Two Papers on Sparse Feature Selection in Multivariate Regression

Gregory Faletto

Department of Data Sciences and Operations
Statistics Group
University of Southern California Marshall School of Business

DSO 607, April 1 2019

Outline

Setup and Notation for Multivariate Linear Regression (single observation)

Setup and Notation for Multivariate Linear Regression (single observation)

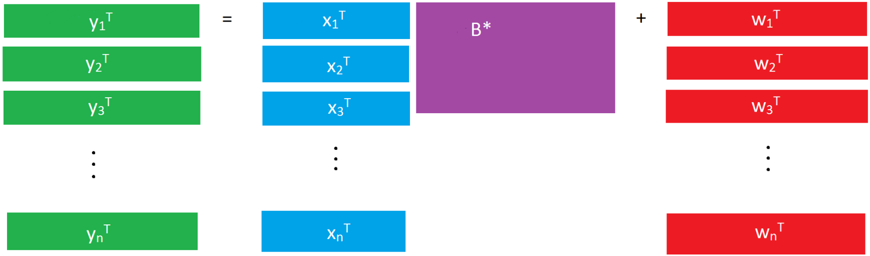
Formulation for individual response $\boldsymbol{y}_i$:

$$\underbrace{\boldsymbol{y}_i^T}_{1 \times K} = \underbrace{\boldsymbol{x}_i^T}_{1 \times p} \underbrace{\boldsymbol{B}^*}_{p \times K} + \underbrace{\boldsymbol{w}_i^T}_{1 \times K} \tag{1}$$

where

- $\boldsymbol{y}_i \in \mathbb{R}^K$ is a K-vector of responses (rather than a scalar response as in univariate linear regression),
- $\boldsymbol{x}_i \in \mathbb{R}^p$ is a p-vector of predictors associated with $\boldsymbol{y}_i$ (same as in univariate regression),
- $\boldsymbol{B}^* \in \mathbb{R}^{p \times K}$ is a $p \times K$ matrix of coefficients (rather than a p-vector of coefficients as in univariate regression),
- $\boldsymbol{w}_i \in \mathbb{R}^K$ is a K-vector of random errors.

Setup and Notation for Multivariate Linear Regression (full data set)



Setup for Multivariate Linear Regression (full data set)

Formulation for full data set:

$$\underbrace{\boldsymbol{Y}}_{n \times K} = \underbrace{\boldsymbol{X}}_{n \times p} \underbrace{\boldsymbol{B}^*}_{p \times K} + \underbrace{\boldsymbol{W}}_{n \times K} \tag{2}$$

where

- $\boldsymbol{Y} \in \mathbb{R}^{n \times K} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)^T$ is an $n \times K$ matrix of responses (row i contains response $\boldsymbol{y}_i$),
- $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ is an $n \times p$ matrix of predictors (same as in univariate regression),
- $\boldsymbol{B}^* \in \mathbb{R}^{p \times K}$ is a $p \times K$ matrix of coefficients,
- $\boldsymbol{W} \in \mathbb{R}^{n \times K} = (\boldsymbol{w}_i, \ldots, \boldsymbol{W}_n)^T$ is an $n \times K$ matrix of random errors. The error vectors are assumed to be IID with $\mathbb{E}(\boldsymbol{w}_i) = \mathbf{0}$.

Some Motivating Examples

- Predicting both Math and Verbal SAT score as a function of parental income, extracurricular activity participation, etc.
- Responses of several measures of health to a drug in the presents of other covariates (age, weight, gender, etc.)
- Prediction of the prices of several stocks as a function of economic indicators
- Used to find sets of genes that are expressed in different forms of cancer in Chen, Chan, and Stenseth (2012)

Some Motivating Examples

- Predicting both Math and Verbal SAT score as a function of parental income, extracurricular activity participation, etc.
- Responses of several measures of health to a drug in the presents of other covariates (age, weight, gender, etc.)
- Prediction of the prices of several stocks as a function of economic indicators
- Used to find sets of genes that are expressed in different forms of cancer in Chen, Chan, and Stenseth (2012)

- Sometimes called "multi-task learning" in machine learning
- Won't go through example of gene expression because it's a different paradigm from the usual case we'll discuss

Outline

Meaning of "Support Union Recovery"

- Feature selection in multivariate setting
- Sparsity assumption: column k of coefficient matrix $\boldsymbol{B}^*$ has nonzero entries in a subset

$$S_k := \{i \in \{1, \ldots, p\} \mid \beta^*_{ik} \neq 0\} \tag{3}$$

  of size $s_k := |S_k|$.
- We seek to *recover* the *union* of the *supports*: the set

$$S := \bigcup_{k=1}^{K} S_k$$

  of size $s := |S|$.

Meaning of "Support Union Recovery"

- Feature selection in multivariate setting
- Sparsity assumption: column k of coefficient matrix $\boldsymbol{B}^*$ has nonzero entries in a subset

$$S_k := \{i \in \{1, \ldots, p\} \mid \beta^*_{ik} \neq 0\} \tag{3}$$

of size $s_k := |S_k|$.
- We seek to *recover* the *union* of the *supports*: the set

$$S := \bigcup_{k=1}^{K} S_k$$

of size $s := |S|$.

- Responses are related, so we expect supports S_k should have some overlap, and that the relationship between the supports and responses (i.e. the coefficients, or the vectors in the columns of $\boldsymbol{B}$ are related).

Block Regularization

- Used in many settings when goal is to choose whether or not to include groups of variables, rather than considering them individually.
- Example: ℓ_1/ℓ_q norm of a matrix takes the sum of the ℓ_q norms of each row:

$$\|\boldsymbol{B}\|_{\ell_1/\ell_q} := \sum_{i=1}^{p} \left(\sum_{j=1}^{K} |\beta_{ij}|^q \right)^{1/q} = \sum_{i=1}^{p} \|\beta_i\|_q \tag{4}$$

where $(\beta_{ik})_{1 \leq i \leq p, 1 \leq k \leq K}$ are the entries of $\boldsymbol{B}$.

