Outline

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- Subset Selection
- Shrinkage Method
- Beyond Lasso

Part 1: Introduction to Dimension Reduction

- **Introduction to Dimension Reduction**
  - **Difference between feature selection and feature extraction**
  - **Feature Selection**
    - **Wrapper method**
    - **Filter method**
    - **Embedded method**
  - **Feature Extraction**
    - **PCA, ICA...**
- Linear Regression and Least Squares (Review)
- Subset Selection
- Shrinkage Method
- Beyond Lasso

Difference between Feature Selection and Feature Extraction

Feature Selection

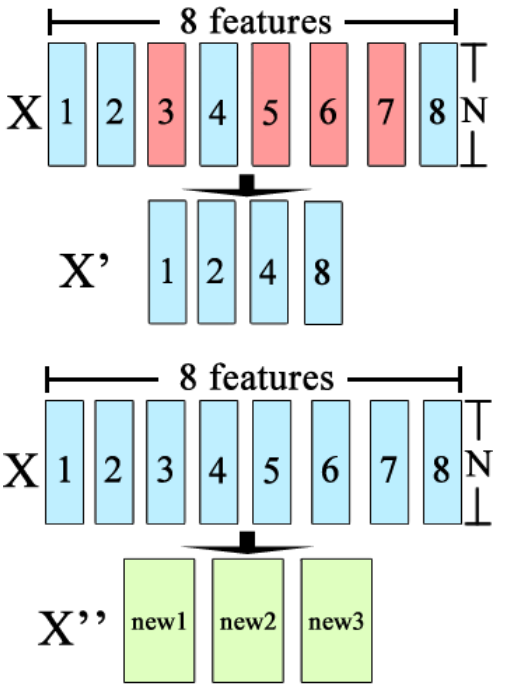
- chooses a subset of features from the original feature set

Feature Extraction

- transforms the original features into new ones
- e.g. projects data from high dimensions to low dimensions

Question:

- Why do we need two frameworks for dimension reduction?



Difference between Feature Selection and Feature Extraction

Example 1: Prostate Cancer

The data come from a study by Stamey et al.(1989). In this task, we are trying to identify a subset of features that are useful for prediction of the level of prostate-specific antigen (lpsa). Our available feature set is

$$\{lcavol, lweight, age, lbph, svi, lcp, gleason, pgg45\}.$$

In this case, we would like to have our result as a subset of the whole set, such as

$$\{lcavol, lweight, age, svi, lcp\},$$

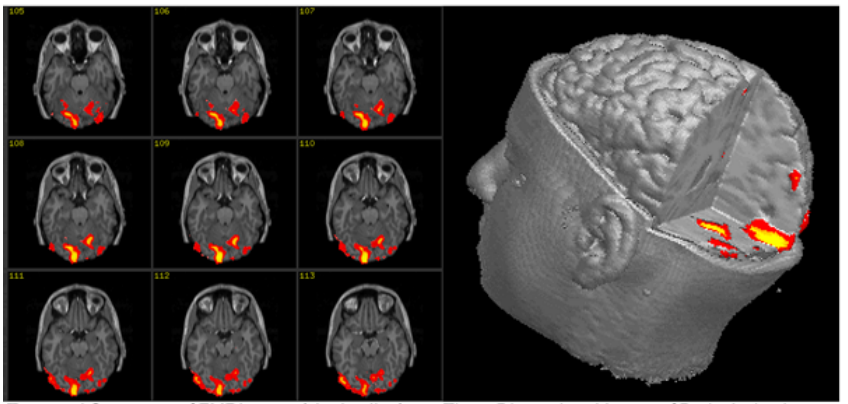
which are important for prediction of lpsa.

Feature selection applies.

Difference between Feature Selection and Feature Extraction

Example 2: classification with fMRI data

fMRI data are 4D images, with one dimension being the time slot.



Temporal Sequence of FMRI scans (single slice)
from
http://www.fmrib.ox.ac.uk/fmri_intro/brief.html

Three Dimensional Image of Brain Activation
from
http://www.fmrib.ox.ac.uk/fmri_intro/brief.html

Difference between Feature Selection and Feature Extraction

Example 2: classification with fMRI data

- fMRI data are 4D images, with one dimension being the time slot.
- Suppose the dimension of images is $50 \times 50 \times 50$ for single time point and we have 200 time points
- $50 \times 50 \times 50 \times 200 = 25,000,000$ dimensions in total! This will cause great computation burdun

In this task, we are not concerned about importance of particular voxels. Our purpose is to decrease the number of dimensions without losing too much information for further prediction task.

Feature extraction applies better.

