to address these issues are outlined in Lemma 7 in the Supplementary Material, where we resort to the covering argument and Bernstein's inequality to bound $\|\nabla L(\mathbf{\Theta}) - \nabla L(\mathbf{\Theta}_0) - \mathbb{E}\nabla L(\mathbf{\Theta}) + \mathbb{E}\nabla L(\mathbf{\Theta}_0)\|_{op}$ uniformly over the restricted set $\Omega = \{\mathbf{\Theta} \in \mathbb{R}^{p\times q} : \|\mathbf{\Theta} - \mathbf{\Theta}_0\|_F \le a_n, \operatorname{rank}(\mathbf{\Theta}) \le Cr\}$ with a_n given in (A5) below, rather than applying the third-order Taylor's expansion as in Jordan, Lee and Yang (2018). We impose the following additional assumptions.

(A5) The initial estimator $\widehat{\mathbf{\Theta}}$ satisfies $\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_F \le a_n$ and $\operatorname{rank}(\widehat{\mathbf{\Theta}}) \le Cr$, with some positive sequence $a_n = o(1)$ as $n \to \infty$.

(A5) characterizes the restrictions on the initial estimator. Regarding the specific choice of $\widehat{\mathbf{\Theta}}$, a natural idea is to solve the optimization problem (2.1) on the first machine, whose estimation accuracy and low-rank property have been proven in Theorem 1 and Theorem 2, respectively. Moreover, (A5) implies that $\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_* \le C\sqrt{r}a_n$ since $\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_* \le C\sqrt{r}\|\widehat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_F$ when $\operatorname{rank}(\widehat{\mathbf{\Theta}}) \le Cr$ holds. The following theorem and corollary guarantee the estimation accuracy of the proposed distributed estimator.

Theorem 3. *Assume conditions* (A1)–(A5) *hold,* $n \ge Cr^2\{(p+q)\log n\}^2$ *and* $\tau = Cv_\delta[N/\{(p+q)\log N\}]^{1/(1+\delta)}$ *for some sufficiently large constant* C*. In addition, assume for some sufficiently large* C*,*

$$\lambda \ge Ca_n\sqrt{\frac{r(p+q)\log n}{n}} + Ca_n\frac{r^{3/2}(p+q)^2(\log n)^2}{n} + Cv_\delta\left\{\frac{(p+q)\log N}{N}\right\}^{\delta/(1+\delta)}.$$

Algorithm 2 Communication-efficient ADMM algorithm for solving the distributed regularized adaptive Huber matrix regression.

Require: $\{(Y_{ij}, \mathbf{X}_{ij}) : i = 1, \ldots, n, j = 1, \ldots, m\}$ on machines $\{\mathcal{M}_j\}_{j=1}^m$.

Calculate the initial value $\widehat{\mathbf{\Theta}}$ on $\mathcal{M}_1$ by Algorithm 1;
Machine $\mathcal{M}_1$ transmits $\widehat{\boldsymbol{\theta}} = \mathrm{vec}(\widehat{\mathbf{\Theta}})$ to $\{\mathcal{M}_j\}_{j=1}^m$;
for $j = 1, 2, \ldots, m$ **do**
    Calculate $\nabla L_j(\widehat{\mathbf{\Theta}})$ on each machine $\mathcal{M}_j$;
    Transmit $\nabla L_j(\widehat{\mathbf{\Theta}})$ to machine $\mathcal{M}_1$;
end for
Calculate $\nabla L(\widehat{\mathbf{\Theta}}) = (1/m)\sum_{j=1}^m \nabla L_j(\widehat{\mathbf{\Theta}})$ and $\mathbf{g} = \mathrm{vec}(\nabla L_1(\widehat{\mathbf{\Theta}}) - \nabla L(\widehat{\mathbf{\Theta}}))$ on machine $\mathcal{M}_1$;
Update $(\mathbf{r}^k, \mathbf{B}^k, \boldsymbol{\theta}^k, \mathbf{u}^k, \mathbf{V}^k)$ on $\mathcal{M}_1$ as follows:
for $k = 0, 1, 2, \ldots, K-1$ **do**
    $h_i = (\mathbf{y}_1 - \bar{\mathbf{X}}_1\boldsymbol{\theta}^k - \mathbf{u}^k/\rho)_i$;
    $r_i^{k+1} = \begin{cases} n\rho h_i/(1+n\rho), & \text{if } |h_i| \le \tau(1+n\rho)/(n\rho), \\ \{h_i - \tau/(n\rho)\}_+ - \{-h_i - \tau/(n\rho)\}_+, & \text{otherwise;} \end{cases}$
    Calculate SVD[1]: $\mathbf{\Theta}^k - \mathbf{V}^k/\rho = \sum_{j=1}^{\min(p,q)} \omega_j \mathbf{a}_j \mathbf{c}_j^{\mathrm{T}}$;
    $\mathbf{B}^{k+1} = \sum_{j=1}^{\min(p,q)} (\omega_j - \lambda/\rho)_+ \mathbf{a}_j \mathbf{c}_j^{\mathrm{T}}$;
    $\boldsymbol{\theta}^{k+1} = (\bar{\mathbf{X}}_1^\top \bar{\mathbf{X}}_1 + \mathbf{I}_{pq})^{-1}\{\bar{\mathbf{X}}_1^\top(-\mathbf{r}^{k+1} + \mathbf{y}_1 - \mathbf{u}^k/\rho) + \mathbf{b}^{k+1} + \mathbf{v}^k/\rho + \mathbf{g}/\rho\}$;
    $\mathbf{u}^{k+1} = \mathbf{u}^k + \rho(\mathbf{r}^{k+1} - \mathbf{y}_1 + \bar{\mathbf{X}}_1\boldsymbol{\theta}^{k+1})$;
    $\mathbf{V}^{k+1} = \mathbf{V}^k + \rho(\mathbf{B}^{k+1} - \mathbf{\Theta}^{k+1})$;
end for
return $\boldsymbol{\theta}^K$.

