Best Subset Selection via a Modern Optimization Lens

Ying Wang & Jiyanglin Li

School of Statistics and Management
Shanghai University of Finance and Economics

December 24, 2015

Outline

1. Background
2. Mordern Optimization Lenz
   - Mix Integrate Optimization Formulations
   - Discrete First Order Algorithm
3. Computational Results
   - The Classical Overdetermined Case ($n > p$)
   - High Dimensional Case ($p >> n$)
   - Others
4. Conclusions

Linear Regression Model

Consider the linear regression model:

$$y = X\beta + \epsilon$$

where $y, \epsilon \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times p}$,$\beta \in \mathbb{R}^p$.

- It's desirable to obtain a parsimonious fit to the data by finding the best k-feature fit to the response y.
- When $p >> n$, it's desirable to assume that the true regression coefficient β is sparse or may be well approximated by a sparse vector.

What is Best Subset Selection?

Best subset selection optimizes

$$\min_{\beta} \frac{1}{2}\|y - X\beta\|_2^2 \quad \text{s. t.} \quad \|\beta\|_0 \leq k. \tag{1}$$

where $\|\beta\|_0 = \sum_{i=1}^{p} 1(\beta_i \neq 0)$.

- Best subset selection is a NP-hard problem, widely dismissed as being intractable as p increases.

Some Methods to Solve Best Subset Selection

- Using Lasso to approximate the solution

$$\min_{\beta} \frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1 \tag{2}$$

where $\|\beta\|_1 = \sum_{i=1}^{p} |\beta_i|$ shrinks the coefficients towards zero and produces a sparse solution by setting many coefficients to be exactly zero.

- Continuous non-convex optimization problems of the form can overcome the shortcomings of Lasso:

$$\min_{\beta} \frac{1}{2}\|y - X\beta\|_2^2 + \sum_{i} p(|\beta_i|; \gamma; \lambda) \tag{3}$$

where $p(|\beta_i|; \gamma; \lambda)$ is a non-convex function in β with γ and λ denoting the degree of regularization and non-convexity respectively.

Author's Approaches

In the paper, the author mainly focus on two discrete optimization methods which will be discussed later to address Problem (1).

- Mixed integer optimization (MIO)
- Discrete first order methods (ideas from projected gradient descent method)

1 Background

2 Mordern Optimization Lenz
- Mix Integrate Optimization Formulations
- Discrete First Order Algorithm

3 Computational Results

4 Conclusions

Mixed Integer Optimization

Mixed Integer Quadratic Optimization(MIQO) problem

$$\begin{aligned} \min \quad & \alpha^T Q\alpha + \alpha^T a \\ s.t. \quad & A\alpha \leq b \\ & \alpha_i \in \mathbb{N}, \ \forall i \in \mathcal{I} \\ & \alpha_j \in \mathbb{R}_+, \ \forall j \notin \mathcal{I} \\ & \mathcal{I} \subset \{1, 2, ..., m\}. \end{aligned}$$

Remark: When $Q = 0_{m\times m}$, this problem is known as the mixed interger programming (MIO) problem.

MIO Formulations for the Best Subset Selection

Transform Problem (1) to a MIO:

$$
\begin{aligned}
Z_1 = \min_{\beta,z} \quad & \frac{1}{2}\|y - X\beta\|_2^2 \\
s.t. \quad & -\mathcal{M}_U z_i \leq \beta_i \leq \mathcal{M}_U z_i, i = 1, ..., p \\
& z_i \in \{0,1\}, i = 1, ..., p \\
& \sum_{i=1}^{p} z_i \leq k,
\end{aligned} \tag{5}
$$

where $z \in \{0,1\}^p$, $\mathcal{M}_U$ is a constant.
Remark: Before more specific discussion, we first normalize the X so that the columns of X satisfy that $\|x_j\|_2 = 1$ and $\sum_i x_{ij} = 0$.

MIO Formulations for the Best Subset Selection

Consider the following two optimization problems,

$$Z_2 = \min_{\beta} \frac{1}{2}\|y - X\beta\|_2^2 s.t. \|\beta\|_\infty \leq \mathcal{M}_U, \|\beta\|_1 \leq \mathcal{M}_U k; \tag{6}$$

$$Z_3 = \min_{\beta} \frac{1}{2}\|y - X\beta\|_2^2 s.t. \|\beta\|_1 \leq \mathcal{M}_U k. \tag{7}$$

And let S_1, S_2, S_3 represent the feasible region of problem (5)-(7) respectively, then it's easy to find that $S_1 \subset S_2 \subset S_3$. Thus we have

$$Z_3 \leq Z_2 \leq Z_1$$

Formulations via Specially Ordered Sets

Given appropriate $\mathcal{M}_\ell, \mathcal{M}_U$, Problem (1) is equivalent to problem below:

$$\begin{aligned}
\min_{\beta,z} \quad & \frac{1}{2}\beta^T X^T X \beta - \langle X'y, \beta \rangle + \frac{1}{2}\|y\|_2^2 \\
s.t \quad & (\beta_i, 1 - z_i) : \text{SOS-1}, i = 1, ..., p \\
& z_i \in \{0, 1\}, i = 1, ..., p \\
& \sum_{i=1}^{p} z_i \leq k \\
& -\mathcal{M}_U \leq \beta_i \leq \mathcal{M}_U, i = 1, ..., p \\
& \|\beta\|_1 \leq \mathcal{M}_\ell
\end{aligned} \tag{9}$$

where $(1 - z_i)\beta_i = 0 \Leftrightarrow (\beta_i, 1 - z_i) : \text{SOS-1}$.

