Nonconvex penalized regression using depth-based penalty functions: multitask learning and support union recovery in high dimensions

Subho Majumdar
Snigdhansu Chatterjee

University of Minnesota, School of Statistics

Motivation

- Modelling multiple quantitative traits based on a large number of Single Nucleotide Polymorphisms;
- Predicting interaction between genes in a certain biological pathway and those in other pathways in the organisms.

In a nutshell

Consider the multitask linear regression model:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E}$$

where $\mathbf{Y} \in \mathbb{R}^{n \times q}$ is the matrix of responses, and $\mathbf{E}$ is $n \times q$ the noise matrix: each row of which is drawn from $\mathcal{N}_q(\mathbf{0}_q, \mathbf{\Sigma})$ for a $q \times q$ positive definite matrix $\mathbf{\Sigma}$.

We are interested in sparse estimates of the coefficient matrix $\mathbf{B}$ through solving penalized regression problems of the form

$$\min_{\mathbf{B}} \mathrm{Tr}\{(\mathbf{Y} - \mathbf{XB})^T(\mathbf{Y} - \mathbf{XB})\} + P_\lambda(\mathbf{B}). \tag{1}$$

Sparse estimators in multivariate regression

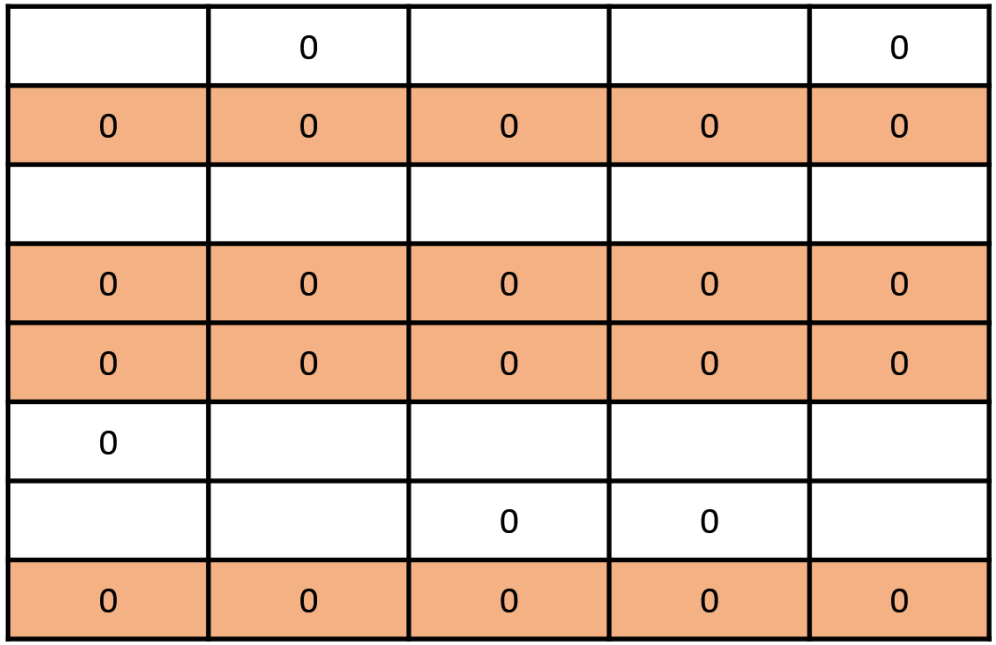
	0			0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0				
		0	0	
0	0	0	0	0

Sparse estimators in multivariate regression

	0			0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0				
		0	0	
0	0	0	0	0

Two types of sparsity are possible: **between-row**, i.e. determining which predictors are *simultaneously* important for all responses, and **within-row**, which responses are affected by a overall non-zero predictor (right panel).

Sparse estimators in multivariate regression



Two types of sparsity are possible: **between-row**, **i.e. determining which predictors are *simultaneously* important for all responses**, and **within-row**, which responses are affected by a overall non-zero predictor (right panel).