- This paper makes use of the ℓ_1/ℓ_2 norm

$$\|\boldsymbol{B}\|_{\ell_1/\ell_2} := \sum_{i=1}^{p} \sqrt{\sum_{j=1}^{K} \beta_{ij}^2} = \sum_{i=1}^{p} \|\beta_i\|_2 \tag{5}$$

Multivariate Group Lasso

- We solve the following optimization problem to estimate $\boldsymbol{B}^*$:

$$\underset{\boldsymbol{B} \in \mathbb{R}^{p \times K}}{\arg\min} \left\{ \frac{1}{2n} \| \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{B} \|_F^2 + \lambda_n \| \boldsymbol{B} \|_{\ell_1/\ell_q} \right\} \tag{6}$$

  where $\|\cdot\|_F$ denotes the Frobenius norm:

$$\|A\|_F := \sqrt{\sum_{i,j} A_{i,j}^2}.$$

- Penalizing the ℓ_1/ℓ_1 norm is equivalent to simply solving K separate lasso solutions.
- Penalizing the ℓ_1/ℓ_2 nor groups rows together; the authors call this the **multivariate group lasso**.

Multivariate Group Lasso

- We solve the following optimization problem to estimate $\boldsymbol{B}^*$:

$$\arg\min_{\boldsymbol{B}\in\mathbb{R}^{p\times K}} \left\{ \frac{1}{2n}\|\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{B}\|_F^2 + \lambda_n\|\boldsymbol{B}\|_{\ell_1/\ell_q} \right\} \tag{6}$$

where $\|\cdot\|_F$ denotes the Frobenius norm:

$$\|A\|_F := \sqrt{\sum_{i,j} A_{i,j}^2}.$$

- Penalizing the ℓ_1/ℓ_1 norm is equivalent to simply solving K separate lasso solutions.
- Penalizing the ℓ_1/ℓ_2 nor groups rows together; the authors call this the **multivariate group lasso**.

- Note that when $K = 1$ this is exactly the lasso (regardless of q):

$$\|\boldsymbol{B}\|_{\ell_1/\ell_1} := \sum_{i=1}^{p} \left(\sum_{j=1}^{K} |\beta_{ij}|^1 \right)^{1/1} = \sum_{i=1}^{p} |\beta_i| \tag{7}$$

- When $q = 1$ the sum decouples (show on board)

Multivariate Group Lasso (continued)

- The Frobenius norm is convex; therefore the optimization problem (6) is convex and can be solved efficiently. (In particular, it is a second-order cone program (SOCP) and can be solved efficiently with interior point methods.)

$$\underset{\boldsymbol{B} \in \mathbb{R}^{p \times K}}{\arg\min} \left\{ \frac{1}{2n} \| \boldsymbol{Y} - \boldsymbol{X}\boldsymbol{B} \|_F^2 + \lambda_n \| \boldsymbol{B} \|_{\ell_1/\ell_q} \right\} \tag{6}$$

Main Assumptions

- (A1) *Bounded eigenspectrum:* There exist fixed constants $C_{\min} > 0$ and $C_{\max} < +\infty$ such that all eigenvalues of the $s \times s$ matrix Σ_{SS} (the covariance matrix of the true support) are contained in the interval $[C_{\min}, C_{\max}]$.
- (A2) *Irrepresentable Condition:* There exists a fixed parameter $\gamma \in (0, 1]$ such that

$$\|\Sigma_{S^c S}\left(\Sigma_{SS}\right)^{-1}\|_\infty \leq 1 - \gamma.$$

- (A3) *Self-incoherence:* There exists some $D_{\max} < +\infty$ such that

$$\|\left(\Sigma_{SS}\right)^{-1}\|_\infty \leq D_{\max}.$$

Main Assumptions

- (A1) *Bounded eigenspectrum:* There exist fixed constants $C_{\min} > 0$ and $C_{\max} < +\infty$ such that all eigenvalues of the $s \times s$ matrix Σ_{SS} (the covariance matrix of the true support) are contained in the interval $[C_{\min}, C_{\max}]$.
- (A2) *Irrepresentable Condition:* There exists a fixed parameter $\gamma \in (0, 1]$ such that

$$\|\Sigma_{S^cS}(\Sigma_{SS})^{-1}\|_\infty \leq 1 - \gamma.$$

- (A3) *Self-incoherence:* There exists some $D_{\max} < +\infty$ such that

$$\|(\Sigma_{SS})^{-1}\|_\infty \leq D_{\max}.$$

- Intuition of bounded eigenspectrum condition: prevents excess dependence among elements of the design matrix associated with the support S.)
- Intuition of Irrepresentable condition: correlation between features in true support and noise features is not too high.

Sparsity Overlap function

- *Sparsity-overlap function* $\phi(\boldsymbol{B}^*)$ measures the sparsity of the matrix $\boldsymbol{B}^*$ as well as the overlap between the different regressions (the different columns of $\boldsymbol{B}^*$).
- In case of univariate regression $K = 1$, entries of design matrix are i.i.d. standard Gaussian; then $\phi(\boldsymbol{B}^*) = s$.
- For suitably correlated designs: sparsity overlap is $\phi(\boldsymbol{B}^*) = s/K$.
- Lemma 1(a): Under assumption (A1), the sparsity overlap $\phi(\boldsymbol{B}^*)$ obeys the bounds

$$\frac{s}{C_{\max} K} \leq \phi(\boldsymbol{B}^*) \leq \frac{s}{C_{\min}}.$$

Sparsity Overlap function

- *Sparsity-overlap function* $\phi(\boldsymbol{B}^*)$ measures the sparsity of the matrix $\boldsymbol{B}^*$ as well as the overlap between the different regressions (the different columns of $\boldsymbol{B}^*$).
- In case of univariate regression $K = 1$, entries of design matrix are i.i.d. standard Gaussian; then $\phi(\boldsymbol{B}^*) = s$.
- For suitably correlated designs: sparsity overlap is $\phi(\boldsymbol{B}^*) = s/K$.
- Lemma 1(a): Under assumption (A1), the sparsity overlap $\phi(\boldsymbol{B}^*)$ obeys the bounds

$$\frac{s}{C_{\max}K} \leq \phi(\boldsymbol{B}^*) \leq \frac{s}{C_{\min}}.$$

- We will see in Theorem 1 (next slide) that when the sparsity overlap is small, multivariate group lasso performs more favorably. Therefore we want C_{min} to be large (not too close to 0, hopefully close to 1).