Formulations via Specially Ordered Sets

We can also consider another formulation for (9):

$$
\begin{aligned}
\min_{\beta,z,\zeta} \quad & \frac{1}{2}\zeta^T\zeta - \langle X'y, \beta\rangle + \frac{1}{2}\|y\|_2^2 \\
s.t \quad & \zeta = X\beta \\
& (\beta_i, 1 - z_i) : \text{SOS-1}, i = 1, ..., p \\
& z_i \in \{0, 1\}, \ \sum_{i=1}^{p} z_i \leq k \qquad (10) \\
& -\mathcal{M}_U \leq \beta_i \leq \mathcal{M}_U, i = 1, ..., p \\
& -\mathcal{M}_\ell \leq \zeta_i \leq \mathcal{M}_\ell, i = 1, ..., n \\
& \|\beta\|_1 \leq \mathcal{M}_\ell, \quad \|\zeta\|_1 \leq \mathcal{M}_\ell
\end{aligned}
$$

for the $p >> n$ case, (10) is more useful than (9) because it involves a quadratic form in n variables rather than p variables.

Further about MIO Formulations

Problem (10) is equivalent to the following variant of the best subset problem:

$$\begin{aligned} \min_{\beta} \quad & \frac{1}{2}\|y - X\beta\|_2^2 \\ s.t. \quad & \|\beta\|_\infty \leq \mathcal{M}_U, \|\beta\|_1 \leq \mathcal{M}_\ell \\ & \|X\beta\|_\infty \leq \mathcal{M}_U^\zeta, \|X\beta\|_1 \leq \mathcal{M}_\ell^\zeta \end{aligned} \tag{11}$$

Next, we will obtain estimates for $\mathcal{M}_U, \mathcal{M}_\ell, \mathcal{M}_U^\zeta, \mathcal{M}_\ell^\zeta$ such that Problem (11) is equivalent to Problem (1), which means we will find the optimal for Problem 1 once we solve Problem (11).

Specification of Parameters

Define that

$$\mu[k] := \max_{|I|=k} \max_{j \notin I} \sum_{i \in I} |\langle X_j, X_i \rangle|,$$

$$\mu := \mu[1] = \max_{i \neq j} |\langle X_i, X_j \rangle|.$$

Then we say X satisfies a restricted eigenvalue condition if

$$\lambda_{\min}(X_I' X_I) \geq \eta_k, \forall I \subset \{1, ..., p\} : |I| \leq k.$$

Specification of Parameters

Recall that X_j represent the columns of X; and we will use x_i to denote the rows of X. We ordered the corelations $|\langle X_j, y\rangle|$:

$$|\langle X_{(1)}, y\rangle| \geq |\langle X_{(2)}, y\rangle| \geq \cdots |\langle X_{(p)}, y\rangle|$$

Finally, we define

$$\|x\|_{1:k} = \max_J \sum_{j\in J} |x_{ij}|,$$

where $J \subset \{1, \cdots, p\}$.

Estimates for $\mathcal{M}_U, \mathcal{M}_\ell, \mathcal{M}_U^\zeta, \mathcal{M}_\ell^\zeta$

Theorem 1

For any $k \geq 1$ such that $\mu[k-1] < 1$ any optimal solution $\hat{\beta}$ to (1) satisfies:

$$(a) \qquad \|\hat{\beta}\|_1 \leq \frac{1}{1-\mu[k-1]} \sum_{j=1}^{k} |\langle X_{(j)}, y \rangle|$$

$$(b) \qquad \|X\hat{\beta}\|_\infty \leq \min \left\{ \frac{1}{\eta_k} \sqrt{\sum_{j=1}^{k} |\langle X_{(j)}, y \rangle|^2}, \frac{1}{\sqrt{\eta_k}} \|y\|_2 \right\}$$

$$(c) \qquad \|\hat{\beta}\|_1 \leq \min \left\{ \sum_{i=1}^{n} \|x_i\|_\infty \|\hat{\beta}\|_1, \sqrt{k}\|y\|_2 \right\}$$

$$(d) \qquad \|X\hat{\beta}\|_1 \leq \left(\max_{i=1,\ldots,n} \|x_i\|_{1:k} \right) \|\hat{\beta}\|_\infty$$

1 Background

2 Mordern Optimization Lenz
- Mix Integrate Optimization Formulations
- Discrete First Order Algorithm

3 Computational Results

4 Conclusions

Some Constraints on Objective Function

Consider the following optimization problem:

$$\min_{\beta} \; g(\beta) \; s.t. \; \|\beta\|_0 \leq k, \tag{28}$$

where $g(\beta) \geq 0$ is convex and has Lipschitz continuous gradient:

$$\|\nabla g(\beta) - \nabla g(\tilde{\beta})\| \leq \ell \|\beta - \tilde{\beta}\|. \tag{29}$$

We first consider the case in which $g(\beta) = \|\beta - c\|_2^2$ and have the following proposition.

