A New Tuning-free Approach to High-dimensional Regression

Lan Wang

School of Statistics
University of Minnesota
(Joint work with Bo Peng, Jelena Bradic, Runze Li and Yunan Wu)

WHOA-PSI, 2018
We consider the linear regression model

\[ Y = X \beta_0 + \epsilon. \]

- \( Y = (Y_1, \ldots, Y_n)^T \) is an \( n \times 1 \) vector of responses
- \( X \) is an \( n \times p \) centered matrix of covariates
- \( \beta_0 = (\beta_{01}, \ldots, \beta_{0p})^T \) is a \( p \times 1 \) vector of unknown parameters
- \( \epsilon = (\epsilon_1, \ldots, \epsilon_n)^T \) is an \( n \times 1 \) vector of independent and identically distributed random errors.
- We are interested in the setting \( p \gg n \).
- Let \( A = \{j : \beta_{0j} \neq 0, \ j = 1, \ldots, p\} \) and \( \|A\|_0 = q \). The model is assumed to be sparse in the sense that \( q \ll n \).
Motivations for this work

Challenge 1: How to determine the right amount of regularization in a computationally efficient way with proper theoretical justification?

- Practical performance of regularized high-dimensional regression depends crucially on the choice of a tuning parameter $\lambda$.
- The optimal choice of $\lambda$ for Lasso depends on both the random error distribution and the design matrix.

$$P\left\{ \| n^{-1}X^T \epsilon \|_\infty \leq \lambda \right\} \geq 1 - \alpha,$$

for some small $\alpha > 0$.

- Exiting theory for Lasso is often derived while fixing $\lambda$ at a theoretical value $\tau \sigma \sqrt{\log p/n}$, where $\sigma$ is the standard deviation of $\epsilon$, and $\tau$ is some positive constant.
Motivations for this work (cont’d)

- Estimation of $\sigma$ in high dimension is itself a very difficult problem.
- Practitioners often employ cross-validation to select $\lambda$.
  - Cross-validation is computationally intensive.
  - It is unknown whether the cross-validated Lasso shares the same near-oracle rate as Lasso does when $\lambda$ is fixed at the ideal theoretical value.
- Hence, there still exits an important gap between the theory and practice of Lasso.
The scaled Lasso of Sun and Zhang (2012) iteratively estimates the regression parameter and $\sigma$.

Square-root Lasso (Belloni et al., 2011) eliminates the need to calibrate $\lambda$ for $\sigma$ but does not adjust for the design matrix nor the tail of the random error distribution.

TREX (Lederer and Müller, 2015) automatically adjusts $\lambda$ for both the tail of the error distribution and the design matrix but the modified loss function is no longer convex.

Sabourin et al. (2015) adopts a permutation approach and Chichignoud et al. (2016) develops a novel testing procedure to select $\lambda$. 
Challenge II: how to properly handle heavy-tailed error contamination in high dimension so that one achieves robustness while maintaining efficiency at the normal error setting?

- High-dimensional M-estimation based on Huber’s loss (Fan et al., 2017; Loh, 2017; Sun et al., 2017): additional tuning parameter.
- Least absolute deviation loss (Belloni et al., 2011; Bradic et al., 2011; Wang et al., 2012; Wang, 2013; Fan et al. 2014): significant efficiency loss may occur for normal random errors.
- Existing work on high-dimensional robust regression has not addressed the problem of tuning parameter selection and may require some additional tuning parameter to achieve robustness.
Motivations for this work (cont’d)

Challenge III: how to coherently interpret the results from Lasso while $\gamma$ undergoes a scale transformation?

- The classical least squares estimator enjoys an equivariance property that permits a coherent interpretation.
- Unfortunately, this property is lost by Lasso.
We will introduce a new estimation procedure that is:

- Computationally convenient
- Tuning free and has theoretical guarantee
- Almost as efficient as LS-Lasso at normal random errors
- Robust and more efficient than LS-Lasso at heavy-tailed errors
- Coherent interpretation: equivariant to scale change of $Y$; estimating the same slope parameters as LS-Lasso does.
Our work (cont’d)