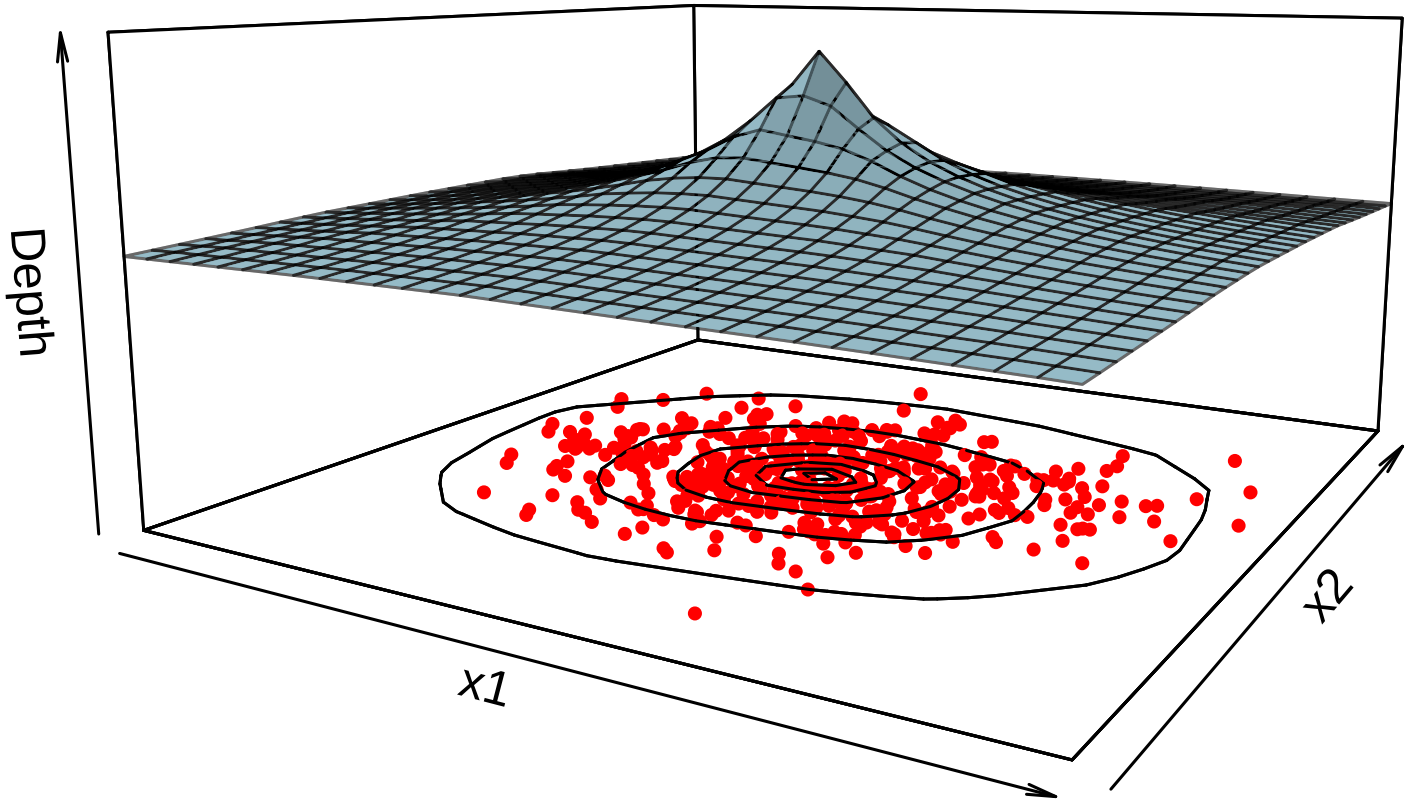
Sparse estimators in multivariate regression

	0			0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0				
		0	0	
0	0	0	0	0

Two types of sparsity are possible: **between-row**, i.e. determining which predictors are *simultaneously* important for all responses, and **within-row, which responses are affected by a overall non-zero predictor**.