Preparation before Using the Algorithm

Proposition 3

An optimal solution, denoted as $H_k(c)$, to the problem

$$\min_{\|\beta\|_0 \leq k} \quad \|\beta - c\|_2^2.$$

is

$$(H_k(c))_i = \begin{cases} c_i, & \text{if } i \in \{(1), ..., (k)\}; \\ 0, & \text{otherwise.} \end{cases}$$

Where $|c_{(1)}| \geq |c_{(2)}| \geq ... \geq |c_{(p)}|$ denote the ordered value of the absolute values of the vector c.

Preparation before Using the Algorithm

Proposition 4

For a convex function $g(\beta)$ satisfying some regulations and for any $L \geq \ell$ we have

$$g(\eta) \leq Q_L(\eta, \beta) := g(\beta) + \frac{L}{2}\|\eta - \beta\|_2^2 + \langle \nabla g(\beta), \eta - \beta \rangle \quad (33)$$

for all β, η with equality holding at $\beta = \eta$.

Preparation before Using the Algorithm

Applying Proposition 3 to the upper bound $Q_L(\boldsymbol{\eta}, \boldsymbol{\beta})$ in Proposition 4 we obtain

$$\begin{aligned}
\eta^* &= \underset{\|\eta\|_0 \leq k}{\operatorname{argmin}} \; Q_L(\eta, \beta) \\
&= \underset{\|\eta\|_0 \leq k}{\operatorname{argmin}} \left(\frac{L}{2} \left\| \eta - \left(\beta - \frac{1}{L} \nabla g(\beta) \right) \right\|_2^2 - \frac{1}{2L} \|\nabla g(\beta)\|_2^2 + g(\beta) \right) \\
&= \underset{\|\eta\|_0 \leq k}{\operatorname{argmin}} \left\| \eta - \left(\beta - \frac{1}{L} \nabla g(\beta) \right) \right\|_2^2 \\
&= H_k \left(\beta - \frac{1}{L} \nabla g(\beta) \right). \qquad (34)
\end{aligned}$$

In light of (34) we are now ready to present the algorithm to find a local optimal solution to Problem (28).

Algorithm 1

Input: $g(\beta), L, \epsilon$.
Output: A local optimal solution β^*.
Algorithm:

1. Initialize with $\beta_1 \in \mathbb{R}^p$ such that $\|\beta_1\|_0 \leq k$.
2. For $m \geq 1$, applying that $\beta = \beta_m$ to obtain β_{m+1} as:

$$\beta_{m+1} = H_k\left(\beta_m - \frac{1}{L}\nabla g(\beta_m)\right)$$

3. Repeat Step 2, until $\|\beta_{m+1} - \beta_m\|_2 \leq \epsilon$.
4. Let $\beta_m := (\beta_{m1}, ..., \beta_{mp})$ denote the current estimate and $I = \text{Supp}(\beta_m) := \{i : \beta_{mi} \neq 0\}$. Solve the continuous optimization problem

$$\beta^* = \min_{\beta, \beta_i = 0, i \notin I} g(\beta). \tag{36}$$

and let β^* be a minimizer.

Algorithm 2

1. Initialize with $\beta_1 \in \mathbb{R}^p$ such that $\|\beta_1\|_0 \leq k$.
2. For $m \geq 1$,

$$\eta_m = H_m\left(\beta_m - \frac{1}{L}\nabla g(\beta_m)\right), \beta_{m+1} \quad = \lambda_m \eta_m + (1-\lambda_m)\beta_m,$$

where λ is chosen as

$$\lambda_m \in \underset{\lambda}{\operatorname{argmin}} \quad g(\lambda\eta_m + (1-\lambda)\beta_m).$$

3. Repeat Step 2, until $\|\beta_{m+1} - \beta_m\|_2 \leq \epsilon$.
4. Let η_m denote the current estimate and let $I = \text{Supp}(\eta_m)$. Solve Problem (36) and let β^* be a minimizer.

Convergence Properties of Algorithm 1

Proposition 5

Consider $g(\beta)$ and ℓ as defined in (28), (29). Let $\beta_m, m \geq 1$ be the sequence generated by algorithm 1. Then

1. For any $L \geq \ell$, the sequence $g(\beta_m)$ satisfies

$$g(\beta_m) - g(\beta_{m+1}) \geq \frac{L-\ell}{2} \|\beta_{m+1} - \beta_m\|_2^2$$

   is decreasing and converges.
2. If $L > \ell$, then $\beta_{m+1} - \beta_m \to 0$ as $m \to \infty$.
3. If $L > \ell$ and $\|\liminf_{m\to\infty} \beta_m\|_0 = k$, then the sequence 1_m converges after finitely many iterations.
4. If $L > \ell$ and $\|\liminf_{m\to\infty} \beta_m\|_0 < k$, then $g(\beta) \to g(\beta^*)$ where $\beta^* \in \arg\min g(\beta)$ is an unconstrained minnimizer.

Applications of the Algorithm

Application to Least Squares:

$$g(\beta) = \frac{1}{2}\|y - X\beta\|_2^2, \nabla g(\beta) = -X'(y - X\beta).$$

Application to Least Absolute Deviation:

Since $g_1(\beta) = \|y - X\beta\|_1$ is non-smooth, we can use the minimax representation of $g_1(\beta)$ and apply our framework to find the optimal solution efficiently. Details about that will not be presented here.