**Figure:** Comparison of four different estimation procedures for Example 1 (normal error versus mixture normal error)
We will consider the following $L_1$ regularized estimator of $\beta_0$:

$$\hat{\beta}(\lambda) = \arg\min_{\beta \in \mathbb{R}^p} \left\{ \left[ n(n - 1) \right]^{-1} \sum_{i \neq j} \sum \left| (Y_i - x_i^T \beta) - (Y_j - x_j^T \beta) \right| + \lambda \sum_{k=1}^{p} |\beta_k| \right\}.$$ 

- Minimizing this loss is equivalent to minimizing Jaeckel’s dispersion function with Wilcoxon scores:

$$\sqrt{12} \sum_{i=1}^{n} \left[ \frac{R(Y_i - x_i^T \beta)}{n+1} - \frac{1}{2} \right] (Y_i - x_i^T \beta),$$

where $R(Y_i - x_i^T \beta)$ denotes the rank of $Y_i - x_i^T \beta$ among $Y_1 - x_1^T \beta, \ldots, Y_n - x_n^T \beta$. 

---

The new method: Rank-Lasso
The new method (cont’d)

With the aid of slack variables $\xi^{+}_{ij}$, $\xi^{-}_{ij}$, and $\zeta_{k}$, the convex optimization problem can be equivalently expressed as a linear programming problem.

$$
\min_{\beta, \xi, \zeta} \left\{ \left[ n(n-1) \right]^{-1} \sum_{i \neq j} \sum (\xi^{+}_{ij} + \xi^{-}_{ij}) + \lambda \sum_{k=1}^{p} \zeta_{k} \right\}
$$

subject to

$$
\xi^{+}_{ij} - \xi^{-}_{ij} = (Y_{i} - Y_{j}) - (x_{i} - x_{j})^{T} \beta, \ i, j = 1, 2, \cdots, n;
$$
$$
\xi^{+}_{ij} \geq 0, \xi^{-}_{ij} \geq 0, \ i, j = 1, 2, \cdots, n;
$$
$$
\zeta_{k} \geq \beta_{k}, \zeta_{k} \geq -\beta_{k}, \ k = 1, 2, \cdots, p.
$$
The new method (cont’d)

- For random errors with distributions symmetric about zero, $x_i^T \beta_0$ coincides with the conditional mean.
- However, symmetric random error distribution assumption is not required. For i.i.d. random errors, $\beta_0$ still bears the interpretation as the effects of the covariates on the conditional mean. This is different from Huber’s loss function.
Completely pivotal property

- Write

\[ Q_n(\gamma) = [n(n-1)]^{-1} \sum_{i \neq j} \sum |(Y_i - x_i^T \gamma) - (Y_j - x_j^T \gamma)| \]

\[ = [n(n-1)]^{-1} \sum_{i \neq j} \sum |(\epsilon_i - \epsilon_j) - (x_i - x_j)^T \gamma|. \]

- Denote the subgradient of \( Q_n(\gamma) \) at \( \gamma = 0 \) (or equivalently \( \beta = \beta_0 \)) by \( S_n = \frac{\partial Q_n(\gamma)}{\partial \gamma} |_{\gamma=0} \). We would like to choose \( \lambda \) such that

\[ P(\lambda > c ||S_n||_\infty) \geq 1 - \alpha_0, \]

for some constant \( c > 1 \) and a given small \( \alpha_0 > 0 \).
We can show that

$$S_n = \frac{\partial Q_n(\gamma)}{\partial \gamma} |_{\gamma=0} = -2[n(n-1)]^{-1} \sum_{j=1}^{n} x_j \left( \sum_{i=1, i \neq j}^{n} \text{sign}(\epsilon_j - \epsilon_i) \right).$$

where \(\text{sign}(t) = 1\) if \(t > 0\), \(= -1\) if \(t < 0\), and \(= 0\) if \(t = 0\).

- Observe that \(\xi_j = \sum_{i=1, i \neq j}^{n} \text{sign}(\epsilon_j - \epsilon_i) = 2\text{rank}(\epsilon_j) - (n+1)\), where \(\text{rank}(\epsilon_j)\) is the the rank of \(\epsilon_j\) among \(\{\epsilon_1, \ldots, \epsilon_n\}\).
- Since \((\text{rank}(\epsilon_1), \ldots, \text{rank}(\epsilon_n))^T\) follows the uniform distribution on the permutations of integers \(\{1, 2, \ldots, n\}\), the gradient function \(S_n\) has a completely known distribution.
Lemma

\[ S_n = -2 [n(n-1)]^{-1} X^T \xi, \]

where \( \xi = (\xi_1, \ldots, \xi_n)^T \) has a completely known distribution that is independent of the random error distribution.