Feature Selection

Wrapper Methods

- search the space of feature subsets
- use the training/validation accuracy of a particular classifier as the measure of utility for a candidate subset

Embedded Methods

- exploit the structure of specific classes of learning models to guide the feature selection process
- e.g. LASSO. It is embedded as part of the model construction process

Filter Methods

- use some general rules/criterions to measure the feature selection results independent of the classifiers
- e.g. mutual information based method

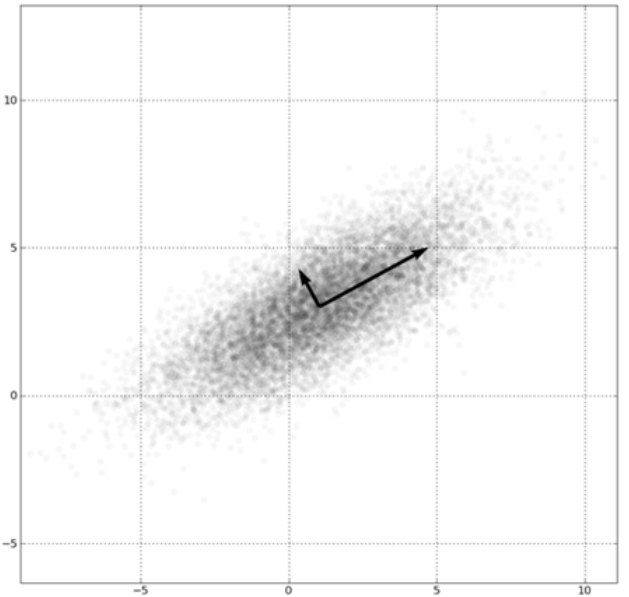
Feature Selection

Comparison

	WRAPPER	FILTER	EMBEDDED
Speed	Low	High	Mid
Chance of Overfitting	High	Low	Mid
Classifier-Independent	No	Yes	No

Feature Extraction

- **A graphical explanation**
  - Each data sample has two features
  - Prefer the direction with larger variance
  - Original features are transformed into new ones



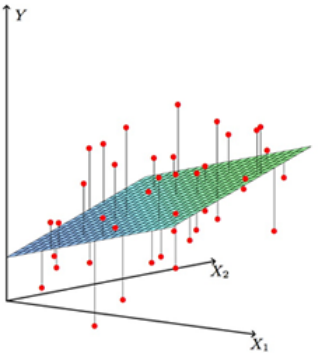
Part 2: Linear Regression and Least Squares (Review)

- Introduction to Dimension Reduction
- **Linear Regression and Least Squares (Review)**
  - **Least Square Fit**
  - **Gauss Markov**
  - **Bias-Variance tradeoff**
  - **Problems**
- Subset Selection
- Shrinkage Method
- Beyond Lasso

Linear Regression and Least Squares (Review)

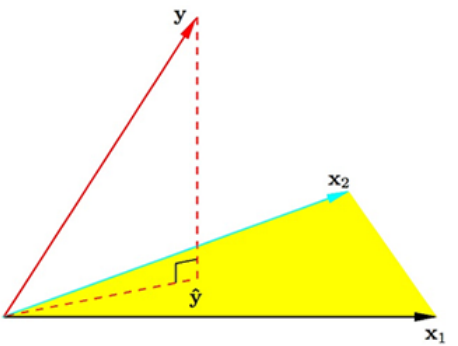
Least Squares Fit

$$RSS(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)$$
$$\frac{\partial RSS}{\partial \beta} = -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta)$$
$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$



Gauss Markov Theorem

The least squares estimates of the parameters β have the smallest variance among all linear unbiased estimates.



Question

Is unbiased assumption necessary?

Part 3: Subset Selection

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- **Subset Selection**
  - **Best-subset selection**
  - **Forward stepwise selection**
  - **Forward stagewise selection**
  - **Problems**
- Shrinkage Method
- Beyond Lasso

Linear Regression and Least Squares (Review)

Bias-Variance tradeoff

$$\begin{aligned} MSE(\tilde{\theta} &= E[(\tilde{\theta} - \theta)^2]) \\ &= Var(\tilde{\theta} + [E[\tilde{\theta}] - \theta]) \end{aligned}$$

where $\theta = \alpha^T \beta$. We can trade some bias for much less variance.

Problems of Least Squares

- **Prediction accuracy**: low bias, but high variance, overfitting noise and sensitive to outlier
- **Interpretation**: Sometimes, especially when faced with numerous features, we may want a "big picture" of the model
- $(\mathbf{X}^T\mathbf{X})$ may be **not invertible** and thus no closed form solution

Subset Selection

Best-subset selection

- Best subset regression finds for each $k \in \{0, 1, 2, \ldots, p\}$ the subset of size k that gives smallest residual sum of squares.
- An efficient algorithm, the leaps and bounds procedure (Furnival and Wilson, 1974), makes this feasible for p as large as 30 or 40.

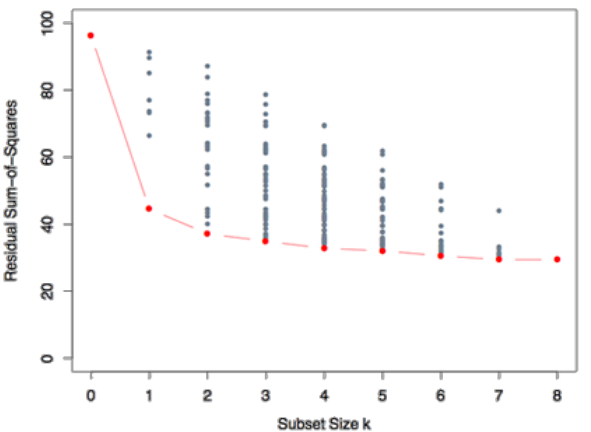


FIGURE 3.5. *All possible subset models for the prostate cancer example. At each subset size is shown the residual sum-of-squares for each model of that size.*

Subset Selection

Forward-stepwise selection

Instead of searching all possible subsets, we can seek a good path through them.