What is depth?

Example: 500 points from $\mathcal{N}_2((0,0)^T, \text{diag}(2,1))$



A scalar measure of how much inside a point is with respect to a data cloud

Formal definition

For any multivariate distribution $F = F_{\mathbf{X}}$, the depth of a point $\mathbf{x} \in \mathbb{R}^p$, say $D(\mathbf{x}, F_{\mathbf{X}})$ is any real-valued function that provides a 'center outward ordering' of $\mathbf{x}$ with respect to F (Zuo and Serfling, 2000).

Desirable properties (Liu, 1990)

(P1) *Affine invariance*: $D(A\mathbf{x} + \mathbf{b}, F_{A\mathbf{X}+\mathbf{b}}) = D(\mathbf{x}, F_{\mathbf{X}})$

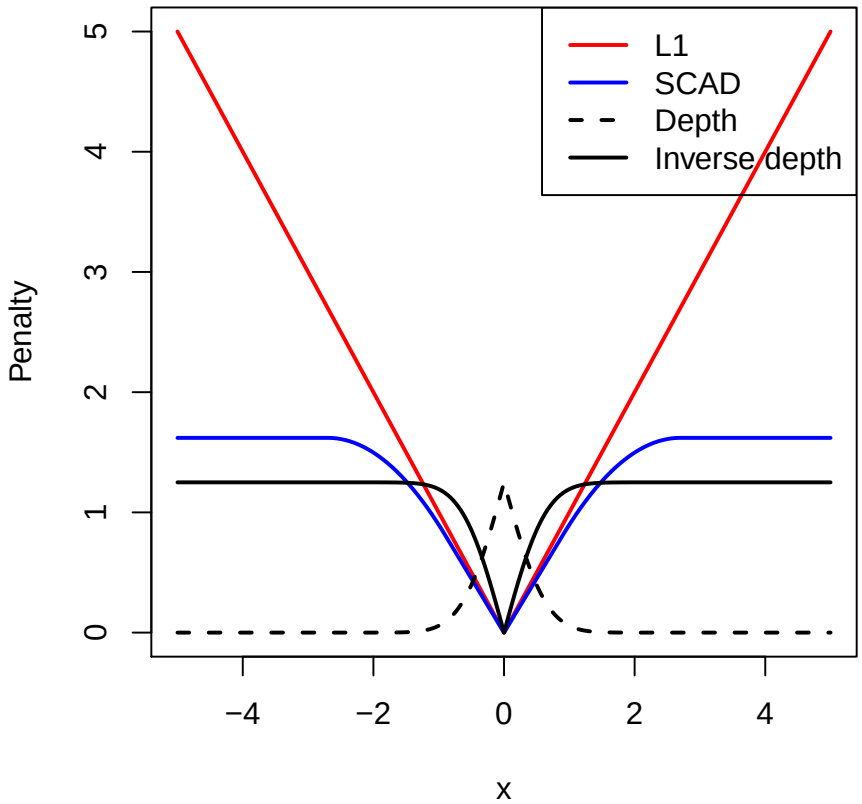
(P2) *Maximality at center*: $D(\theta, F_{\mathbf{X}}) = \sup_{\mathbf{x} \in \mathbb{R}^p} D(\mathbf{x}, F_{\mathbf{X}})$ for $F_{\mathbf{X}}$ with center of symmetry θ, the *deepest point* of $F_{\mathbf{X}}$.

(P3) *Monotonicity w.r.t. deepest point*: $D(\mathbf{x}; F_{\mathbf{X}}) \leq D(\theta + a(\mathbf{x} - \theta), F_{\mathbf{X}})$

(P4) *Vanishing at infinity*: $D(\mathbf{x}; F_{\mathbf{X}}) \to \mathbf{0}$ as $\|\mathbf{x}\| \to \infty$.

Why use depth?