For any given \( c \) and \( \alpha_0 \), we suggest to take \( \lambda \) equal to

\[ \lambda^* = c G_{\|S_n\|_{\infty}}^{-1} (1 - \alpha_0) \]

where \( G_{\|S_n\|_{\infty}}^{-1} (1 - \alpha_0) \) denotes the \((1 - \alpha_0)\)-quantile of the distribution of \( \|S_n\|_{\infty} \).
Square-root Lasso has a partial pivotal property. The gradient of its loss function being evaluated at \( \beta_0 \) has the form

\[
(\sum_{i=1}^{n} \epsilon_i^2)^{1/2} \sum_{i=1}^{n} x_i \epsilon_i,
\]

which does not depend on the random error standard deviation \( \sigma \) but still depends on other aspects of the error distribution.

The careful analysis in Hebiri and Lederer (2013) reveals that ignoring the correlation structure of the design matrix may lead to unsatisfactory choice of \( \lambda \) and sub-optimal performance of Lasso.
Equivariance of the penalized estimator

Lemma

Let $\hat{\beta}(\lambda^*, Y, X)$ be the proposed new estimator with the simulated tuning parameter $\lambda^*$ based on a response vector $Y$ and a design matrix $X$. Then

$$\hat{\beta}(\lambda^*, cY, X) = c\hat{\beta}(\lambda^*, Y, X)$$

for any nonzero constant $c$. 
Near-oracle rate of $L_2$ error bound

Consider the following cone set

$$
\Gamma = \{ \gamma \in \mathbb{R}^p : \|\gamma_B^c\|_1 \leq \bar{c}\|\gamma_B\|_1, \ B \subset \{1, 2, \ldots, p\} \text{ and } \|B\|_0 \leq q \},
$$

where $\bar{c} = \frac{c+1}{c-1}$.

Lemma

(i) Let $\hat{\gamma}(\lambda) = \hat{\beta}(\lambda) - \beta_0$. For any $\lambda \geq c\|S_n\|_\infty$, we have

$$
\hat{\gamma}(\lambda) \in \Gamma.
$$

(ii) There exists a universal constant $c_0$ such that for any positive constant $l > 1$,

$$
P(c\|S_n\|_\infty < lc_0 \sqrt{\log p/n}) \geq 1 - 2 \exp\left(-\left(l^2 - 1\right)\log p\right).
$$
Near-oracle rate of $L_2$ error bound (cont’d)

Recall: for any given $c$ and $\alpha_0$, we suggest to take $\lambda$ equal to

$$
\lambda^* = cG_{||S_n||_\infty}^{-1} (1 - \alpha_0)
$$

where $G_{||S_n||_\infty}^{-1} (1 - \alpha_0)$ denotes the $(1 - \alpha_0)$-quantile of the distribution of $||S_n||_\infty$.

**Theorem**

Suppose conditions (C1)–(C3) hold. If $p > (2/\alpha_0)^{1/3}$, then the proposed new estimator $\hat{\beta}(\lambda^*)$ satisfies

$$
||\hat{\beta}(\lambda^*) - \beta_0||_2 \leq \frac{8(1 + \bar{c})c_0}{b_2 b_3} \sqrt{\frac{q \log p}{n}}
$$

with probability at least $1 - \alpha_0 - \exp(-2 \log p)$. 
A numerical example

We generate random data from the regression model
\[ Y = X^T \beta_0 + \epsilon, \]
where \( X \sim N(0, \Sigma) \), and
\[ \beta_0 = (\sqrt{3}, \sqrt{3}, \sqrt{3}, 0, \ldots, 0)^T, \]
\( \Sigma \) is the equally correlated correlation matrix: \( \Sigma_{ij} = 0.5 \) for \( i \neq j \); and \( \Sigma_{ij} = 1 \) for \( i = j \).