Forward-Stepwise Selection builds a model sequentially, adding one variable at a time. At each step, it

- identifies the best variable (with the highest correlation with the residual error) to include in the active set

$$\mathbf{x_k} = argmax_{\mathbf{x}_j}(|\mathbf{x}_j^T \mathbf{r}|)$$

- then updates the least squares fit to include all the active variables

Subset Selection

Forward-Stagewise Regression

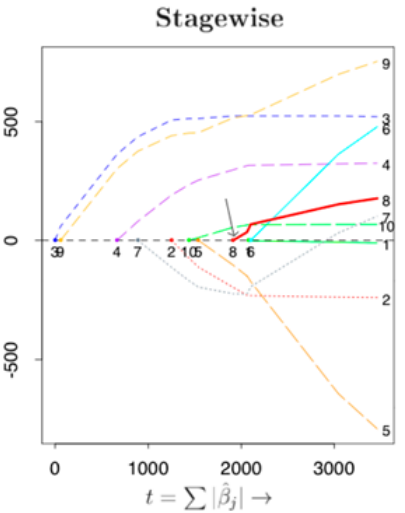
- Initialize the fit vector $\mathbf{f} = 0$
- Compute the correlation vector

$$\mathbf{c} = \mathbf{c}(\mathbf{f}) = \mathbf{X}^T(\mathbf{y} - \mathbf{f})$$

- $k = argmax_{j \in \{1,2,\ldots,p\}} |\mathbf{c}_j|$
- Coefficients and fit vector are updated

$$\mathbf{f} \leftarrow \mathbf{f} + \alpha \cdot sign(\mathbf{c}_j)\mathbf{x}_j$$

$$\beta_j \leftarrow \beta_j + \alpha \cdot sign(\mathbf{c}_j)$$



$m = 10$

$n = 442$

Subset Selection

Comparison

- It takes at most p steps for forward-stepwise selection to get the final fit
- Forward stagewise selection is a slow fitting algorithm, at each time step we only update one β_j, which can take more than p steps
- Forward stagewise is useful in high dimensional problem

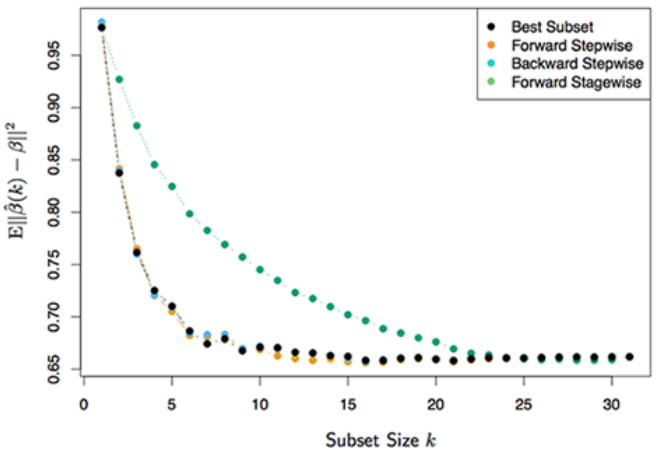


FIGURE 3.6. *Comparison of four subset-selection techniques on a simulated linear regression problem* $Y = X^T\beta + \varepsilon$. *There are* $N = 300$ *observations on* $p = 31$ *standard Gaussian variables, with pairwise correlations all equal to* 0.85. *For* 10 *of the variables, the coefficients are drawn at random from a* $N(0, 0.4)$ *distribution; the rest are zero. The noise* $\varepsilon \sim N(0, 6.25)$, *resulting in a signal-to-noise ratio of* 0.64. *Results are averaged over* 50 *simulations. Shown is the mean-squared error of the estimated coefficient* $\hat{\beta}(k)$ *at each step from the true* β.

Subset Selection

Pros

- More interpretable and compact model

Cons

- It is a discrete process, and thus has high variance and sensitivity to the change in dataset.
- Thus may not be able to lower prediction error

Ridge Regression

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- **Shrinkage Method**
  - **Ridge Regression**
    - **Formulations and closed form solution**
    - **Singular value decomposition**
    - **Degree of Freedom**
  - Lasso
- Beyond Lasso

Ridge Regression

- **Linear regression with l_2-regularization**
  - Least squares with quadratic constraints

$$\hat{\beta}^{ridge} = argmin_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \mathbf{x_{ij}} \beta_j)^2, s.t. \sum_{j=1}^{p} \beta_j^2 \leq t$$

  - Its dual form

$$\hat{\beta}^{ridge} = argmin_{\beta} \{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \mathbf{x_{ij}} \beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \}$$

  - The l_2-regularization can be viewed as a Gaussian prior on the coefficients, and our estimates are the posterior means

- **Solution**

$$RSS(\lambda) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta) + \lambda \beta^T \beta$$
$$\hat{\beta}^{ridge} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$$

Ridge Regression

Singular Value Decomposition (SVD)

- **The SVD of $\mathbf{X}$**:

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

  - $\mathbf{U}$: $N \times p$ **orthogonal** matrix with columns spanning the column space of $\mathbf{X}$
  - $\mathbf{V}$: $p \times p$ **orthogonal** matrix with columns spanning the row space of $\mathbf{X}$
  - $\mathbf{D}$: $p \times p$ **diagonal** matrix with diagonal entries $d_1 \geq d_2 \geq \ldots \geq d_p \geq 0$ being the singular values of $\mathbf{X}$