Outline

Main Results

(under assumptions A1 - A3 and some other mild conditions)

- (Summary of Theorem 1.) The *sample complexity*

$$\theta(n, p, s) := \frac{n}{2\psi(\boldsymbol{B}^*) \log(p - s)},$$

  determines whether the multivariate group lasso recovers the exact row pattern (with high probability).
- (Summary of Corollary 1.) In particular, in the special case of the standard Gaussian ensemble, if $\theta(n, p, s)$ exceeds a critical level θ, the multivariate group lasso has a unique solution $\hat{\boldsymbol{B}}$ with row support $S(\boldsymbol{B})$ that is contained within the true row support $S(\boldsymbol{B}^*)$. Additionally, $\|\hat{\boldsymbol{B}} - \boldsymbol{B}^*\|_{\ell_\infty/\ell_2}$ is bounded. If $\theta(n, p, s)$ is below the critical level θ, the multivariate group lasso fails.
- More generally, the multivariate group lasso succeeds for problem sequences (n, p, s) such that $\theta(n, p, s)$ exceeds a critical level θ_u and fails for sequences such that $\theta(n, p, s)$ lies below a critical level θ_ℓ.

Main Results (continued)

(under assumptions A1 - A3 and some other mild conditions)

- Recall Lemma 1(a): under assumption (A1) we have bounds on $\phi(\boldsymbol{B}^*)$, so $\theta(n, p, s)$ ranges between

$$\frac{nC_{\min}}{2s\log(p-s)} \leq \theta(n, p, s) \leq \frac{nC_{\max}K}{2s\log(p-s)}. \tag{8}$$

  This generalizes a previous result from Wanwright (2009): the lasso succeeds in performing exact support recovery if the ratio $n/[s\log(p-s)]$ exceeds a certain threshold.
- (Summary of Corollary 3 and part of Corollary 1.) If $\boldsymbol{X}$ is uncorrelated for the variables corresponding to the active rows of $\boldsymbol{B}^*$, ℓ_1/ℓ_2 group regularization never harms performance relative to an ordinary lasso approach.

Main Results (continued)
(under assumptions A1 - A3 and some other mild conditions)

- Recall Lemma 1(a): under assumption (A1) we have bounds on $\phi(\boldsymbol{B}^*)$, so $\theta(n, p, s)$ ranges between

$$\frac{nC_{\min}}{2s\log(p-s)} \leq \theta(n,p,s) \leq \frac{nC_{\max}K}{2s\log(p-s)}. \tag{8}$$

  This generalizes a previous result from Wanwright (2009): the lasso succeeds in performing exact support recovery if the ratio $n/[s\log(p-s)]$ exceeds a certain threshold.
- (Summary of Corollary 3 and part of Corollary 1.) If $\boldsymbol{X}$ is uncorrelated for the variables corresponding to the active rows of $\boldsymbol{B}^*$, ℓ_1/ℓ_2 group regularization never harms performance relative to an ordinary lasso approach.

- Per (8), ℓ_1/ℓ_2 regularization for multivariate regression can yield substantial improvements in sample complexity (up to a factor of K) when the coefficient vectors are suitably orthogonal.

Main Results (continued)

(under assumptions A1 - A3 and some other mild conditions)

(Summary of Corollary 2.) Given a method that returns the row supports S (with high probability) with $|S| \ll p$, recovering the individual supports S_k with high probability is easy using the following procedure:

1. Compute the (restricted) multivariate ordinary least squares estimate

$$\tilde{\boldsymbol{B}}_{\hat{S}} = \arg\min_{\boldsymbol{B}_{\hat{S}}} \left\{ \| \boldsymbol{Y} - \boldsymbol{X}_{\hat{S}} \boldsymbol{B}_{\hat{S}} \|_F \right\} \tag{9}$$

   where $\boldsymbol{B}_{\hat{S}}$ is the coefficient matrix $\boldsymbol{B}$ but only with rows in the estimated support union $\hat{S}$ and $\boldsymbol{X}_{\hat{S}}$ contains only the columns of $\boldsymbol{X}$ corresponding to those features.

2. Compute $T(\tilde{\boldsymbol{B}}_{\hat{S}})$ by setting every entry in $\tilde{\boldsymbol{B}}_{\hat{S}}$ with absolute value less than $2\sqrt{2\log(K|\hat{S}|)/(C_{\min} n}$ equal to 0. Then the support is the nonzero entries of $T(\tilde{\boldsymbol{B}}_{\hat{S}})$.

Outline

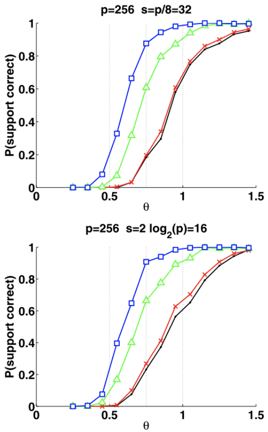


FIG. 3. *Plots of support recovery probability,* $\mathbb{P}[\widehat{S} = S]$, *versus the control parameter* $\theta = n/[2s\log(p-s)]$ *for two different type of sparsity: logarithmic sparsity on top* ($s = \mathcal{O}(\log(p))$) *and linear sparsity on bottom* ($s = \alpha p$), *and for increasing values of p from left to right. The noise level is set at* $\sigma = 0.1$. *Each graph shows four curves (black, red, green, blue) corresponding to the case of independent* ℓ_1 *regularization, and, for* ℓ_1/ℓ_2 *regularization, the cases of identical regression, intermediate angles and orthonormal regressions. Note how curves corresponding to the same case across different problem sizes p all coincide, as predicted by Theorems 1 and 2. Moreover, consistent with the theory, the curves for the identical regression group reach* $\mathbb{P}[\widehat{S} = S] \approx 0.50$ *at* $\theta \approx 1$, *whereas the orthonormal regression group reaches 50% success substantially earlier.*