We consider the following six different distributions for \( \epsilon_i \):

- \( N(0, 1) \): normal distribution with mean 0 and variance 1;
- \( N(0, 2) \): normal distribution with mean 0 and variance 2;
- \( MN \): mixture normal distribution
  
  \( \epsilon \sim 0.95N(0, 1) + 0.05N(0, 100); \)

- \( t \) distribution: \( \epsilon \sim \sqrt{2}t(4) \), where \( t(4) \) denotes \( t \) distribution with 4 degree of freedom;
- Cauchy distribution: \( \epsilon \sim \text{Cauchy}(0, 1) \).
<table>
<thead>
<tr>
<th>Error</th>
<th>Method</th>
<th>L1 error</th>
<th>L2 error</th>
<th>FP</th>
<th>FN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$N(0, 1)$</td>
<td>Lasso</td>
<td>1.54 (0.59)</td>
<td>0.57 (0.15)</td>
<td>13.08 (6.06)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Sqrt Lasso</td>
<td>1.41 (0.39)</td>
<td>0.54 (0.12)</td>
<td>11.46 (4.71)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank Lasso</td>
<td>1.24 (0.34)</td>
<td>0.59 (0.12)</td>
<td>6.32 (3.24)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank SCAD</td>
<td>0.41 (0.23)</td>
<td>0.28 (0.16)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>$N(0, 2)$</td>
<td>Lasso</td>
<td>2.16 (0.70)</td>
<td>0.80 (0.18)</td>
<td>12.32 (4.41)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Sqrt Lasso</td>
<td>2.07 (0.61)</td>
<td>0.78 (0.18)</td>
<td>11.52 (3.76)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank Lasso</td>
<td>1.79 (0.51)</td>
<td>0.82 (0.16)</td>
<td>6.65 (3.30)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank SCAD</td>
<td>0.81 (0.42)</td>
<td>0.52 (0.27)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>MN</td>
<td>Lasso</td>
<td>3.02 (1.67)</td>
<td>1.12 (0.55)</td>
<td>12.00 (4.91)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Sqrt Lasso</td>
<td>3.10 (1.56)</td>
<td>1.09 (0.53)</td>
<td>14.23 (3.78)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank Lasso</td>
<td>0.14 (0.04)</td>
<td>0.07 (0.01)</td>
<td>6.85 (3.15)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank SCAD</td>
<td>0.03 (0.02)</td>
<td>0.02 (0.01)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>$\sqrt{2t_4}$</td>
<td>Lasso</td>
<td>3.42 (1.21)</td>
<td>1.18 (0.29)</td>
<td>15.52 (7.70)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Sqrt Lasso</td>
<td>3.01 (0.80)</td>
<td>1.10 (0.23)</td>
<td>12.58 (4.25)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank Lasso</td>
<td>2.20 (0.63)</td>
<td>1.02 (0.22)</td>
<td>7.39 (3.50)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank SCAD</td>
<td>1.21 (0.59)</td>
<td>0.78 (0.38)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>Cauchy</td>
<td>Lasso</td>
<td>7.32 (2.27)</td>
<td>3.16 (0.39)</td>
<td>5.84 (5.44)</td>
<td>2.17 (1.12)</td>
</tr>
<tr>
<td></td>
<td>Sqrt Lasso</td>
<td>9.31 (2.88)</td>
<td>3.45 (0.89)</td>
<td>6.93 (3.90)</td>
<td>2.16 (1.12)</td>
</tr>
<tr>
<td></td>
<td>Rank Lasso</td>
<td>2.60 (0.70)</td>
<td>1.27 (0.31)</td>
<td>6.00 (2.81)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td>Rank SCAD</td>
<td>1.89 (0.96)</td>
<td>1.21 (0.58)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
</tbody>
</table>
A second-stage enhancement with some light tuning to
- recover the support of the generative model with high probability;
- estimate the nonzero coefficients with high efficiency.

Let the initial estimator $\tilde{\beta}^{(0)} = (\tilde{\beta}_1^{(0)}, \ldots, \tilde{\beta}_p^{(0)})^T$ be $\hat{\beta}(\lambda^*)$. The second-stage estimator is defined as

$$
\tilde{\beta}^{(1)} = \arg\min_{\beta} \left\{ \left[ n(n-1) \right]^{-1} \sum_{i \neq j} \left| (Y_i - x_i^T \beta) - (Y_j - x_j^T \beta) \right| + \sum_{k=1}^{p} p'_{\eta}(|\tilde{\beta}_k^{(0)}|) |\beta_j| \right\},
$$

where $p'_{\eta}(\cdot)$ denotes the derivative of some nonconvex penalty function $p_{\eta}(\cdot)$, and $\eta > 0$ is a tuning parameter.
Bias reduction and efficiency improvement (cont’d)

Two popular choices of nonconvex penalty functions are the SCAD penalty function (Fan and Li, 2001) and the MCP penalty function (Zhang, 2010).