- **For least squares**

$$\begin{aligned} \mathbf{X}\hat{\beta}^{ls} &= \mathbf{X}(\mathbf{X}^{\mathbf{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathbf{T}}\mathbf{y} \\ &= \mathbf{U}\mathbf{U}^{\mathbf{T}}\mathbf{y} \end{aligned}$$

Ridge Regression

Singular Value Decomposition (SVD)

- **For ridge regression**

$$\begin{aligned}\mathbf{X}\hat{\beta}^{ridge} &= \mathbf{X}(\mathbf{X}^{\mathbf{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathbf{T}}\mathbf{y} \\ &= \sum_{j=1}^{p} \mathbf{u}_j \frac{d_j^2}{d_j^2 + \lambda} \mathbf{u}_j^T \mathbf{y}\end{aligned}$$

  - Compared with the solution of least square, we have an additional shrinkage term, the smaller d is and the larger λ is, the more shrinkage we have.

- The SVD of the centered matrix X is another way of expressing the principal components of the variables in X.

Ridge Regression

Singular Value Decomposition (SVD)

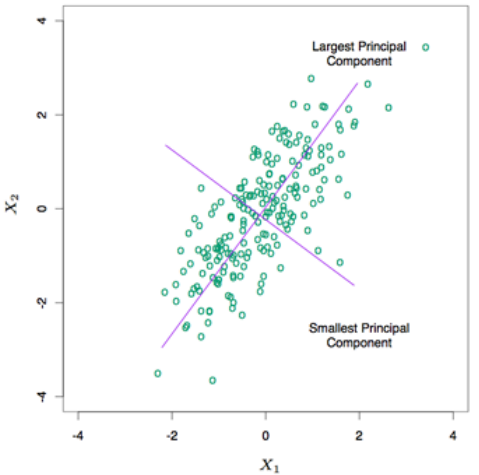


FIGURE 3.9. *Principal components of some input data points. The largest principal component is the direction that maximizes the variance of the projected data, and the smallest principal component minimizes that variance. Ridge regression projects* $\mathbf{y}$ *onto these components, and then shrinks the coefficients of the low–variance components more than the high-variance components.*

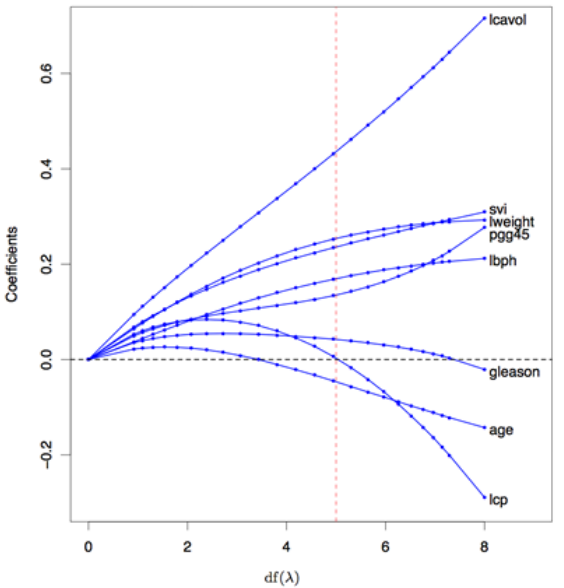
Ridge Regression

Degree of Freedom

- In statistics, the number of degrees of freedom is the number of values in the final calculation of a statistic that are free to vary.
- Computation

$$
\begin{aligned}
d(\lambda) &= tr[\mathbf{X}(\mathbf{X}^{\mathbf{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathbf{T}}] \\
&= tr[\mathbf{H}_{\lambda}] \\
&= \sum_{j=1}^{p} \frac{d_j^2}{d_j^2 + \lambda}
\end{aligned}
$$

- The smaller d is and the larger λ is, the less degree of freedom we have



LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- **Shrinkage Method**
  - Ridge Regression
  - **Lasso**
    - **Formulations**
    - **Comparisons with ridge regression and subset selection**
    - **Quadratic Programming**
    - **Least Angle Regression**
    - **Viewed as approximation for l_0-regularization**
- Beyond Lasso

LASSO

Linear regression with l_1-regularization

- **Problems with l_2-regularization**
  - Interpretability and compactness: Though coefficients are shrinked, but not to zero.
  - Least squares with constraints

$$\hat{\beta}^{ridge} = argmin_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \mathbf{x_{ij}} \beta_j)^2, s.t. \sum_{j=1}^{p} |\beta_j| \leq t$$

  - Its dual form

$$\hat{\beta}^{ridge} = argmin_{\beta} \{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \mathbf{x_{ij}} \beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \}$$