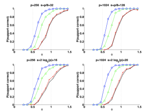


FIG. 3. *Plots of support recovery probability, $\mathbb{P}[\widehat{S} = S]$, versus the control parameter $\theta = n/[2s\log(p-s)]$ for two different type of sparsity: logarithmic sparsity on top ($s = \mathcal{O}(\log(p))$) and linear sparsity on bottom ($s = \alpha p$), and for increasing values of p from left to right. The noise level is set at $\sigma = 0.1$. Each graph shows four curves (black, red, green, blue) corresponding to the case of independent ℓ_1 regularization, and, for ℓ_1/ℓ_2 regularization, the cases of identical regression, intermediate angles and orthonormal regressions. Note how curves corresponding to the same case across different problem sizes p all coincide, as predicted by Theorems 1 and 2. Moreover, consistent with the theory, the curves for the identical regression group reach $\mathbb{P}[\widehat{S} = S] \approx 0.50$ at $\theta \approx 1$, whereas the orthonormal regression group reaches 50% success substantially earlier.*

- This illustrates threshold effect predicted by Theorems 1 and 2. Case of 2 responses ($K = 2$) which share an identical support set S with cardinality s (so $\boldsymbol{B}^*$ is an $s \times 2$ matrix).
- Fixed regularization parameter

$$\lambda_n = \sqrt{\frac{\log(p-s)\log(s)}{n}}.$$

  Move across independent axis by varying θ (p fixed, s determined by ratio or logarithm)
- In all cases here, design matrix is sampled from standard Gaussian ensemble. What we see here is that the probability of support recovery increases with the control parameter

$$\theta = \frac{n}{2s\log(p-s)}.$$

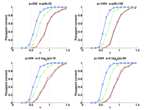


FIG. 3. *Plots of support recovery probability, $\mathbb{P}[\hat{S} = S]$, versus the control parameter $\theta = n/[2s\log(p-s)]$ for two different type of sparsity: logarithmic sparsity on top ($s = \mathcal{O}(\log(p))$) and linear sparsity on bottom ($s = \alpha p$), and for increasing values of p from left to right. The noise level is set at $\sigma = 0.1$. Each graph shows four curves (black, red, green, blue) corresponding to the case of independent ℓ_1 regularization, and, for ℓ_1/ℓ_2 regularization, the cases of identical regression, intermediate angles and orthonormal regressions. Note how curves corresponding to the same case across different problem sizes p all coincide, as predicted by Theorems 1 and 2. Moreover, consistent with the theory, the curves for the identical regression group reach $\mathbb{P}[\hat{S} = S] \approx 0.50$ at $\theta \approx 1$, whereas the orthonormal regression group reaches 50% success substantially earlier.*

- Black curve is independent ℓ_1 regularization (ordinary lasso), which we see does the worst in pretty much all settings. Red is ℓ_1/ℓ_2 regularization for identical regression (the columns of $\boldsymbol{B}^*$ are identical). For orthonormal regressions (blue lines), the columns of $\boldsymbol{B}^*$ are orthonormal, which is the most favorable setting for this method. For intermediate angles (green lines), the columns of $\boldsymbol{B}^*$ are at a 60 degree angle.
- Note that the improvement of this method over lasso is largest when the regressions are orthonormal; in the other extreme when the columns are identical, it does about the same as lasso.
- Note that varying p doesn't really change anything if the control parameter θ is equal.
- In identical regression group, Pr support recovery reaches 0.5 at $\theta = 1$; improvement in other regimes.

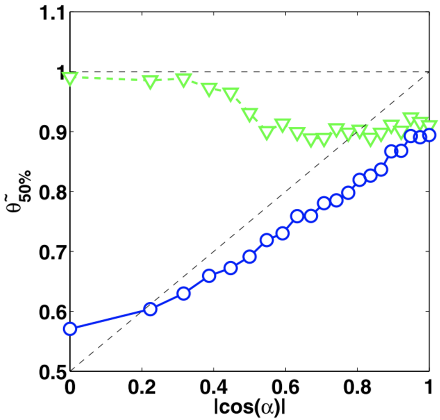


FIG. 5. *Plots of the Lasso sample complexity $\theta = n/[2s\log(p-s)]$ for which the probability of support union recovery exceeds 50% empirically as a function of $|\cos(\alpha)|$ for ℓ_1-based recovery and ℓ_1/ℓ_2-based recovery, where α is the angle between $Z^{(1)*}$ and $Z^{(2)*}$ for the family $\mathcal{B}_1$. We consider the two following methods for performing row selection: Ordinary Lasso (ℓ_1, green triangles) and multivariate group Lasso (blue circles).*

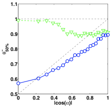


FIG. 5. *Plots of the Lasso sample complexity $\theta = n/[2s\log(p-s)]$ for which the probability of support union recovery exceeds 50% empirically as a function of $|\cos(\alpha)|$ for ℓ_1-based recovery and ℓ_1/ℓ_2-based recovery, where α is the angle between $Z^{(1)*}$ and $Z^{(2)*}$ for the family $\mathcal{B}_1$. We consider the two following methods for performing row selection: Ordinary Lasso (ℓ_1, green triangles) and multivariate group Lasso (blue circles).*

- We have fixed $p = 2048$ and sparsity $s = \log_2(p) = 22$.
- α is the angle between columns of $\boldsymbol{B}^*$, so as its cosine increases, the columns come closer together.
- We see that lasso needs a higher value of θ to get a 50% chance of support recovery, but the gap decreases as the columns of $\boldsymbol{B}^*$ come closer and closer together, until there is no gap when they are equal.
- Theory from this paper predicts that the circles and triangles should lie at or below dotted lines.
- Intuition of why this method does better when columns are orthonormal: in that case independent estimates of the support are more likely to include (by union) spurious covariates in the row support.