What produces sparse solutions? **Non-differentiability at 0.**



Lasso penalty is unbounded, and produces biased solutions for non-zero coefficients. Nonconvex penalty functions remedy this (Fan and Li (2001): SCAD penalty).

Our estimator

We incorporate measures of data depth as a row-level penalty function. Specifically, we estimate the coefficient matrix $\mathbf{B}$ by solving the following constrained optimization problem:

$$\hat{\mathbf{B}} = \arg\min_{\mathbf{B}} \left[\mathrm{Tr}\{(\mathbf{Y} - \mathbf{XB})^T(\mathbf{Y} - \mathbf{XB})\} + \lambda \sum_{j=1}^{p} D^-(\mathbf{b}_j, F) \right]$$

where $D^-(\mathbf{x}, F)$ is an *inverse depth* function. This can be any *nonnegative-valued monotonically decreasing transformation* on a depth function, e.g. $D^-(\mathbf{x}, F) := \max_{\mathbf{x}} D(\mathbf{x}, F) - D(\mathbf{x}, F)$ and $D^-(\mathbf{x}, F) := \exp(-D(\mathbf{x}, F))$.

Simplifications

- Assume F to be spherically symmetric. This makes D^- a function of the row-norm $r_j = \|\mathbf{b}_j\|_2$: $D^-(\mathbf{b}_j, F) = p_F(r_j)$. This is essential to enforce row-level sparsity (i.e. make entire rows to be zero).
- Use the first order Taylor approximation around a 'close enough' point r_j^* instead of $p_F(r_j)$. This is local linear approximation (Zou and Li, 2008):

$$p_F(r_j) \simeq p_F(r_j^*) + p'_F(r_j^*)(r_j - r_j^*)$$

Thus the modified solution is:

$$\hat{\mathbf{B}}^{(1)} = \arg\min_{\mathbf{B}} \left[\mathrm{Tr}\{(\mathbf{Y} - \mathbf{XB})^T(\mathbf{Y} - \mathbf{XB})\} + \lambda \sum_{j=1}^{p} p'_F(r_j^*) r_j \right]$$

The close enough matrix $\mathbf{B}^*$ to start from can be the least squares estimate. We call this Local Approximation by Row Norm (LARN).

Simplifications

- Assume F to be spherically symmetric. This makes D^- a function of the row-norm $r_j = \|\mathbf{b}_j\|_2$: $D^-(\mathbf{b}_j, F) = p_F(r_j)$. This is essential to enforce row-level sparsity (i.e. make entire rows to be zero).
- Use the first order Taylor approximation around a 'close enough' point r_j^* instead of $p_F(r_j)$. This is local linear approximation (Zou and Li, 2008):

$$p_F(r_j) \simeq p_F(r_j^*) + p'_F(r_j^*)(r_j - r_j^*)$$

Thus the modified solution is:

$$\hat{\mathbf{B}}^{(1)} = \arg\min_{\mathbf{B}} \left[\mathrm{Tr}\{(\mathbf{Y} - \mathbf{XB})^T(\mathbf{Y} - \mathbf{XB})\} + \lambda \sum_{j=1}^{p} p'_F(r_j^*) r_j \right]$$

The close enough matrix $\mathbf{B}^*$ to start from can be the least squares estimate. We call this Local Approximation by Row Norm (LARN).

- **Oracle property:** When $\hat{\mathbf{B}}^{(1)} = (\hat{\mathbf{B}}_{01}^T, \hat{\mathbf{B}}_{00}^T)^T$ (with the component matrix having dimensions $p_1 \times q$ and $p - p_1 \times q$, respectively) as $n \to \infty$: **(a)**

  $P[\text{vec}(\hat{\mathbf{B}}_{00}) = \mathbf{0}_{(p-p_1)q}] \to 1$;
  (b) $\sqrt{n}(\text{vec}(\hat{\mathbf{B}}_{01}) - \text{vec}(\mathbf{B}_{01})) \rightsquigarrow \mathcal{N}_{p_1 q}(\mathbf{0}_{p_1 q}, \mathbf{\Sigma} \otimes \mathbf{C}_{11}^{-1})$ under certain conditions. Here $\mathbf{C}_{11}$ is the first $p_1 \times p_1$ block in $\mathbf{C}$.