  - The 1_1-regularization can be viewed as a Laplace prior on the coefficients

LASSO

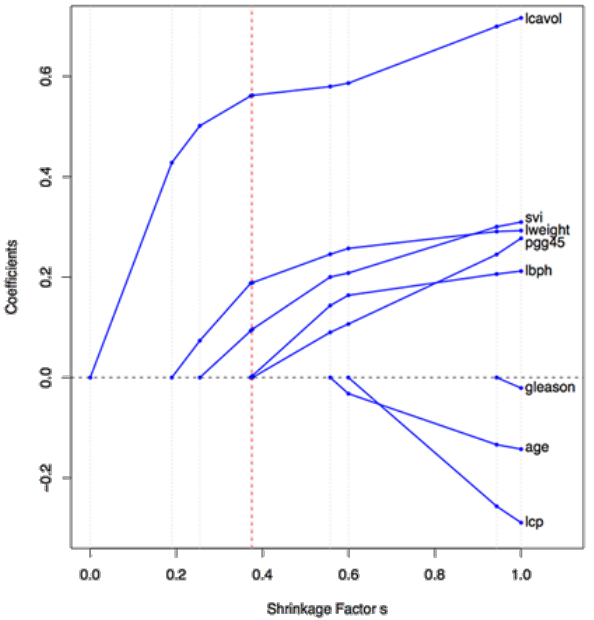


FIGURE 3.10. *Profiles of lasso coefficients, as the tuning parameter t is varied. Coefficients are plotted versus* $s = t/\sum_1^p |\hat{\beta}_j|$. *A vertical line is drawn at* $s = 0.36$, *the value chosen by cross-validation. Compare Figure 3.8 on page 65; the lasso profiles hit zero, while those for ridge do not. The profiles are piece-wise linear, and so are computed only at the points displayed; see Section 3.4.4 for details.*

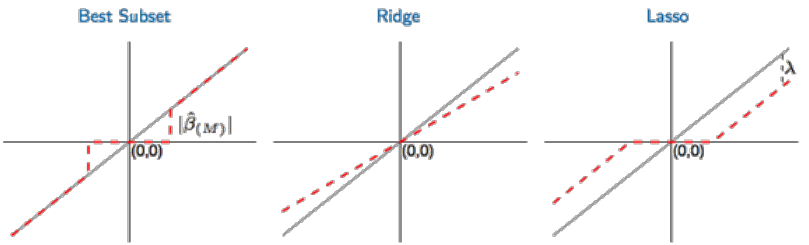
LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- **Shrinkage Method**
  - Ridge Regression
  - **Lasso**
    - Formulations
    - **Comparisons with ridge regression and subset selection**
      - **Orthonormal inputs**
      - **Non-orthonormal inputs**
    - Quadratic Programming
    - Least Angle Regression
    - Viewed as approximation for l_0-regularization
- Beyond Lasso

LASSO

Comparison

- **Orthonormal Input X**
  - **Best subset**: [Hard thresholding] Only keep the top M largest coefficeints of $\hat{\beta}^{ls}$
  - **Ridge**: [Pure shrinkage] does proportional shrinkage of $\hat{\beta}^{ls}$
  - **Lasso**: [Soft thresholding] translates each coefficient of $\hat{\beta}^{ls}$ by λ, truncating at 0



LASSO

Comparison

- Non-orthonormal Input X

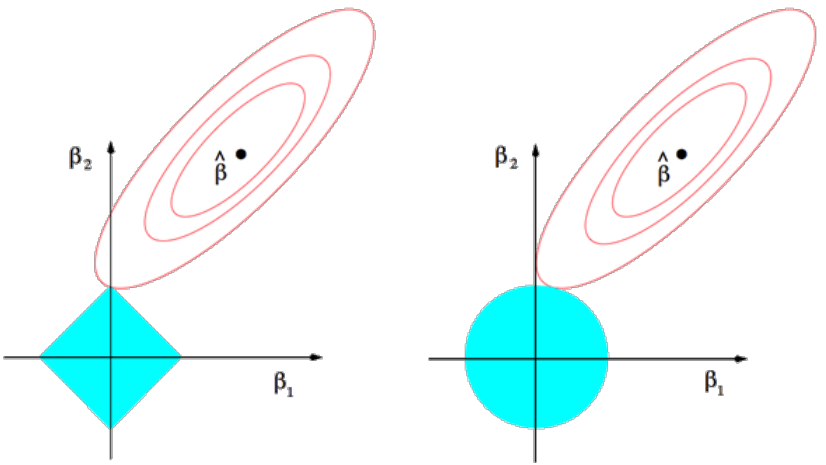


FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions* $|\beta_1| + |\beta_2| \leq t$ *and* $\beta_1^2 + \beta_2^2 \leq t^2$*, respectively, while the red ellipses are the contours of the least squares error function.*

LASSO

Other unit circles for different p-norms

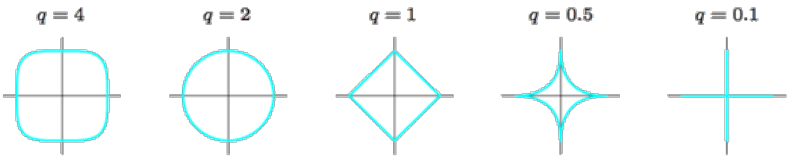


FIGURE 3.12. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for given values of* q.

	CONVEX	SMOOTH	SPARSE
$q < 1$	No	No	Yes
$q > 1$	Yes	Yes	No
$q = 1$	Yes	No	Yes

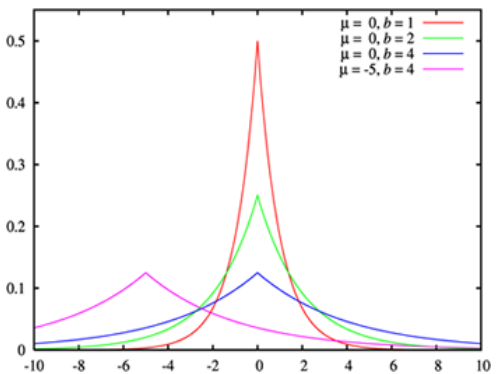
Here $q = 0$ is the pure variable selection procedure, as it is counting the **number of non-zero coefficients**.