Outline

Singular Value Decomposition of the coefficient matrix $\boldsymbol{B}^*$

- Reduced rank regression model: $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{B}^* + \boldsymbol{W}$ as before. We will be concerned with $\text{rank}(\boldsymbol{B}^*) = r^* \leq \min\{p, K\}$.
- Singular value decomposition (SVD):

$$\boldsymbol{B}^* = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T = \sum_{k=1}^{r^*} d_k \boldsymbol{u}_k \boldsymbol{v}_k^T = \sum_{k=1}^{r^*} \boldsymbol{B}_k \tag{10}$$

  - $\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{r^*}) \in \mathbb{R}^{p \times r^*}$ consists of orthonormal left singular vectors
  - $\boldsymbol{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{r^*}) \in \mathbb{R}^{K \times r^*}$ consists of orthonormal right singular vectors
  - $\boldsymbol{D} \in \mathbb{R}^{r^* \times r^*}$ is a diagonal matrix with positive singular values $d_1 > \ldots, > d_{r^*}$ on its diagonal.

Singular Value Decomposition of the coefficient matrix $\boldsymbol{B}^*$

- Reduced rank regression model: $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{B}^* + \boldsymbol{W}$ as before. We will be concerned with $\operatorname{rank}(\boldsymbol{B}^*) = r^* \leq \min\{p, K\}$.
- Singular value decomposition (SVD):

$$\boldsymbol{B}^* = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T = \sum_{k=1}^{r^*} d_k \boldsymbol{u}_k \boldsymbol{v}_k^T = \sum_{k=1}^{r^*} \boldsymbol{B}_k \tag{10}$$

  - $\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{r^*}) \in \mathbb{R}^{p \times r^*}$ consists of orthonormal left singular vectors
  - $\boldsymbol{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{r^*}) \in \mathbb{R}^{K \times r^*}$ consists of orthonormal right singular vectors
  - $\boldsymbol{D} \in \mathbb{R}^{r^* \times r^*}$ is a diagonal matrix with positive singular values $d_1 > \ldots, > d_{r^*}$ on its diagonal.

- Using singular value decomposition, $\boldsymbol{B}^*$ can be expressed as a sum of r^* unit rank matrices $\boldsymbol{B}_k$ that are proportional to the outer product of the left and right singular vectors.
- (The singular values are assumed to be distinct, so the decomposition is unique up to the signs of the singular vectors. In practice this is usually the case.)

Singular Value Decomposition of B^* (continued)

- **Key insight:** $\boldsymbol{B}^*$ is the sum of r^* orthogonal layers of decreasing importance. For each layer k, $\boldsymbol{u}_k$ are the predictor effects, $\boldsymbol{v}_k$ are the response effects, and d_k indicates the relative importance of the association.
- If we believe that each pathway of association involves only a subset of the responses and predictors, then the left and right singular vectors should be sparse.

Outline

Optimization Problem

We plug the expression for $\boldsymbol{B}^*$ from (10) into the typical penalized least squares objective function (similar to (6)) to obtain the objective function to optimize:

$$\underset{d_k,\boldsymbol{u}_k,\boldsymbol{v}_k,k\in 1,\ldots,r^*}{\arg\min} \left\{ \frac{1}{2} \left\| \boldsymbol{Y} - \boldsymbol{X} \sum_{k=1}^{r^*} d_k \boldsymbol{u}_k, \boldsymbol{v}_k^T \right\|_F^2 + \sum_{k=1}^{r^*} p_\lambda\big(\lambda_k, (d_k, \boldsymbol{u}_k, \boldsymbol{v}_k)\big) \right\} \tag{11}$$

where p_λ is a penalty function (to be specified), $\|\boldsymbol{u}_k\|_2 = 1$ and $\|\boldsymbol{v}_k\|_2 = 1$.

- The penalty function used to encourage sparsity is

$$p_\lambda\big(\lambda_k, (d_k, \boldsymbol{u}_k, \boldsymbol{v}_k)\big) = \lambda_k \sum_{i=1}^{p} \sum_{j=1}^{K} w_{ijk} |d_k u_{ik} v_{jk}|$$

$$= \lambda_k (w_k^{(d)} d_k) \left(\sum_{i=1}^{p} w_{ik}^{(u)} |u_{ik}| \right) \left(\sum_{j=1}^{K} w_{jk}^{(v)} |v_{jk}| \right) \qquad (12)$$

- Penalizes each singular vector in the SVD layer, creating automatic adjustment of sparsity between $\boldsymbol{u}_k$ and $\boldsymbol{v}_k$.
- w_{ijk} is a weighting term (similar to the adaptive lasso (Zou 2006))

Optimization Problem

Penalty Function

- The penalty function used to encourage sparsity is

$$\begin{aligned} p_\lambda(\lambda_k, (d_k, \boldsymbol{u}_k, \boldsymbol{v}_k)) &= \lambda_k \sum_{i=1}^{p} \sum_{j=1}^{K} w_{ijk} |d_k u_{ik} v_{jk}| \\ &= \lambda_k (w_k^{(d)} d_k) \left(\sum_{i=1}^{p} w_{ik}^{(u)} |u_{ik}| \right) \left(\sum_{j=1}^{K} w_{jk}^{(v)} |v_{jk}| \right) \end{aligned} \tag{12}$$

- Penalizes each singular vector in the SVD layer, creating automatic adjustment of sparsity between $\boldsymbol{u}_k$ and $\boldsymbol{v}_k$.
- w_{ijk} is a weighting term (similar to the adaptive lasso (Zou 2006))

- Weight term allows for flexibility in how much we penalize individual terms in each SVD layer.