The SCAD penalty function is given by

\[ p_\eta(\beta) = \eta |\beta| I(0 \leq |\beta| < \eta) + \frac{a\eta |\beta| - (\beta^2 + \eta^2)/2}{a - 1} I(\eta \leq |\beta| \leq a\eta) + \frac{(a + 1)\eta^2}{2} I(|\beta| > a\eta), \]

for some \( a > 2 \).

The MCP function has the form

\[ p_\eta(\beta) = \eta \left( |\beta| - \frac{\beta^2}{2a\eta} \right) I(0 \leq |\beta| < a\eta) + \frac{a\eta^2}{2} I(|\beta| \geq a\eta), \]

for some \( a > 1 \).
Let $\beta_\eta = (\beta_{\eta,1}, \ldots, \beta_{\eta,p})^T$ be obtained using tuning parameter $\eta$. Let $S_\eta = \{j : \beta_{\eta,j} \neq 0, 1 \leq j \leq p\}$. Define

$$
HBIC(\eta) = \log \left( \sum \sum_{i \neq j} |(Y_i - x_i^T \beta_\eta) - (Y_j - x_j^T \beta_\eta)| \right) + |S_\eta| \frac{\log(\log n)}{n} C_n,
$$

where $C_n$ is a sequence of positive constants diverging to infinity as $n$ increases. In practice, we recommend to take $C_n = O(\log(p))$. 

WLOG, write $\beta_0 = (\beta_{01}^T, 0_{p-q}^T)^T$.

Let $L_n(\beta_1) = \sum \sum_{i \neq j} |(Y_i - x_{1i}^T \beta_1) - (Y_j - x_{1j}^T \beta_1)|$ and $\hat{\beta}_1^{(o)} = \arg\min_{\beta_1} L_n(\beta_1)$. The oracle estimator for $\beta_0$ is

\[ \hat{\beta}^o = (\hat{\beta}_1^{(o)T}, 0_{p-q}^T)^T. \]

**Theorem**

*Under some regularity conditions, suppose $q = O(n^{c_1})$, $\eta = O(n^{-(1-c_2)/2})$, $\min_{1 \leq j \leq q} |\beta_{0j}| \geq bn^{-(1-c_3)/2}$, $p = \exp(n^{c_4})$ for some positive constants $b$ and $c_i$ ($i = 1, \ldots, 4$) such that $2c_1 < c_2 < c_3 \leq 1$ and $c_1 + c_4 < c_2$. We have*

\[ P(\hat{\beta}^{(1)} = \hat{\beta}^{(o)}) \to 1, \quad \text{as } n \to \infty. \]
Insights into efficiency

- \( \sqrt{n}(\tilde{\beta}_1 - \beta_{01}) \) follows an asymptotic normal distribution in the case \( q \) is fixed. The relative efficiency (ARE) of \( \tilde{\beta}_1 \) with respect to the least-squares oracle for estimating \( \beta_{01} \) is

\[
ARE = 12\sigma^2 \left[ \int f^2(u)du \right]^2.
\]

- The ARE is 0.955 for normal error distribution, and can be significantly higher than one for many heavier-tailed error distributions. ARE is 1.5 for the double exponential distribution, and is 1.9 for the \( t_3 \) distribution.

- For symmetric error distributions with finite Fisher information, the ARE is known to have a lower bound equal to 0.864.
We consider three different choices of $\Sigma$:

- $\Sigma = \Sigma_1$ is the compound symmetry correlated correlation matrix with correlation coefficient 0.8;
- $\Sigma = \Sigma_2$ is the compound symmetry correlation matrix with correlation coefficient 0.2;
- $\Sigma = \Sigma_3$ is the AR(1) correlation matrix with auto-correlation coefficient 0.5.