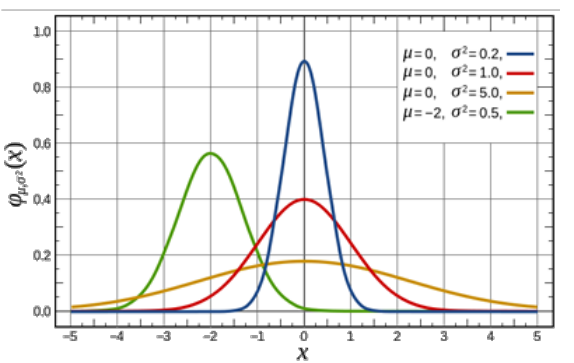
LASSO

Regularizations as priors

$|\beta_j|^q$ can be viewed as the log-prior density for β_j, these three methods are bayes estimates with different priors

- **Subset selection**: corresponds to $q = 0$
- **LASSO**: corresponds to $q = 1$, Laplace prior, $density = (\frac{1}{\tau})exp(\frac{-|\beta|}{\tau}), \tau = 1/\lambda$
- **Ridge regression**: corresponds to $q = 2$, Gaussian Prior





LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- **Shrinkage Method**
  - Ridge Regression
  - **Lasso**
    - Formulations
    - Comparisons with ridge regression and subset selection
    - **Quadratic Programming**
    - Least Angle Regression
    - Viewed as approximation for l_0-regularization
- Beyond Lasso

LASSO

Quadratic Programming

- Formulation

$$min_{\beta}\{\frac{1}{2}(\mathbf{X}\beta-\mathbf{y})^{T}(\mathbf{X}\beta-\mathbf{y})+\lambda\|\beta\|_{1}\}$$

is equivalent to

$$min_{w,\xi}\{\frac{1}{2}(\mathbf{X}\beta-\mathbf{y})^{T}(\mathbf{X}\beta-\mathbf{y})+\lambda\mathbf{1}^{T}\xi\}$$

$$s.t.\ \beta_j \le \xi_j$$
$$\beta_j \ge -\xi_j$$

- Note that QP can only solve LASSO with a fixed λ

LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- **Shrinkage Method**
  - Ridge Regression
  - **Lasso**
    - Formulations
    - Comparisons with ridge regression and subset selection
    - Quadratic Programming
    - **Least Angle Regression**
    - Viewed as approximation for l_0-regularization
- Beyond Lasso

Least Angle Regression

Notations

- $\mathcal{A}_k$: active set, the set of features we already included in the model at time step k
- $\beta_{\mathcal{A}_k}$: coefficients vector at the beginning of time step k
- $\beta_{\mathcal{A}_k}(\alpha)$: coefficients vector in time step k w.r.t. α,
- $\mathbf{f}_k$: the fit vector at the beginning of time step k, $\mathbf{f}_0 = 0$
- $\mathbf{f}_k(\alpha)$: the fit vector in time step k w.r.t. α
- $\mathbf{r}_k$: residual vector at the beginning of time step k, $\mathbf{r}_0 = \mathbf{y} - \bar{\mathbf{y}}$
- $\mathbf{r}_k(\alpha)$: residual vector in time step k, w.r.t. α

LAR Algorithm

- Initialization:
  - Standardized all predictors s.t. $\bar{\mathbf{x}}_j = 0, \mathbf{x}_j^T \mathbf{x}_j = 1$; $\mathbf{r}_0 = \mathbf{y} - \bar{\mathbf{y}}$; $\beta = \mathbf{0}$; $\mathbf{f}_0 = \mathbf{0}$; $\mathcal{A}_k = \varnothing$
  - $\mathbf{x}_k = argmax_{\mathbf{x}_j} |\mathbf{x}_j^T \mathbf{r}_0|$, $\mathcal{A}_1 = \{\mathbf{x}_k\}$
- Main
  - While termination_cond != true
    - $\mathbf{r}_k = \mathbf{y} - \mathbf{X}_{\mathcal{A}_k} \beta_{\mathcal{A}_k}$, $\mathbf{f}_k = \mathbf{X}_{\mathcal{A}_k} \beta_{\mathcal{A}_k}$
    - Search α
      - $\beta_{\mathcal{A}_k}(\alpha) = \mathcal{A}_k + \alpha \cdot \delta_k$, where $\delta_k = (\mathbf{X}_{\mathcal{A}_k}^{\mathbf{T}} \mathbf{X}_{\mathcal{A}_k})^{-1} \mathbf{X}_{\mathcal{A}_k}^{\mathbf{T}} \mathbf{r_k}$
      - Concurrently, $\mathbf{f}_k(\alpha) = \mathbf{f}_k + \alpha \cdot \mathbf{u}_k$, where $\mathbf{u}_k = \mathbf{X}_{\mathcal{A}_k} \delta_k$
    - Until $|\mathbf{X}_{\mathcal{A}_k} \mathbf{r}_k(\alpha)| = max_{\mathbf{x}_j \in \bar{\mathcal{A}}_k} |\mathbf{x}_j^T \mathbf{r}_k(\alpha)|$
    - $\mathbf{x}_k = argmax_{\mathbf{x}_j \in \bar{\mathcal{A}}_k} |\mathbf{x}_j \mathbf{r}_k(\alpha)|$
    - $\mathcal{A}_{k+1} = \mathcal{A}_k \cup \{\mathbf{x}_k\}$