Adaptive Lasso (review)

- Recall from Zou (2006): adaptive lasso estimates $\hat{\boldsymbol{\beta}}^{*(n)}$ are given by

$$\hat{\boldsymbol{\beta}}^{*(n)} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \left\{ \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda_n \sum_{j=1}^{p} \hat{w}_j |\beta_j| \right\} \tag{13}$$

  where $\hat{\boldsymbol{w}} = (|\hat{\beta}_1|^{-2}, \ldots, |\hat{\beta}_p|^{-2})$ where $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)$ is the OLS estimate for β.
- Zou shows the adaptive lasso is consistent in variable selection and that the asymptotic distribution of $\hat{\boldsymbol{\beta}}^{*(n)}$ is normal, unbiased, and efficient.

Adaptive Lasso (review)

- Recall from Zou (2006): adaptive lasso estimates $\hat{\boldsymbol{\beta}}^{*(n)}$ are given by

$$\hat{\boldsymbol{\beta}}^{*(n)} = \underset{\beta \in \mathbb{R}^p}{\arg\min} \left\{ \|\boldsymbol{y} - \boldsymbol{X}\beta\|_2^2 + \lambda_n \sum_{j=1}^{p} \hat{w}_j |\beta_j| \right\} \tag{13}$$

where $\hat{\boldsymbol{w}} = (|\hat{\beta}_1|^{-2}, \ldots, |\hat{\beta}_p|^{-2})$ where $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \ldots, \hat{\beta}_p)$ is the OLS estimate for β.

- Zou shows the adaptive lasso is consistent in variable selection and that the asymptotic distribution of $\hat{\boldsymbol{\beta}}^{*(n)}$ is normal, unbiased, and efficient.

- in the (univariate) linear model $\boldsymbol{y} = \boldsymbol{X}\beta + \epsilon$
- (We will see that CCS prove similar results for their estimator.)

- In the rank one case ($r^* = 1$, so we don't index over k), Chen, Chan and Stenseth (CCS) choose the penalty weightings $w^{(d)}, \boldsymbol{w}^{(u)}, \boldsymbol{w}^{(v)}$ in (12) as follows:
  1. Estimate $\boldsymbol{B}^*$ (call the estimate $\tilde{\boldsymbol{B}}$)
  2. Find the SVD of $\tilde{\boldsymbol{B}}$: $\tilde{\boldsymbol{B}} = \tilde{d}\tilde{\boldsymbol{u}}\tilde{\boldsymbol{v}}^T$
  3. Set

$$\begin{cases} w^{(d)} = |\tilde{d}|^{-2} \\ \boldsymbol{w}^{(u)} = (|\tilde{u}_1|^{-2}, \ldots, |\tilde{u}_p|^{-2}) \\ \boldsymbol{w}^{(v)} = (|\tilde{v}_1|^{-2}, \ldots, |\tilde{v}_K|^{-2}) \end{cases} \tag{14}$$

- CCS provide a (more complicated) generalization of this rule for the general case ($r^* \in \mathbb{N}$).

Optimization Problem

Penalty Function (continued)

- In the rank one case ($r^* = 1$, so we don't index over k), Chen, Chan and Stenseth (CCS) choose the penalty weightings $w^{(d)}, \boldsymbol{w}^{(u)}, \boldsymbol{w}^{(v)}$ in (12) as follows:
  1. Estimate $\boldsymbol{B}^*$ (call the estimate $\tilde{\boldsymbol{B}}$)
  2. Find the SVD of $\tilde{\boldsymbol{B}}$: $\tilde{\boldsymbol{B}} = \tilde{d}\tilde{\boldsymbol{u}}\tilde{\boldsymbol{v}}^T$
  3. Set

$$\begin{cases} w^{(d)} = |\tilde{d}|^{-2} \\ \boldsymbol{w}^{(u)} = (|\tilde{u}_1|^{-2}, \ldots, |\tilde{u}_p|^{-2}) \\ \boldsymbol{w}^{(v)} = (|\tilde{v}_1|^{-2}, \ldots, |\tilde{v}_K|^{-2}) \end{cases} \tag{14}$$

- CCS provide a (more complicated) generalization of this rule for the general case ($r^* \in \mathbb{N}$).

- Note the similarity to the adaptive lasso weightings.
- Other choices besides -2 for the exponent are possible; CCS choose -2 based on simulation studies as well as the suggestion of Zou (2006).

Exclusive Extraction Algorithm (EEA) for sparse SVD estimation of B^*

Basic idea: Start with an initial estimator $\tilde{\boldsymbol{B}} = \sum_{k=1}^{r^*} \tilde{d}_k \tilde{\boldsymbol{u}}_k \tilde{\boldsymbol{v}}_k^T$ for $\boldsymbol{B}^*$, then compute r^* parallel sparse unit rank regression problems.

(a) For each $k \in \{1, \dots, r^*\}$:
   (i) Construct the adaptive weights $w_k^{(d)} = |\tilde{d}_k|^{-2}$, $\boldsymbol{w}_k^{(u)} = (|\tilde{u}_{1k}|^{-2}, \dots, |\tilde{u}_{pk}|^{-2})$, and $\boldsymbol{w}_k^{(v)} = (|\tilde{v}_{1k}|^{-2}, \dots, |\tilde{v}_{Kk}|^{-2})$.
   (ii) Let $\tilde{\boldsymbol{B}}_k = \tilde{d}_k \tilde{\boldsymbol{u}}_k \tilde{\boldsymbol{v}}_k^T$, and construct the exclusive layer $\boldsymbol{Y}_k = \boldsymbol{Y} - \boldsymbol{X}(\tilde{\boldsymbol{B}} - \tilde{\boldsymbol{B}}_k)$.
   (iii) Find $(\hat{d}_k, \hat{\boldsymbol{u}}_k, \hat{\boldsymbol{v}}_k)$ by performing the sparse unit rank regression of $\boldsymbol{Y}_k$ on $\boldsymbol{X}$.