1 Background

2 Mordern Optimization Lenz

3 Computational Results
- The Classical Overdetermined Case ($n > p$)
- High Dimensional Case ($p >> n$)
- Others

4 Conclusions

Experimental Model

The model is

$$y = X\beta^0 + \epsilon, x_i \sim N(0, \Sigma), \Sigma = (\sigma_{ij}), \epsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$$

The columns of X were normalized so that has mean zero and unit ℓ_2 norm. k_0 represents the number of non-zeros in β^0. Define the Signal-to-Noise Ratio (SNR) of the problem as

$$\text{SNR} = \frac{var(x'\beta^0)}{\sigma^2}.$$

Experimental Data

Synthetic Datasets:

Consider 4 different cases:

1. $\sigma_{ij} = \rho^{|i-j|}, k_0 \in \{5, 10\}$ and $\beta_i^0 = 1$ for k_0 equi-spaced values of i in the range $\{1, 2, ..., p\}$.
2. $\Sigma = I_{p\times p}, k_0 = 5$ and $\beta^0 = (1'_{5\times 1}, 0'_{(p-5)\times 1})' \in \mathbb{R}^p$.
3. $\Sigma = I_{p\times p}, k_0 = 10$, and $\beta_i^0 = \frac{1}{2} + (10 - \frac{1}{2})\frac{(i-1)}{k_0}, i = 1, ..., 10$, and $\beta_i^0 = 0, \forall i > 10$.
4. $\Sigma = \mathbf{I}_{p\times p}, k_0 = 6$ and $\beta^0 = (-10, -6, -2, 2, 6, 10, 0_{p-6})$.

Real Datasets:

1. Diabetes Datasets. $p = 64, n = 350$.
2. Leukemia Dataset. $n = 72, p = 1000$.

Obtaining Good Upper Bounds

To evaluate the performance of our methods in terms of obtaining high quality solutions for Problem (1), consider the following three algorithms:

1. Algorithm 2 with fifty random initializations. Choose the solution corresponding to the best to objective value.
2. MIO with cold start, formulation (9) with a time limit of 500 seconds.
3. MIO with warm start.

Obtaining Good Upper Bounds

Let f_* be the best optimal objective value among the solutions obtained by using all the algorithms for every instance, and f_{alg} denotes the value of the best subset objective function for method "alg", then we define the relative accuracy of the solution obtained by "alg" as:

$$\text{Relative Accuracy} = \frac{f_{\text{alg}} - f_*}{f_*}.$$

Obtaining Good Upper Bounds

The experiment result of the diabetes datasets is shown in the table below:

k	Discrete First Order		MIO Cold Start		MIO Warm Start	
	Accuracy	Time	Accuracy	Time	Accuracy	Time
9	0.1306	1	0.0036	500	0	346
20	0.1541	1	0.0042	500	0	77
49	0.1915	1	0.0015	500	0	87
57	0.1933	1	0	500	0	2

Figure: Quality of upper bounds for Problem (1) for the Diabetes dataset, for different values of k.