[1]SVD denotes the singular value decomposition.

Then with probability at least $1 - n^{-C}$ *for some* $C > 0$, *we have*

$$\|\widetilde{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_F \le C\sqrt{r}\lambda, \quad \|\widetilde{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_* \le Cr\lambda.$$

Theorem 3 presents the convergence rates of the proposed distributed estimator, which are closely related to the initial estimation error a_n. The last term $Cv_\delta\sqrt{r}\{(p+q)\log N/N\}^{\delta/(1+\delta)}$ matches the upper bound obtained in Theorem 1 when we directly use the full data to estimate (2.1) (the resulting estimate is known as the central estimator). In addition, this term can be the dominating term under further sample size condition, as described in Corollary 1 below.

Corollary 1. *Assume that the conditions of Theorem* 3 *hold and*

$$a_n\sqrt{\frac{r(p+q)\log n}{n}} + a_n\frac{r^{3/2}(p+q)^2(\log n)^2}{n} \ll v_\delta\left\{\frac{(p+q)\log N}{N}\right\}^{\delta/(1+\delta)}. \quad (3.3)$$

Then with probability at least $1 - n^{-C}$ *for some* $C > 0$, *we have*

$$\|\widetilde{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_F \le Cv_\delta\sqrt{r}\left\{\frac{(p+q)\log N}{N}\right\}^{\delta/(1+\delta)},$$

$$\|\widetilde{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_* \leq C v_\delta r \left\{ \frac{(p+q)\log N}{N} \right\}^{\delta/(1+\delta)}.$$

Corollary 1 directly results from Theorem 3. As mentioned in Wang and Lian (2020), some quantitative relationship between m and N is normally required in order for the distributed learning to work, implying that n cannot be too small, or equivalently, m cannot be too large. To simplify (3.3), we consider some specific values of a_n to further illustrate the convergence rates of the distributed estimator. For example, if $v_\delta \asymp 1$ and $a_n \asymp n^{-1/4}$, then $N/\log N \ll n^{3(1+\delta)/4\delta} r^{-3(1+\delta)/2\delta}(p+q)^{-(2+\delta)/\delta}(\log n)^{-\{2(1+\delta)\}/\delta}$ (if $\delta = 1$ and we ignore the logarithmic terms, this further simplifies to $N \ll n^{3/2}/\{r^{3/2}(p+q)^3\}$) suffices to make $\widetilde{\mathbf{\Theta}}$ achieve the same rate as the central estimator which uses the full data directly, as established in Theorem 1. For the second case, we set $v_\delta \asymp 1$ and $a_n \asymp \sqrt{r}\{(p+q)\log n/n\}^{\delta/(1+\delta)}$ (this rate can be obtained by using the local data of size n). Then we require $N/\log N \ll n^{(1+3\delta)/2\delta} r^{-2(1+\delta)/\delta}(p+q)^{-2(1+\delta)/\delta}(\log n)^{(2+3\delta)/\delta}$ (if $\delta = 1$ and we ignore the logarithmic terms, this further simplifies to $N \ll n^2/\{r^4(p+q)^4\}$). Finally, we note that the theoretically choice of λ is not directly useful in practice, and we will use 5-fold cross-validation to choose the optimal λ.