(b) The final estimator of $\boldsymbol{B}^*$ is given by $\hat{\boldsymbol{B}} = \sum_{k=1}^{r^*} \hat{d}_k \hat{\boldsymbol{u}}_k \hat{\boldsymbol{v}}_k^T$.

Exclusive Extraction Algorithm (EEA) for sparse SVD estimation of $\boldsymbol{B}^*$

Basic idea: Start with an initial estimator $\tilde{\boldsymbol{B}} = \sum_{k=1}^{r^*} \tilde{d}_k \tilde{\boldsymbol{u}}_k \tilde{\boldsymbol{v}}_k^T$ for $\boldsymbol{B}^*$, then compute r^* parallel sparse unit rank regression problems.

1. For each $k \in \{1, \dots, r^*\}$:
   1. Construct the adaptive weights $w_k^{(d)} = |\tilde{d}_k|^{-2}, \boldsymbol{w}_k^{(u)} = (|\tilde{u}_{1k}|^{-2}, \dots, |\tilde{u}_{pk}|^{-2})$, and $\boldsymbol{w}_k^{(v)} = (|\tilde{v}_{1k}|^{-2}, \dots, |\tilde{v}_{Kk}|^{-2})$.
   2. Let $\tilde{\boldsymbol{B}}_k = \tilde{d}_k \tilde{\boldsymbol{u}}_k \tilde{\boldsymbol{v}}_k^T$, and construct the exclusive layer $\boldsymbol{Y}_k = \boldsymbol{Y} - \boldsymbol{X}(\tilde{\boldsymbol{B}} - \tilde{\boldsymbol{B}}_k)$.
   3. Find $(\hat{d}_k, \hat{\boldsymbol{u}}_k, \hat{\boldsymbol{v}}_k)$ by performing the sparse unit rank regression of $\boldsymbol{Y}_k$ on $\boldsymbol{X}$.
2. The final estimator of $\boldsymbol{B}^*$ is given by $\hat{\boldsymbol{B}} = \sum_{k=1}^{r^*} \hat{d}_k \hat{\boldsymbol{u}}_k \hat{\boldsymbol{v}}_k^T$.

- Note that this requires you to know r^*; CCS show that this method has some robustness if r^* is misspecified.
- Typically the initial estimator is the reduced rank least squares estimator (ordinary regression but allowing for possibility that coefficient matrix is rank-deficient (e.g. two columns are identical)
- If instead of using $\boldsymbol{Y}_k = \boldsymbol{Y} - \boldsymbol{X}(\tilde{\boldsymbol{B}} - \tilde{\boldsymbol{B}}_k)$ you use $\boldsymbol{Y}_k = \boldsymbol{Y} - \boldsymbol{X}\tilde{\boldsymbol{B}}$ the you get the sequential extraction algorithm (SEA) method which will come up later in simulations.
- (The optimal λ_k is usually chosen by a BIC specified by CCS, because cross-validation becomes infeasible due to the number of parameters to optimize over. CCS did simulations and found that results were similar with five-fold CV versus BIC.)
- Computational cost is linear in r^*; estimation for different layers can be done in parallel

Iterative Exclusive Extraction Algorithm (IEEA) and Orthogonality Relaxation

- Iterative exclusive extraction algorithm (IEEA): perform EEA once, then uses this as the initial estimate for another EEA iteration, repeating until the difference between estimates is sufficiently small.
- Recall that $\boldsymbol{U}$ and $\boldsymbol{V}$ are typically orthonormal in SVD. Relax the orthogonality condition to obtain sparsity in SVD.
- The estimators of different layers are consistent (under Theorem 2), so the estimators are "asymptotically orthogonal," even though the solutions yielded are in general not exactly orthogonal.

- Iterative exclusive extraction algorithm (IEEA): perform EEA once, then uses this as the initial estimate for another EEA iteration, repeating until the difference between estimates is sufficiently small.
- Recall that $\boldsymbol{U}$ and $\boldsymbol{V}$ are typically orthonormal in SVD. Relax the orthogonality condition to obtain sparsity in SVD.
- The estimators of different layers are consistent (under Theorem 2), so the estimators are "asymptotically orthogonal," even though the solutions yielded are in general not exactly orthogonal.

- The optimization (11) is carried out locally near an initial consistent estimator of $\boldsymbol{B}^*$ without enforcing exact orthogonality.
- The relaxation of the orthogonality requirement also improves local search efficiency.
- Usually only need a small number of additional iterations (less than 10) to get convergence (according to simulations even 1 or 2 gets you very close to optimal solution)

- The optimization problem (11) is nonconvex, but CCS developed an efficient parallelized coordinate descent optimization algorithm.
- Typically convergence occurs within only a few iterations.

Outline

Theorems

- (Summary of Theorem 1) Under very mild assumptions, the optimization problem (11) solved using the IEEA has a local minimum with $\sqrt{n}$ convergence.
- (Summary of Theorem 2) Under very mild assumptions, the asymptotic distributions of the IEEA estimators of the nonzero elements of $\boldsymbol{U}$ and $\boldsymbol{V}$ are normal with zero mean. Further, the estimators are consistent.
- (Summary of Theorem 3) Under very mild assumptions, the IEEA estimators are selection consistent; that is, the elements of $\boldsymbol{U}$ and $\boldsymbol{V}$ that equal 0 will be set to 0 by IEEA with probability 1 as the number of iterations tends to infinity.

Theorems

- (Summary of Theorem 1) Under very mild assumptions, the optimization problem (11) solved using the IEEA has a local minimum with $\sqrt{n}$ convergence.
- (Summary of Theorem 2) Under very mild assumptions, the asymptotic distributions of the IEEA estimators of the nonzero elements of $\boldsymbol{U}$ and $\boldsymbol{V}$ are normal with zero mean. Further, the estimators are consistent.
- (Summary of Theorem 3) Under very mild assumptions, the IEEA estimators are selection consistent; that is, the elements of $\boldsymbol{U}$ and $\boldsymbol{V}$ that equal 0 will be set to 0 by IEEA with probability 1 as the number of iterations tends to infinity.