Statistical Performance

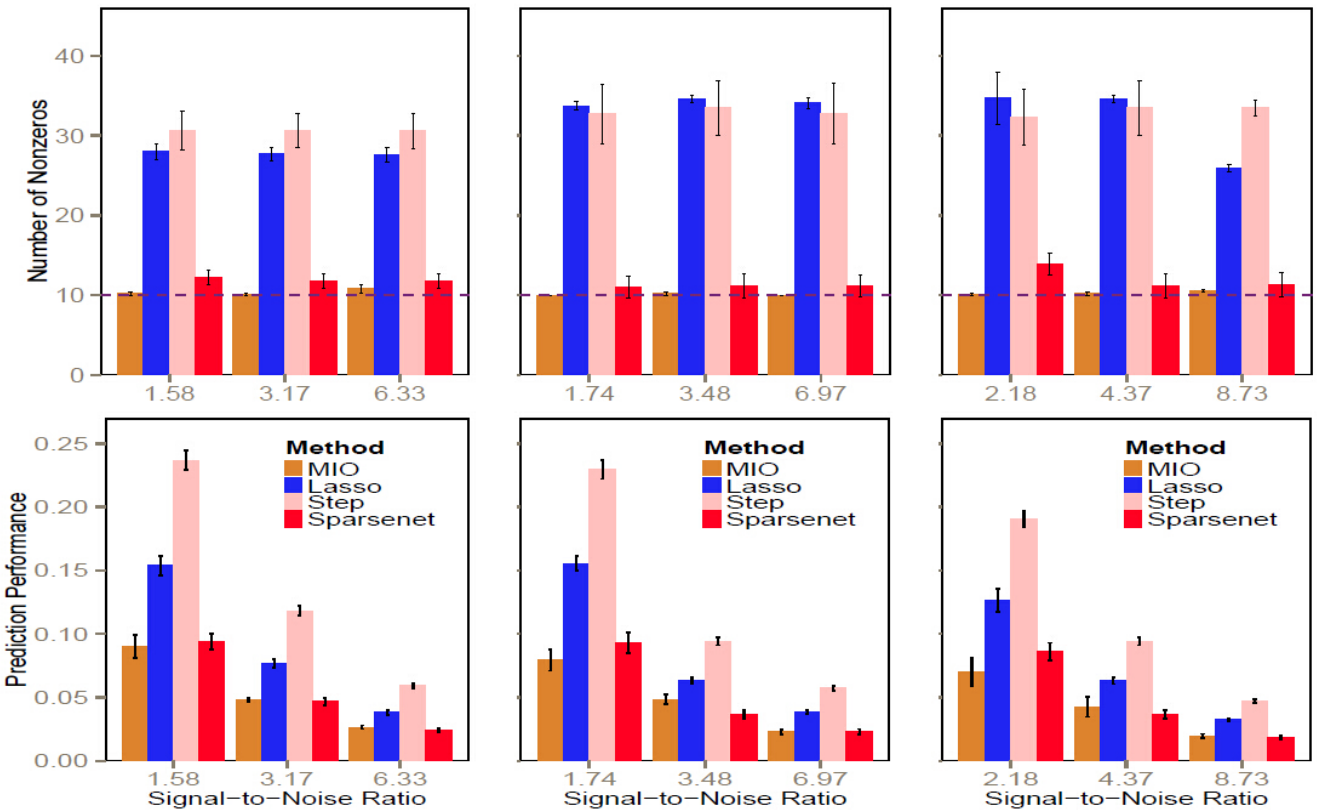

Figure: Result for data generated as per Example 1 with $n = 500$, $p = 100$.

1 Background

2 Mordern Optimization Lenz

3 Computational Results
- The Classical Overdetermined Case ($n > p$)
- High Dimensional Case ($p >> n$)
- Others

4 Conclusions

Obtaining Good Upper Bounds

The experiment result of the diabetes datasets is shown in the table below:

	k	Discrete First Order		MIO Cold Start		MIO Warm Start	
		Accuracy	Time	Accuracy	Time	Accuracy	Time
SNR = 3	5	0.1647	37.2	1.0510	500	0	72.2
	6	0.6152	41.1	0.2769	500	0	77.1
	7	0.7843	40.7	0.8715	500	0	160.7
	8	0.5515	38.8	2.1797	500	0	295.8
	9	0.7131	45.0	0.4204	500	0	96.0
SNR = 7	5	0.5072	45.6	0.7737	500	0	65.6
	6	1.3221	40.3	0.5121	500	0	82.3
	7	0.9745	40.9	0.7578	500	0	210.9
	8	0.8293	40.5	1.8972	500	0	262.5
	9	1.1879	44.2	0.4515	500	0	254.2

Figure: Quality of upper bounds for Problem (1) for the synthetic dataset of Example 2 with $n = 30, p = 2000$ and different values of SNR.

Obtaining Good Upper Bounds

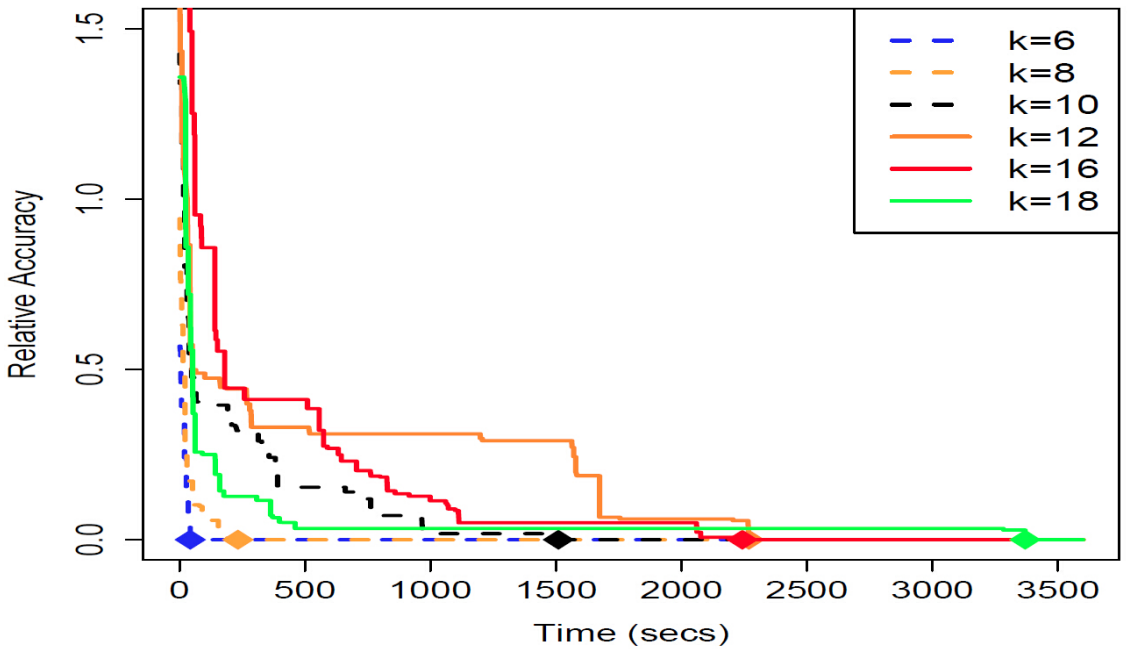

Figure: Behavior of MIO aided with warm start in obtaining good upper bounds over time for the Leukemia dataset $(n = 72; p = 1000)$. The vertical axis shows relative accuracy, i.e., $(f_t - f_*)/f_*$, where f_t is the objective value obtained after t seconds and f_* denotes the best objective value obtained by the method after 4000 seconds.

Difficult in Computation

- In the typical "high-dimensional" regime, with $p >> n$, it takes long time to certificate the global optimality as the lower bounds of the problem "evolve" slowly.