For each choice of $\Sigma$, we consider $N(0, 1)$ and mixture normal error distributions.
<table>
<thead>
<tr>
<th>Σ</th>
<th>Error</th>
<th>Method</th>
<th>L1 error</th>
<th>L2 error</th>
<th>FP</th>
<th>FN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Σ</td>
<td></td>
<td>Lasso</td>
<td>2.42 (0.79)</td>
<td>0.88 (0.22)</td>
<td>14.42 (6.48)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sqrt Lasso</td>
<td>2.22 (0.56)</td>
<td>0.83 (0.18)</td>
<td>13.09 (4.18)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank Lasso</td>
<td>2.23 (0.59)</td>
<td>0.91 (0.20)</td>
<td>11.48 (3.99)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank SCAD</td>
<td>0.55 (0.29)</td>
<td>0.34 (0.17)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>N(0,1)</td>
<td></td>
<td>Lasso</td>
<td>4.87 (2.51)</td>
<td>1.75 (0.83)</td>
<td>12.30 (3.56)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sqrt Lasso</td>
<td>4.79 (2.45)</td>
<td>1.72 (0.82)</td>
<td>12.23 (3.65)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank Lasso</td>
<td>0.27 (0.08)</td>
<td>0.11 (0.03)</td>
<td>12.29 (3.76)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank SCAD</td>
<td>0.05 (0.02)</td>
<td>0.04 (0.01)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>MN</td>
<td></td>
<td>Lasso</td>
<td>1.21 (0.50)</td>
<td>0.45 (0.11)</td>
<td>13.08 (8.80)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sqrt Lasso</td>
<td>0.99 (0.24)</td>
<td>0.44 (0.09)</td>
<td>6.50 (3.21)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank Lasso</td>
<td>0.94 (0.17)</td>
<td>0.55 (0.09)</td>
<td>1.34 (1.34)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank SCAD</td>
<td>0.32 (0.16)</td>
<td>0.20 (0.10)</td>
<td>0.56 (1.04)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>Σ1</td>
<td></td>
<td>Lasso</td>
<td>2.49 (1.41)</td>
<td>0.92 (0.47)</td>
<td>11.37 (4.96)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sqrt Lasso</td>
<td>2.18 (1.14)</td>
<td>0.93 (0.47)</td>
<td>7.65 (3.29)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank Lasso</td>
<td>0.11 (0.03)</td>
<td>0.07 (0.01)</td>
<td>1.62 (1.60)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank SCAD</td>
<td>0.03 (0.01)</td>
<td>0.02 (0.01)</td>
<td>0.40 (0.59)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>N(0,1)</td>
<td></td>
<td>Lasso</td>
<td>0.80 (0.36)</td>
<td>0.36 (0.11)</td>
<td>9.38 (8.55)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sqrt Lasso</td>
<td>0.71 (0.19)</td>
<td>0.347 (0.09)</td>
<td>4.47 (2.06)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank Lasso</td>
<td>0.64 (0.13)</td>
<td>0.43 (0.09)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank SCAD</td>
<td>0.41 (0.24)</td>
<td>0.25 (0.13)</td>
<td>1.11 (1.80)</td>
<td>0 (0)</td>
</tr>
<tr>
<td>MN</td>
<td></td>
<td>Lasso</td>
<td>1.53 (0.88)</td>
<td>0.67 (0.35)</td>
<td>7.12 (4.64)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sqrt Lasso</td>
<td>1.55 (0.80)</td>
<td>0.68 (0.34)</td>
<td>6.32 (2.67)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank Lasso</td>
<td>0.08 (0.02)</td>
<td>0.05 (0.01)</td>
<td>0 (0)</td>
<td>0 (0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rank SCAD</td>
<td>0.04 (0.02)</td>
<td>0.02 (0.0)</td>
<td>0.62 (0.72)</td>
<td>0 (0)</td>
</tr>
</tbody>
</table>
Conclusions and discussions

- The proposed $L_1$ regularized new estimator achieves several goals simultaneously:
  - It keeps the convex structure for convenient computation.
  - It has a tuning parameter that can be easily simulated and automatically adapts to both the error distribution and the design matrix.
  - The penalized estimator is equivariant to scale transformation of the response variable.
  - The $L_2$ estimation error bound of new estimator achieves the same near-oracle rate as Lasso does.
  - It has similar performance as Lasso does with normal random random error distribution and can be substantially more efficient with heavy-tailed error distribution.

- Its efficiency can be further improved via a second-stage enhancement with some light tuning.
Future work:

- From tuning-free estimation to tuning-free inference...
- Equivariance to scale change of covariates without sacrificing the existing nice properties...