- **Near-minimax optimal performance:** approximately achieves lowest possible maximum risk for independent responses and othogonal predictors;

- **Within-row sparsity:** It is possible to consistently recover the zeros within the non-zero rows by setting a common threshold for all elements of $\hat{\mathbf{B}}^{(1)}$.

- **Oracle property:** When $\hat{\mathbf{B}}^{(1)} = (\hat{\mathbf{B}}_{01}^T, \hat{\mathbf{B}}_{00}^T)^T$ (with the component matrix having dimensions $p_1 \times q$ and $p - p_1 \times q$, respectively) as $n \to \infty$: **(a)**

  $P[\text{vec}(\hat{\mathbf{B}}_{00}) = \mathbf{0}_{(p-p_1)q}] \to 1$;
  (b) $\sqrt{n}(\text{vec}(\hat{\mathbf{B}}_{01}) - \text{vec}(\mathbf{B}_{01})) \rightsquigarrow \mathcal{N}_{p_1 q}(\mathbf{0}_{p_1 q}, \mathbf{\Sigma} \otimes \mathbf{C}_{11}^{-1})$ under certain conditions. Here $\mathbf{C}_{11}$ is the first $p_1 \times p_1$ block in $\mathbf{C}$.

- **Near-minimax optimal performance:** approximately achieves lowest possible maximum risk for independent responses and othogonal predictors;

- **Within-row sparsity:** It is possible to consistently recover the zeros within the non-zero rows by setting a common threshold for all elements of $\hat{\mathbf{B}}^{(1)}$.

- **Oracle property:** When $\hat{\mathbf{B}}^{(1)} = (\hat{\mathbf{B}}_{01}^T, \hat{\mathbf{B}}_{00}^T)^T$ (with the component matrix having dimensions $p_1 \times q$ and $p - p_1 \times q$, respectively) as $n \to \infty$: **(a)**

  $P[\text{vec}(\hat{\mathbf{B}}_{00}) = \mathbf{0}_{(p-p_1)q}] \to 1$;
  (b) $\sqrt{n}(\text{vec}(\hat{\mathbf{B}}_{01}) - \text{vec}(\mathbf{B}_{01})) \rightsquigarrow \mathcal{N}_{p_1 q}(\mathbf{0}_{p_1 q}, \mathbf{\Sigma} \otimes \mathbf{C}_{11}^{-1})$ under certain conditions. Here $\mathbf{C}_{11}$ is the first $p_1 \times p_1$ block in $\mathbf{C}$.

- **Near-minimax optimal performance:** approximately achieves lowest possible maximum risk for independent responses and othogonal predictors;

- **Within-row sparsity:** It is possible to consistently recover the zeros within the non-zero rows by setting a common threshold for all elements of $\hat{\mathbf{B}}^{(1)}$.

Computation

- We use a block coordinate descent algorithm (Li et al., 2015) to iteratively obtain $\hat{\mathbf{B}}^{(1)}$, starting from the OLS solution;
- Use cross-validation to select the best tuning parameter λ, as well as the optimal common threshold;
- Given a fixed λ, first compute $\mathbf{B}^{(1)}$ corresponding to it, then cycle through all possible threshold values;
- Use warm starts to get good starting values when dealing with a range of λ-s.