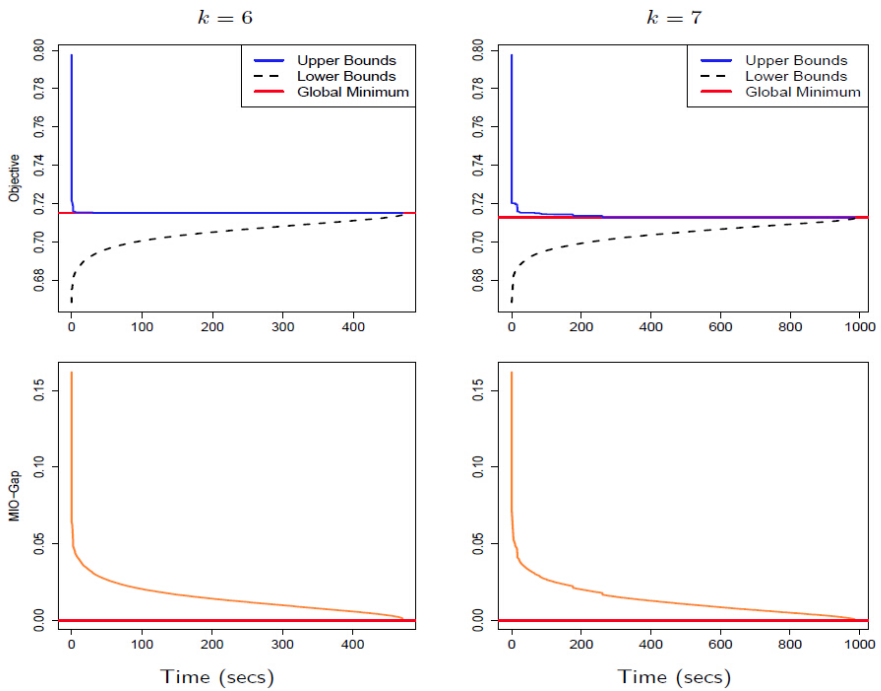

Figure: The evolution of the MIO formulation (8) for the diabetes dataset.

How to Make the Computation Quicker?

- To address this difficult, we add bounding boxes around a local solution, these restrictions guide the MIO in restricting its search space and enable the MIO to certify global optimality inside that bounding box.
- The bounding fox constraints to the MIO formulation:

$$\{\beta : \|X\beta - X\beta_0\|_1 \leq \mathcal{L}^{\zeta}_{\ell,loc}\} \cap \{\beta : \|\beta - \beta_0\|_1 \leq \mathcal{L}^{\beta}_{\ell,loc}\}$$

Bounding Box Formulations

Using the notation $\zeta = X\beta$ we have the following MIO formulation:

$$\begin{aligned}
\min_{\beta,z,\zeta} \quad & \frac{1}{2}\zeta^T\zeta - \langle X'y, \beta\rangle + \frac{1}{2}\|y\|_2^2 \\
s.t \quad & \zeta = X\beta, z_i \in \{0,1\}, \ \sum_{i=1}^{p} z_i \leq k \\
& (\beta_i, 1 - z_i) : \text{SOS-1}, i = 1, ..., p \qquad (54) \\
& -\mathcal{M}_U \leq \beta_i \leq \mathcal{M}_U, i = 1, ..., p \\
& -\mathcal{M}_\ell \leq \zeta_i \leq \mathcal{M}_\ell, i = 1, ..., n \\
& \|\beta\|_1 \leq \mathcal{M}_\ell, \quad \|\zeta\|_1 \leq \mathcal{M}_\ell \\
& \|\beta - \beta_0\|_1 \leq \mathcal{L}^{\beta}_{\ell,loc} \\
& \|\zeta - \zeta_0\|_1 \leq \mathcal{L}^{\zeta}_{\ell,loc}
\end{aligned}$$

Bounding Box Performance

Evolution of the MIO gap for (54), effect of type of bounding box (Example 1 with $\rho = 0.9, k_0 = 5, n = 50, p = 500$)