Remark 2. Although not directly seen from the statement of Theorem 1, our proof actually shows that if the initial estimator is computed by (2.1) using local data on $\mathcal{M}_1$, we have

$$a_n = C\sqrt{r}\left\{ \sqrt{\frac{\sigma_\delta \tau^{1-\delta}(p+q)\log n}{n}} + \frac{\tau(p+q)\log n}{n} + \sigma_\delta \tau^{-\delta} \right\}.$$

In fact, as in the proof of Theorem 1 and Lemma 6, when computing the initial estimator, λ should be set to the order $\{\sqrt{\sigma_\delta \tau^{1-\delta}(p+q)\log n/n} + \tau(p+q)\log n/n + \sigma_\delta \tau^{-\delta}\}$. In this case, Corollary 1 would imply $n \geq (N/\log N)^{2/3}\{r(p+q)\log n\}^2$. Fortunately, this restriction can be removed if the communication-efficient distributed learning is iterated: once the t-th round distributed estimator is obtained, we can use it as the initial estimator of the $(t+1)$-th round and apply the distributed learning procedure summarized in Algorithm 2 again to further reduce the error. Let $\widetilde{\mathbf{\Theta}}^t$ denote the distributed estimator obtained at t-th round. After T rounds (T is specified below), $\widetilde{\mathbf{\Theta}}^T$ can achieve the convergence rate $C v_\delta \sqrt{r}\{(p+q)\log N/N\}^{\delta/(1+\delta)}$ without the above stated stringent condition on n or m. We state this relatively straightforward extension as follows whose proof is also contained in the supplementary material.

Corollary 2. *Assume that the conditions of Theorem* 3 *hold, except that in round* t*, we need to choose* $\lambda_t \asymp \{\sqrt{r(p+q)\log n/n} + r^{3/2}(p+q)^2(\log n)^2/n\}a_{n,t} + v_\delta\{(p+q)\log N/N\}^{\delta/(1+\delta)}$*, where* $a_{n,t} = \|\widetilde{\mathbf{\Theta}}^{t-1} - \mathbf{\Theta}_0\|_F$ *is the error of the initial estimator at the* t*-th round (also the estimator at the end of the* $(t-1)$*-th*

round). Define $b_n = Cr\sqrt{(p+q)\log n/n} + C\{r^2(p+q)^2(\log n)^2\}/n$ *and* $S_N = Cv_\delta\sqrt{r}\{(p+q)\log N/N\}^{\delta/(1+\delta)}$*, and assume* $b_n < 1/2$ *(or any other constant in* $(0,1)$*). After* $T \geq \log(S_N/a_n)/\log b_n$ *rounds, we have*

$$\|\widetilde{\boldsymbol{\Theta}}^T - \boldsymbol{\Theta}_0\|_F \leq Cv_\delta\sqrt{r}\left\{\frac{(p+q)\log N}{N}\right\}^{\delta/(1+\delta)},$$
$$\|\widetilde{\boldsymbol{\Theta}}^T - \boldsymbol{\Theta}_0\|_* \leq Cv_\delta r\left\{\frac{(p+q)\log N}{N}\right\}^{\delta/(1+\delta)}.$$

with probability at least $1 - Tn^{-C}$ *for some* $C > 0$*.*

Again, we can show the distributed estimator (3.2) has a low rank.

Theorem 4. *Assume conditions* (A1)–(A5) *are satisfied,* $\tau = Cv_\delta[N/\{(p+q)\log N\}]^{1/(1+\delta)}$*,* $\lambda \geq Cv_\delta\{(p+q)\log N/N\}^{\delta/(1+\delta)} \gg a_n\sqrt{r(p+q)\log n/n} + a_n r^{3/2}(p+q)^2(\log n)^2/n$*, and* $N \geq C\{\min(p,q)(p+q)\log N\}^4$ *for some sufficiently large constant* C*. Then with probability approaching one as* n *goes to infinity, we have*