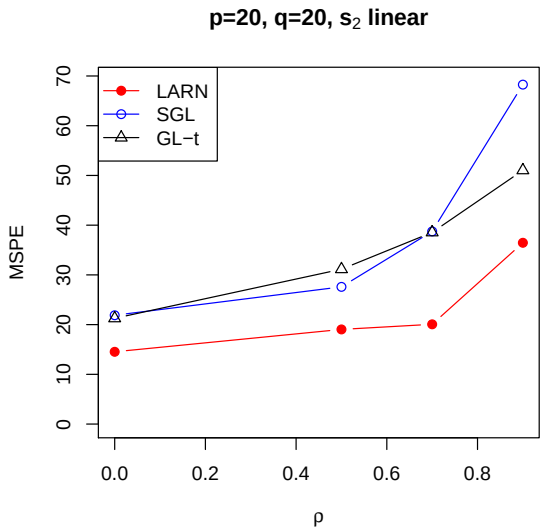
Simulation: setup (Rothman et al., 2010)

- Rows of $\mathbf{X}$ as $n = 50$ independent draws from $\mathcal{N}(\mathbf{0}_p, \mathbf{\Sigma}_X)$, where the $(i,j)^{\text{th}}$ element of $\mathbf{\Sigma_X}$ is given by $0.7^{|i-j|}$;
- Rows of $\mathbf{E}$ are generated as independent draws from $\mathcal{N}(\mathbf{0}_q, \mathbf{\Sigma})$: with $(i,j)^{\text{th}}$ element of $\mathbf{\Sigma}$ is given by 0.7^{ρ};
- Finally, we obtain the three $p \times q$ matrices: $\mathbf{W} \sim N(2,1)$ elements, $\mathbf{K} \sim$ Bernoulli(0.3) elements, and $\mathbf{Q} \sim$ Bernoulli $1/8$ rows. Following this, we multiply individual elements of these matrices (denoted by $(*)$) to obtain a sparse $\mathbf{B}$:

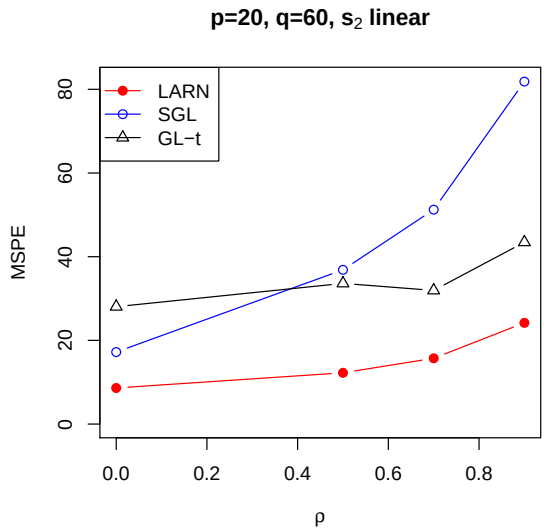
$$\mathbf{B} = \mathbf{W} * \mathbf{K} * \mathbf{Q}$$

- Replicate simulation 100 times for three settings of (p,q), and $\rho \in \{0, .5, .7, .9\}$.
- Methods compared:
  (a) GL-t = Group lasso with thresholding,
  (b) SGL = Sparse Group Lasso (Simon et al., 2013).

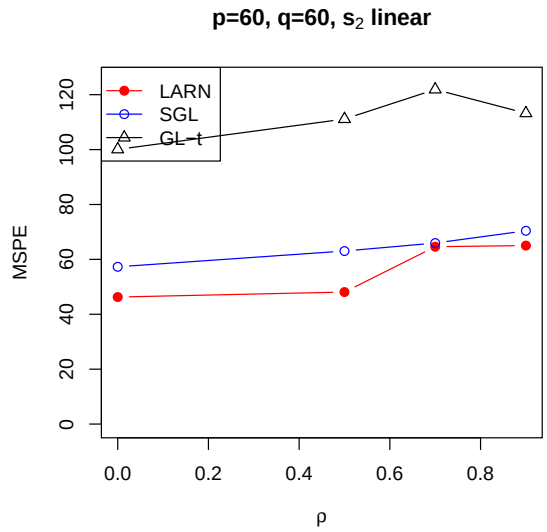
Simulation: accuracy comparison



(a)



(b)



(c)

$$\text{MSTE (mean squared testing error)} = \frac{1}{pq}\text{Tr}\left[(\mathbf{Y}_{test} - \mathbf{X}_{test})(\mathbf{Y}_{test} - \mathbf{X}_{test})^T\right]$$

with $(\mathbf{Y}_{test}, \mathbf{X}_{test})$ generated from the same simulation setup but using the same true $\mathbf{B}$.