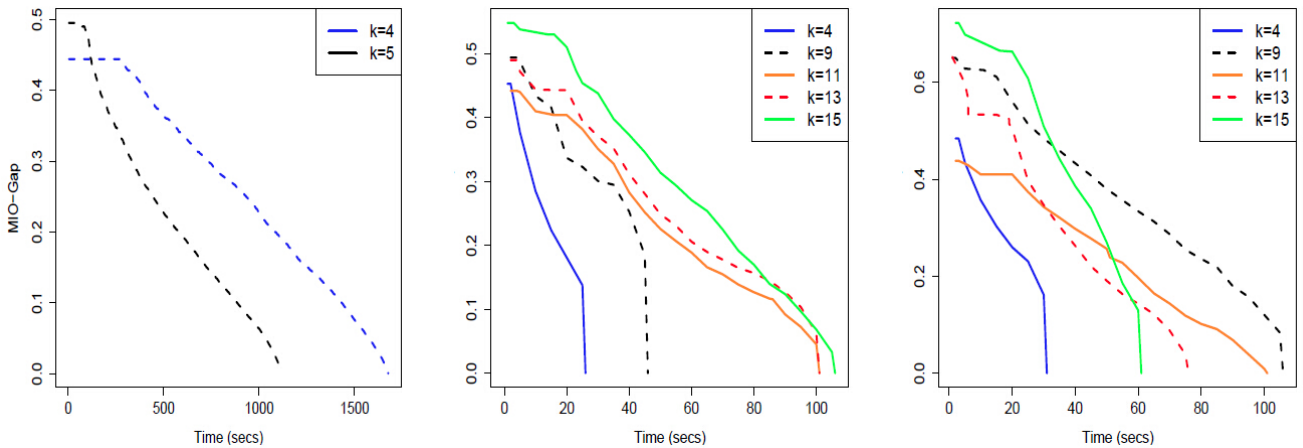

Figure: [Left Panel] $\mathcal{L}^{\zeta}_{\ell,loc} = 0.5\|X\beta_0\|_1, \mathcal{L}^{\zeta}_{\ell,loc} = \infty$ and SNR=1; [Middle Panel] $\mathcal{L}^{\zeta}_{\ell,loc} = \infty, \mathcal{L}^{\zeta}_{\ell,loc} = \|\beta_0\|_1/k$ and SNR=1; [Right Panel] $\mathcal{L}^{\zeta}_{\ell,loc} = \infty, \mathcal{L}^{\zeta}_{\ell,loc} = \|\beta_0\|_1/k$ and SNR=3.

Bounding Box Performance

Evolution of the MIO gap for (54), effect of bounding box radii $(n = 50, p = 500)$

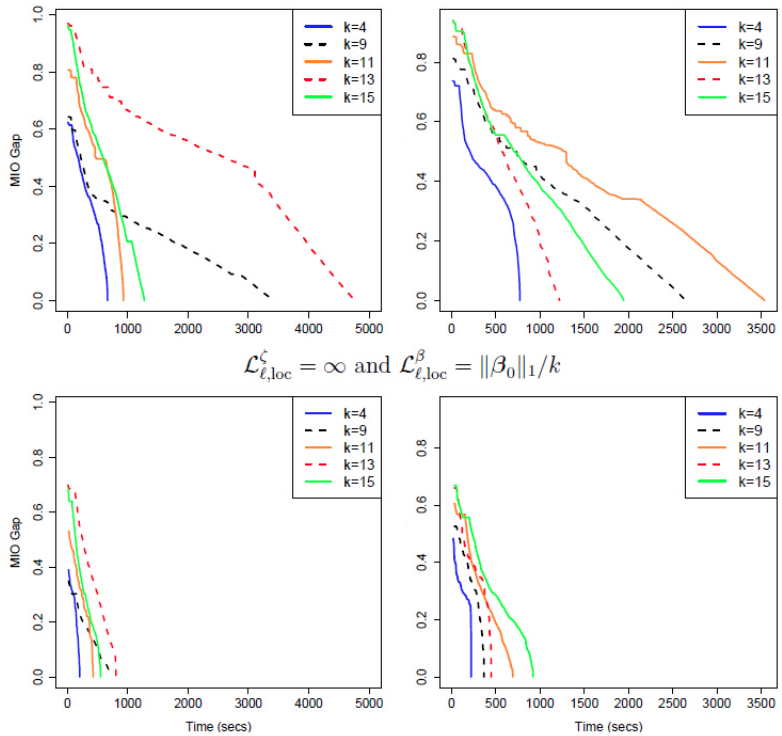

Figure: [Left Panel] SNR=1; [Right Panel] SNR=1.