$$\operatorname{rank}(\widetilde{\boldsymbol{\Theta}}) \leq Cr.$$

4. Numerical Studies: An Application to Beijing Air-Quality Dataset

Here we only report an analysis on a real data set and our simulation studies are deferred to the Supplementary Material. The Beijing Air-Quality dataset (Du et al., 2019) is available at UCI: `https://archive.ics.uci.edu/dataset/501/beijing+multi+site+air+quality+data`. The dataset provides hourly air quality information collected from 12 monitoring stations (each with 35,064 records and there are 420,768 records in total) in Beijing from March 2013 to February 2017. It is recognized that PM2.5 has become an important index to measure the level of air pollution. Here we are interested in predicting daily average PM2.5 concentrations. The data contains daily (from 0hr to 23hr) air quality information including SO2, NO2, CO, O3, temperature, pressure, dew point and wind speed serving as matrix covariates. That is, each observation $(\mathbf{X}_i, Y_i) \in \mathbb{R}^{24\times 8} \times \mathbb{R}$ and the total sample size is $N = 420768/24 = 17352$. The multi-site structure naturally leads to $m = 12$ and thus $n = N/m = 1461$. We fill the missing values using the average of the column and normalize the original data to $[0,1]$. The kurtosis of the normalized PM2.5 data is plotted in Figure 1, which shows that PM2.5 data at all 12 stations have heavy-tailed distributions.

We compare five estimators:

(a) LHuber: the local Huber estimator using the only data on $\mathcal{M}_1$;

(b) NHuber: the naive average of all local Huber estimators calculated on $\{\mathcal{M}_j\}_{j=1}^m$;

(c) DHuber: the proposed distributed Huber estimator;

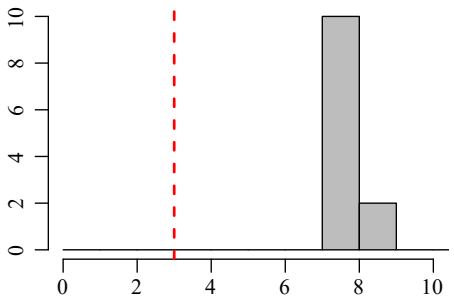

Figure 1. Histogram of kurtosis for PM2.5 (y-axis denotes the frequency). The dashed line at 3 is the kurtosis of standard normal distribution.

Table 3. Means and standard errors (in parentheses) of the prediction errors by different methods over 50 partitions.

	LHuber	NHuber	DHuber	DLS	DMed
RMSE	0.160(0.030)	0.128(0.002)	0.121(0.002)	0.161(0.003)	0.136(0.003)
MAE	0.119(0.023)	0.094(0.001)	0.089(0.001)	0.110(0.002)	0.095(0.001)

(d) DLS: the distributed least squares estimator using surrogate loss, that is, replacing Huber loss ℓ_τ in (3.2) with least squares loss $\ell(\cdot) = (\cdot)^2$ and still applying the CSL framework for the purpose of fair comparison;

(e) DMed: the distributed median estimator using surrogate loss, that is, replacing Huber loss ℓ_τ in (3.2) with median/absolute value loss $\ell(\cdot) = |\cdot|$ and still applying the CSL framework.

For each monitoring station, we randomly split the data into a training set containing 70% of the sample and use the remaining as a test set. Then we combine the training sets obtained from 12 stations as the final training set. The final test set is also obtained in this way. The above procedure is repeated 50 times. We use the training set to fit the model and select the tuning parameters and robustification parameters by 5-fold cross-validation. We report the averaged RMSE and MAE over the test data for 50 splittings, which are seperately defined as

$$\text{RMSE} = \left\{ \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} (Y_i - \widehat{Y}_i)^2 \right\}^{1/2} \quad \text{and} \quad \text{MAE} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} |Y_i - \widehat{Y}_i|.$$

In terms of both RMSE and MAE summarized in Table 3, we see that the DHuber achieves the smallest errors among all methods.

5. Discussions

In this paper, we have studied the adaptive Huber regression with matrix covariates and applied the communication-efficient distributed framework to deal with large-scale settings, enjoying robustness to heavy-tailed noises and outliers. It inherits the strengths of the adaptive Huber loss and distributed learning, and can be efficiently implemented by ADMM based algorithms. The estimation error bound and the low-rank properties of the proposed estimators are established. Extensive simulation results confirm the effectiveness of the proposed method.

It would also be interesting to extend to more general cases such as the covariate $\mathbf{X}$ and noise e are both heavy-tailed, or develop the robust tensor regression to handle tensor covariates and further investigate the distributed estimation for regularized Huber tensor regression. Besides, establishing the matching minimax lower bounds under the restricted strong convexity condition seems promising but challenging, since we allow heavy-tailed noises with only bounded $(1+\delta)$-th moment for some $\delta \in (0, 1]$. We leave these interesting topics for future research.

Supplementary Material

Proofs of the theorems and two simulation studies are contained in the Supplementary Materials.