Simulation: computation time comparison

ρ	GL-t	SGL	LARN
(a) $p = 20, q = 20$			
0.9	0.77/0.83	0.92/0.99	0.91/0.92
0.7	0.81/0.83	0.91/0.99	0.89/0.93
0.5	0.78/0.79	0.89/0.99	0.88/0.92
0.0	0.85/0.78	0.90/0.99	0.90/0.91
(b) $p = 20, q = 60$			
0.9	0.90/0.66	0.95/0.97	0.89/0.92
0.7	0.91/0.70	0.93/0.96	0.90/0.92
0.5	0.80/0.69	0.94/0.98	0.93/0.92
0.0	0.85/0.68	0.93/0.97	0.91/0.92
(c) $p = 60, q = 60$			
0.9	0.57/0.79	0.68/0.99	0.85/0.93
0.7	0.50/0.79	0.64/0.99	0.83/0.93
0.5	0.54/0.81	0.64/0.99	0.85/0.93
0.0	0.58/0.79	0.63/0.99	0.84/0.93

Table: True Positive and True negative (TP/TN) rates for the three methods

Results

Setting	GL-t	SGL	LARN
(a)	332	490	209
(b)	676	52	328
(c)	4994	39760	3883

Table: Total runtimes in seconds for SGL and LARN algorithms for the three simulation settings

- SGL is a single-response method. To adapt it in our scenario, we apply the method on vec($\mathbf{Y}$) and $\mathbf{X} \otimes \mathbf{I}_q$: hence high computation times for large data dimensions.
- GL-t is an unweighted version of LARN. Still LARN is faster than its unweighted counterpart, indicating faster convergence.

Application on gene expression data

- Gene expression data of *Arabidopsis thaliana*: $n = 118$;
- Responses: expressions of $q = 40$ genes in two pathways for biosynthesis of isoprenoid compounds;
- Predictors: expressions of $p = 795$ other genes corresponding to 56 other pathways;
- Compare with SGL and GL-t: LARN has least prediction error and proportion of non-zero entries in coefficient matrix;
- Top gene-network connections all previously reported in literature.

Coming soon

- Extension to Generalized Linear Models;
- Ultra-high dimension extension: p varies with n.

ArXiV: **https://arxiv.org/abs/1610.07540**

References

J. Fan and R. Li. Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties. *J. Amer. Statist. Assoc.*, 96:1348–1360, 2001.

Y. Li, B. Nan, and J. Zhu. Multivariate Sparse Group Lasso for the Multivariate Multiple Linear Regression with an Arbitrary Group Structure. *Biometrics*, 71:354–363, 2015.

R.Y. Liu. On a notion of data depth based on random simplices. *Ann. Statist.*, 18:405–414, 1990.

A. J. Rothman, E. Levina, and J. Zhu. Sparse Multivariate Regression With Covariance Estimation. *J. Comp. Graph. Stat.*, 19:947–962, 2010.

N. Simon, J. Friedman, T. Hastie, and R. Tibshirani. A Sparse-Group Lasso. *J. Comp. Graph. Stat.*, 22: 231–245, 2013.

H. Zou and R. Li. One-step sparse estimates in nonconcave penalized likelihood models. *Ann. Statist.*, 36: 1509–1533, 2008.

Y. Zuo and R. Serfling. General notions of statistical depth functions. *Ann. Statist.*, 28-2:461–482, 2000.

THANK YOU!

Acknowledgement: NSF grants IIS-1029711, SES-0851705;
NASA grant 1502546