LAR

Comments

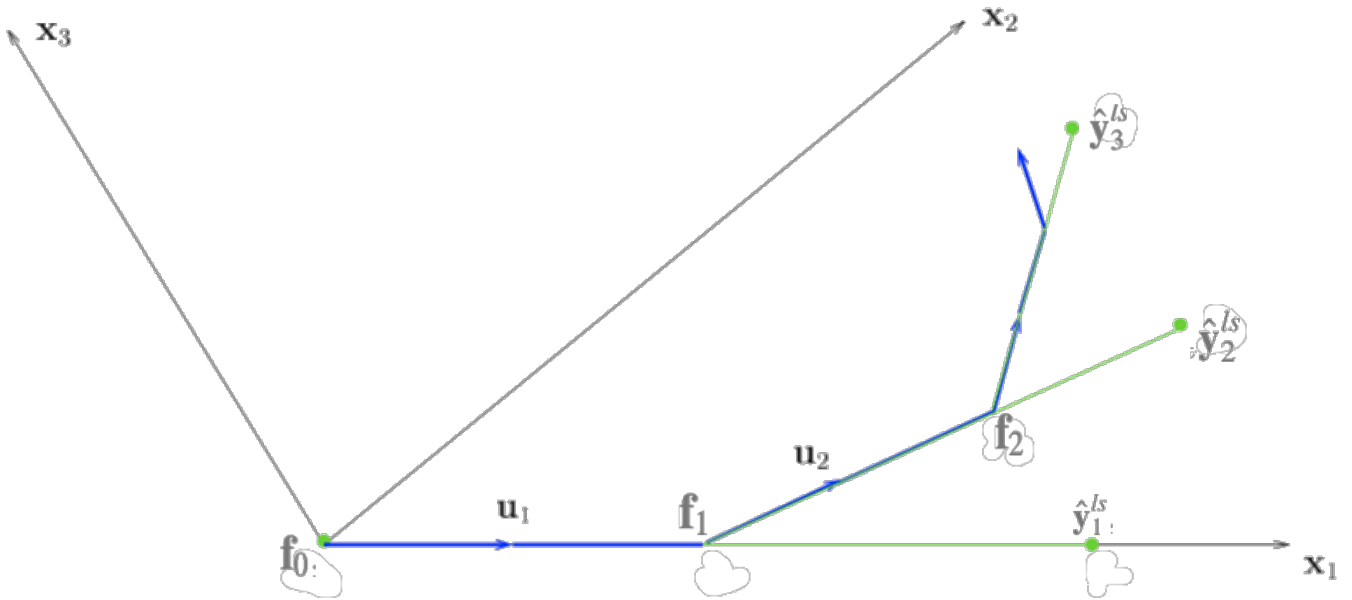
1. **Why called Least Angle**: the direction $\mathbf{u}_k = \mathbf{X}_{\mathcal{A}_k}\delta_k$ that our fit $\mathbf{f}_k(\alpha)$ increases actually has the same angle with any $\mathbf{x}_j \in \mathcal{A}_k$.
2. Note that the left-hand side of the termination condition for searching is a vector, while the right-hand side is a single value.

   $$|\mathbf{X}_{\mathcal{A}_k}\mathbf{r}_k(\alpha)| = max_{\mathbf{x}_j \in \bar{\mathcal{A}}_k} |\mathbf{x}_j^T \mathbf{r}_k(\alpha)|$$

   This actually comes from the fact that the absolute values of correlations of $\mathbf{x}_j \in \mathcal{A}_k, \forall j$ with the residual error are tied and decrease at the same rate.
3. The procedure of searching is approaching the least-squares coefficients of fitting $\mathbf{y}$ on $\mathcal{A}_k$
4. LAR solves the subset selection problem for all $t, s.t. \|\beta\| \leq t$
5. Actually, α can be computed instead of searching

LAR

Example



LAR

Result compared with LASSO

Observations:

When the blue line coefficient cross zero, LAR and LASSO become different.

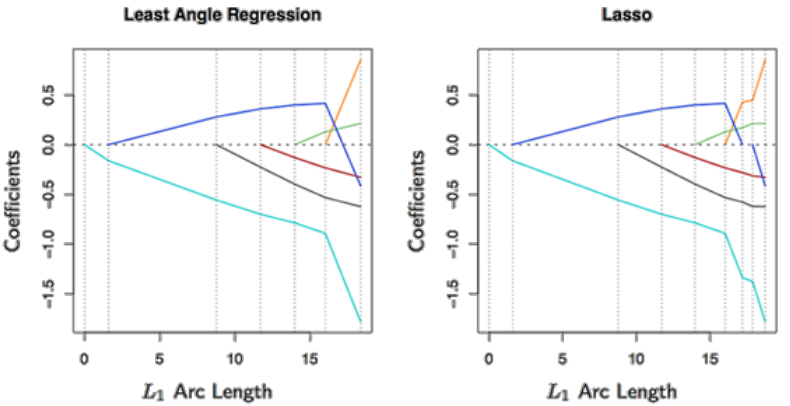


FIGURE 3.15. *Left panel shows the LAR coefficient profiles on the simulated data, as a function of the L_1 arc length. The right panel shows the Lasso profile. They are identical until the dark-blue coefficient crosses zero at an arc length of about* 18.