Acknowledgments

We sincerely thank the editor Professor Su-Yun Huang, an associate editor and two referees for valuable comments and constructive suggestions. Hongmei Lin's research is partially supported by the National Natural Science Foundation of China (11701360, 11971171, 11971300), the Shanghai Natural Science Foundation (20ZR1421800,19ZR1420900), and the Open Research Fund of Key Laboratory of Advanced Theory and Application in Statistics and Data Science (East China Normal University). The research of Heng Lian is supported by NSP of Jiangxi Province (No 20223BCJ25017), and by Hong Kong RGC general research fund 11300519, 11300721 and 11311822.

References

Battey, H., Fan, J., Liu, H., Lu, J. and Zhu, Z. (2018). Distributed testing and estimation under sparse high dimensional models. *The Annals of Statistics* **46**, 1352–1382.

Boyd, S., Parikh, N. and Chu, E. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. Now Publishers Inc.

Candès, E. J. and Tao, T. (2010). The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory* **56**, 2053–2080.

Chen, X., Liu, W., Mao, X. and Yang, Z. (2020). Distributed high-dimensional regression under a quantile loss function. *Journal of Machine Learning Research* **21**, 1–43.

Chen, X. and Zhou, W.-X. (2020). Robust inference via multiplier bootstrap. *The Annals of Statistics* **48**, 1665–1691.

Du, S., Li, T., Yang, Y. and Horng, S.-J. (2019). Deep air quality forecasting using hybrid deep learning framework. *IEEE Transactions on Knowledge and Data Engineering* **33**, 2412–2424.

Elsener, A. and van de Geer, S. (2018). Robust low-rank matrix estimation. *The Annals of Statistics* **46**, 3481–3509.

Fan, J., Gong, W. and Zhu, Z. (2019). Generalized high-dimensional trace regression via nuclear norm regularization. *Journal of Econometrics* **212**, 177–202.

Fan, J., Li, Q. and Wang, Y. (2017). Estimation of high dimensional mean regression in the absence of symmetry and light tail assumptions. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* **79**, 247–265.

Fan, J., Wang, W. and Zhu, Z. (2021). A shrinkage principle for heavy-tailed data: High-dimensional robust low-rank matrix recovery. *The Annals of Statistics* **49**, 1239–1266.

Huber, P. J. (1973). Robust regression: Asymptotics, conjectures and Monte Carlo. *The Annals of Statistics* **1**, 799–821.

Jordan, M. I., Lee, J. D. and Yang, Y. (2018). Communication-efficient distributed statistical inference. *Journal of the American Statistical Association* **114**, 668–681.

Koltchinskii, V., Lounici, K. and Tsybakov, A. B. (2011). Nuclear-norm penalization and optimal rates for noisy low-rank matrix completion. *The Annals of Statistics* **39**, 2302–2329.

Lambert-Lacroix, S. and Zwald, L. (2011). Robust regression through the Huber's criterion and adaptive Lasso penalty. *Electronic Journal of Statistics* **5**, 1015–1053.

Lee, J. D., Liu, Q., Sun, Y. and Taylor, J. E. (2017). Communication-efficient sparse regression. *Journal of Machine Learning Research* **18**, 115–144.

Loh, P.-L. (2017). Statistical consistency and asymptotic normality for high-dimensional robust m-estimators. *The Annals of Statistics* **45**, 866–896.

Naseem, I., Togneri, R. and Bennamoun, M. (2012). Robust regression for face recognition. *Pattern Recognition* **45**, 104–118.

Negahban, S. and Wainwright, M. J. (2011). Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics* **39**, 1069–1097.

Negahban, S. N., Ravikumar, P., Wainwright, M. J. and Yu, B. (2012). A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statistical Science* **27**, 538–557.

Rohde, A. and Tsybakov, A. B. (2011). Estimation of high-dimensional low-rank matrices. *The Annals of Statistics* **39**, 887–930.

Senneret, M., Malevergne, Y., Abry, P., Perrin, G. and Jaffres, L. (2016). Covariance versus precision matrix estimation for efficient asset allocation. *IEEE Journal of Selected Topics in Signal Processing* **10**, 982–993.

Sun, Q., Zhou, W.-X. and Fan, J. (2020). Adaptive Huber regression. *Journal of the American Statistical Association* **115**, 254–265.

Wang, L. and Lian, H. (2020). Communication-efficient estimation of high-dimensional quantile regression. *Analysis and Applications* **18**, 1057–1075.

Wang, L., Zheng, C., Zhou, W. and Zhou, W.-X. (2021). A new principle for tuning-free huber regression. *Statistica Sinica* **31**, 2153–2177.

Yang, J., Luo, L., Qian, J., Tai, Y., Zhang, F. and Xu, Y. (2016). Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39**, 156–171.

Zhou, H. and Li, L. (2014). Regularized matrix regression. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* **76**, 463–483.

(Received May 2022; accepted June 2023)