Statistical Performance

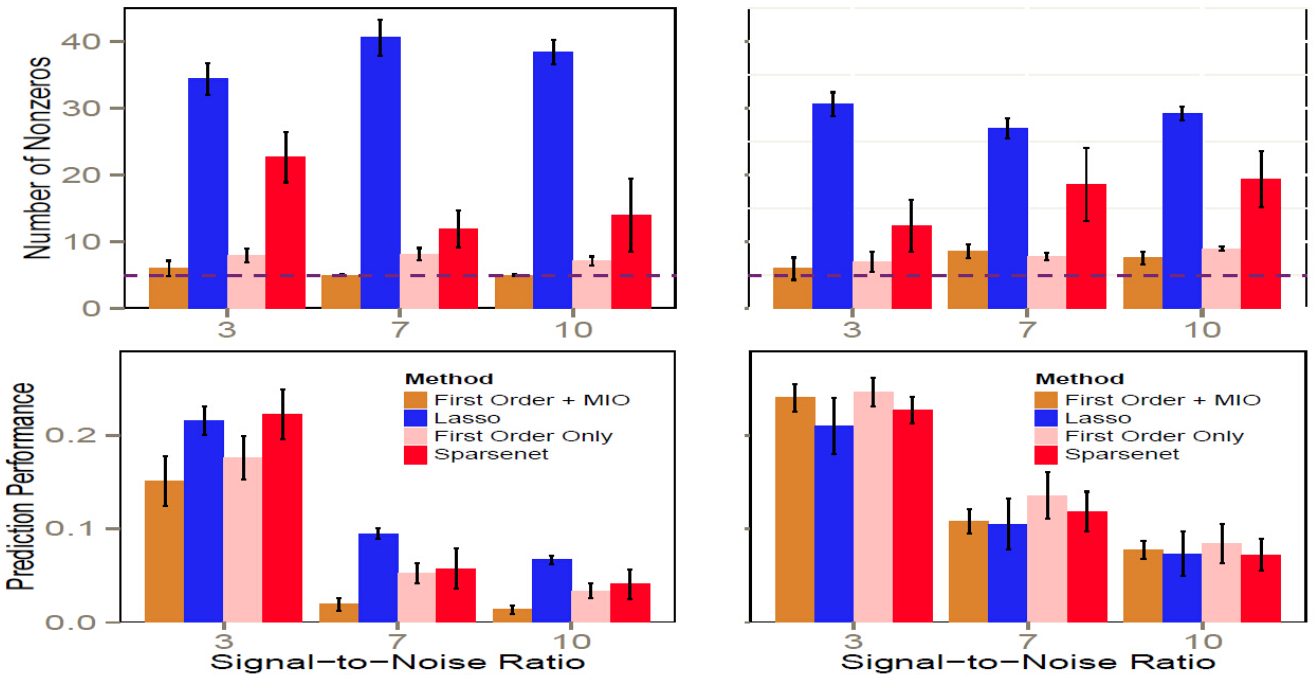

Figure: Left Panel: Example 1 with $n = 50, p = 1000, \rho = 0.8, k_0 = 5$; Right Panel: Example 2 with $n = 30, p = 1000$.

Statistical Performance

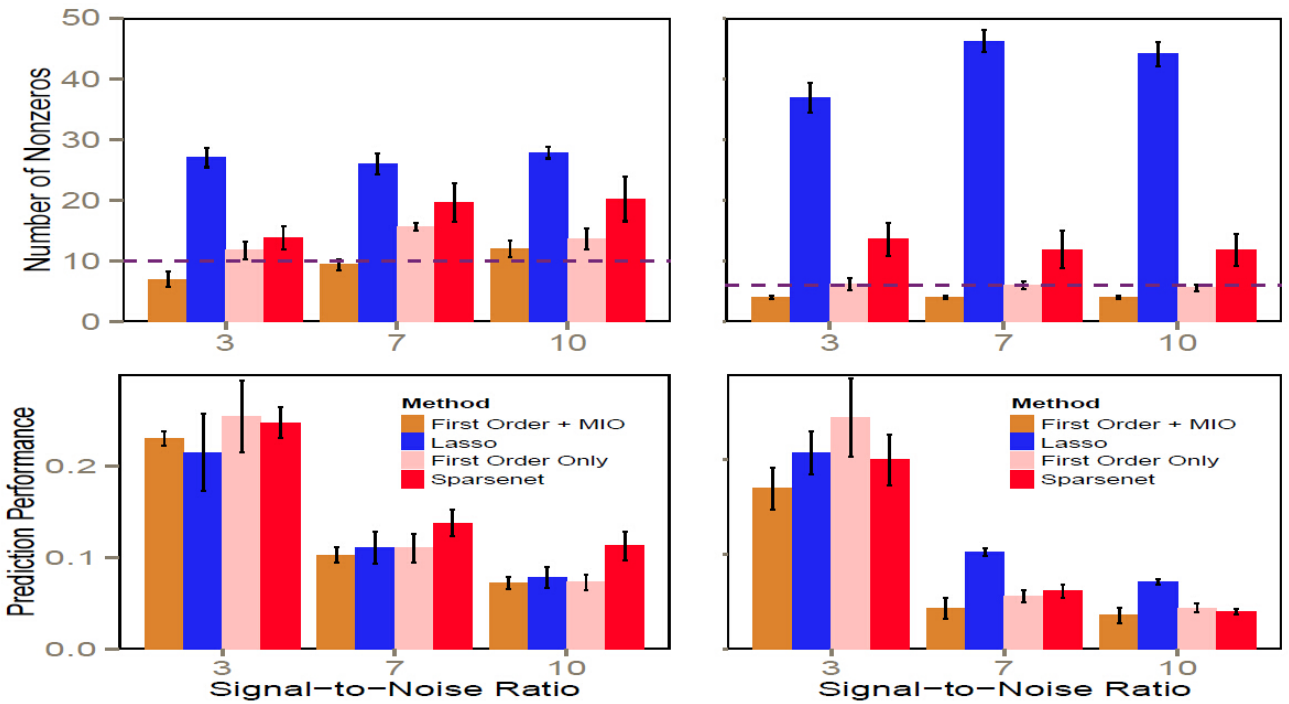

Figure: Left Panel: Example 3 with $n = 30, p = 1000$; Right Panel: Example 4 with $n = 50, p = 2000$.

1 Background

2 Mordern Optimization Lenz

3 Computational Results
- The Classical Overdetermined Case ($n > p$)
- High Dimensional Case ($p >> n$)
- Others

4 Conclusions

Results for Subset Selection with LAD Loss

All we discussed above are the results for subset selection with least squares loss, there is another kind of loss function called least absolute deviation loss (LAD). The results for the LAD case is similar with the least squares case and they are not presented here.

Conclusions

1. In terms of prediction error, the MIO performs the best, only to be marginally outperformed by Sparsenet in a few instances. This further illustrates the importance of using non-convex methods in sparse learning.
2. As the value of SNR increases, the predictive power of the methods improve, as expected.

Conclusions

1. The MIO best subset algorithm has a significant edge in detecting the correct sparsity structure compared to Lasso, Sparsenet and the stand-alone discrete first order method.
2. Lasso perform marginally better than MIO, as a predictive model for small values of SNR.
3. The solutions provided by the MIO approach significantly outperform other state of the art methods like Lasso in achieving sparse models with good predictive power.

Thank you!