- Note the similarity to the adaptive lasso results discussed earlier

Outline

Table 3. Estimation and prediction accuracy of the IEEA, SEA, OLS, RRR and NNP estimators

Model	*SNR*	*Error*	*Results for the following methods:*				
			IEEA	*SEA*	*OLS*	*RRR*	*NNP*
I	0.125	Er-Est	3.29	4.28	55.32	9.90	10.99
		Er-Pred	52.10	67.37	416.34	106.40	139.10
	0.25	Er-Est	1.10	1.52	27.21	4.56	6.65
		Er-Pred	17.68	23.20	197.12	46.16	74.55
	0.5	Er-Est	0.57	0.99	15.12	2.44	3.75
		Er-Pred	9.48	15.03	110.45	25.89	43.74
	1	Er-Est	0.23	0.52	7.41	1.17	2.19
		Er-Pred	4.01	7.59	52.95	11.87	23.23
II	0.125	Er-Est	0.52	0.51	51.50	4.87	5.01
		Er-Pred	12.86	14.84	342.59	60.12	72.00
	0.25	Er-Est	0.15	0.29	28.09	3.89	4.22
		Er-Pred	4.47	7.85	176.22	23.64	45.44
	0.5	Er-Est	0.06	0.17	14.81	3.23	3.67
		Er-Pred	1.80	4.27	84.94	9.67	26.87
	1	Er-Est	0.03	0.10	8.40	2.75	3.13
		Er-Pred	0.84	2.35	42.57	4.55	15.43

Table 3. Estimation and prediction accuracy of the IEEA, SEA, OLS, RRR and NNP estimators

Model	*SNR*	*Error*	*Results for the following methods:*				
			IEEA	*SEA*	*OLS*	*RRR*	*NNP*
I	0.125	Er-Est	3.29	4.28	55.32	9.90	10.99
		Er-Pred	52.10	67.37	416.34	106.40	139.10
	0.25	Er-Est	1.10	1.52	27.21	4.56	6.65
		Er-Pred	17.68	23.20	197.12	46.16	74.55
	0.5	Er-Est	0.57	0.99	15.12	2.44	3.75
		Er-Pred	9.48	15.03	110.45	25.89	43.74
	1	Er-Est	0.23	0.52	7.41	1.17	2.19
		Er-Pred	4.01	7.59	52.95	11.87	23.23
II	0.125	Er-Est	0.52	0.51	51.50	4.87	5.01
		Er-Pred	12.86	14.84	342.59	60.12	72.00
	0.25	Er-Est	0.15	0.29	28.09	3.89	4.22
		Er-Pred	4.47	7.85	176.22	23.64	45.44
	0.5	Er-Est	0.06	0.17	14.81	3.23	3.67
		Er-Pred	1.80	4.27	84.94	9.67	26.87
	1	Er-Est	0.03	0.10	8.40	2.75	3.13
		Er-Pred	0.84	2.35	42.57	4.55	15.43

- IEEA: proposed method; regularization parameter chosen using BIC.
- SEA: simplification of EEA. In SEA: sequentially perform sparse unit rank regression, each time with data matrix Y replaced by residual matrix Y - X B. Has been used in many penalized matrix decomposition problems. Correspond to a different decomposition of the coefficient matrix other than SVD, and need not produce SVD layers of B, so it is not suitable for recovering the desired sparse SVD structure in B. Regularization parameter chosen using BIC.
- OLS: least squares
- RRR: reduced rank regression (ordinary regression but allowing for possibility that coefficient matrix is rank-deficient (e.g. two columns are identical)
- NNP: nuclear-norm penalized regression (nuclear norm is sum of absolute values of eigenvalues)

Table 3. Estimation and prediction accuracy of the IEEA, SEA, OLS, RRR and NNP estimators

Model	*SNR*	*Error*	*Results for the following methods:*				
			IEEA	*SEA*	*OLS*	*RRR*	*NNP*
I	0.125	Er-Est	3.29	4.28	55.32	9.90	10.99
		Er-Pred	52.10	67.37	416.34	106.40	139.10
	0.25	Er-Est	1.10	1.52	27.21	4.56	6.65
		Er-Pred	17.68	23.20	197.12	46.16	74.55
	0.5	Er-Est	0.57	0.99	15.12	2.44	3.75
		Er-Pred	9.48	15.03	110.45	25.89	43.74
	1	Er-Est	0.23	0.52	7.41	1.17	2.19
		Er-Pred	4.01	7.59	52.95	11.87	23.23
II	0.125	Er-Est	0.52	0.51	51.50	4.87	5.01
		Er-Pred	12.86	14.84	342.59	60.12	72.00
	0.25	Er-Est	0.15	0.29	28.09	3.89	4.22
		Er-Pred	4.47	7.85	176.22	23.64	45.44
	0.5	Er-Est	0.06	0.17	14.81	3.23	3.67
		Er-Pred	1.80	4.27	84.94	9.67	26.87
	1	Er-Est	0.03	0.10	8.40	2.75	3.13
		Er-Pred	0.84	2.35	42.57	4.55	15.43

- Model I: $p = K = 25, n = 50, r^* = 3$.
- Model II: $p = K = 60, n = 50, r^* = 3$ (same but more noise features, and more responses even though their columns of B don't add to rank of matrix).
- Then generate data via $Y = XB + W$ with σ chosen according to desired SNR.
- Vary SNR as well as estimation and prediction error. Er-Est: $\|B^* - \hat{B}\|_F^2/(pq)$. Er-Pred: $\|XB^* - X\hat{B}\|_F^2/(pq)$.
- We see that authors are correct that IEEA pretty much uniformly outperforms SEA, althoguh this is the second best method. (Sparsity assumption benefits method when model dimension is high and number of irrelevant responses or predictors is large.)
- Next RRR, then NNP, finally OLS.