LAR

Modification for LASSO

During the searching procedure, if a non-zero coefficient hits zero, drop this variable from $\mathcal{A}_k$, and recompute the direction δ_k

Some heuristic analysis

- At a certain time point, we know that all $\mathbf{x}_j \in \mathcal{A}$ share the same absolute values of correlations with the residual error. That is

$$\mathbf{x}_j^T(\mathbf{y} - \mathbf{X}\beta) = \gamma \cdot s_j, \forall \mathbf{x}_j \in \mathcal{A}$$

  where $s_j \in \{-1, 1\}$ indicates the sign of the left hand inner product and γ is the common value. We also know that $|\mathbf{x_j}(\mathbf{y} - \mathbf{X}\beta)| \leq \gamma, \forall \mathbf{x}_j \notin \mathcal{A}$

LAR

Some heuristic analysis

- At a certain time point, we know that all $\mathbf{x}_j \in \mathcal{A}$ share the same absolute values of correlations with the residual error. That is

$$\mathbf{x}_j^T(\mathbf{y} - \mathbf{X}\beta) = \gamma \cdot s_j, \forall \mathbf{x}_j \in \mathcal{A}$$

  where $s_j \in \{-1, 1\}$ indicates the sign of the left hand inner product and γ is the common value. We also know that $|\mathbf{x_j}(\mathbf{y} - \mathbf{X}\beta)| \leq \gamma, \forall \mathbf{x}_j \notin \mathcal{A}$

- Now consider about LASSO for a fixed given λ. Let $\mathcal{B}$ with non-zero coefficients, then we differentiate the objective function w.r.t. these non zero coefficients and set the gradient to zero

$$\mathbf{x}_j^T(\mathbf{y} - \mathbf{X}\beta) = \lambda \cdot sign(\beta_j), \forall j \in \mathcal{B}$$

- They are identical only if $sign(\beta_j)$ matches the sign of the lefthand side. In $\mathcal{A}$, we allow for the β_j, where $sign(\beta_j) \neq sign(\mathbf{x}_j^T(\mathbf{y} - \mathbf{X}\beta))$, while this is forbidden in $\mathcal{B}$. Thus, once a coefficent hits zero, we drop it.

LAR

Some heuristic analysis

- For LAR, we have

$$|\mathbf{x}_j^T(\mathbf{y} - \mathbf{X}\beta)| \leq \gamma, \forall \mathbf{x}_j \notin \mathcal{A}$$

- According to the stationary conditions, for LASSO, we have

$$|\mathbf{x}_j^T(\mathbf{y} - \mathbf{X}\beta)| \leq \lambda, \forall \mathbf{x}_j \notin \mathcal{B}$$

- They match for variables with zero coefficients too.

LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- **Shrinkage Method**
  - Ridge Regression
  - **Lasso**
    - Formulations
    - Comparisons with ridge regression and subset selection
    - Quadratic Programming
    - Least Angle Regression
    - **Viewed as approximation for l_0-regularization**
- Beyond Lasso

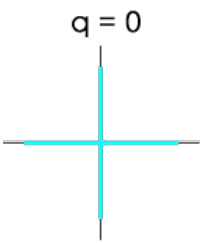
Viewed as approximation for l_0-regularization

Pure variable selection

$$\hat{\beta}^{ridge} = argmin_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \mathbf{x_{ij}} \beta_j)^2, s.t. \#nonzero\beta_j \leq t$$

Actually $\#nonzero\beta_j = \|\beta\|_0$, where

$$\|\beta\|_0 = lim_{q \to 0} (\sum_{j=1}^{p} |\beta_j|^q)^{\frac{1}{q}} = card(\{\beta_j | \beta_j \neq 0\})$$



Viewed as approximation for l_0-regularization

Problem

l_0-norm is not convex, which makes it very hard to optimize.

Solutions

- **LASSO**: Approximated objective function (l_1-norm), with exact optimization
- **Subset selection**: Exact objective function, with approximated optimization (greedy strategy)

Beyond LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- Shrinkage Method
- **Beyond LASSO**
  - **Elastic-Net**
  - **Fused Lasso**
  - **Group Lasso**
  - $l_1 - lp$ **norm**
  - **Graph-guided Lasso**

Beyond LASSO

Elastic Net

- **Formualtion**

$$\lambda \sum_{j=1}^{p} (\alpha \beta_j^2 + (1-\alpha)|\beta_j|)$$

which is a compromise between ridge regression and LASSO.

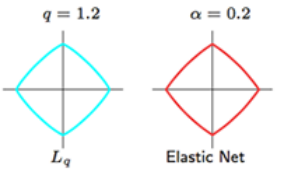


FIGURE 3.13. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for* $q = 1.2$ *(left plot), and the elastic-net penalty* $\sum_j (\alpha\beta_j^2 + (1-\alpha)|\beta_j|)$ *for* $\alpha = 0.2$ *(right plot). Although visually very similar, the elastic-net has sharp (non-differentiable) corners, while the* $q = 1.2$ *penalty does not.*

- **Advantages**
  - The elastic-net selects variables like the lasso, and shrinks together the coefficients of correlated predictors like ridge.
  - It also has considerable computational advantages over the l_q penalties. (detailed in Section 18.4 in Elements of Statistical Learning)

Beyond LASSO

Fused Lasso

- **Intuition**
  - Fused lasso is a generalization that is designed for problems with features that can be ordered in some meaningful way.
  - The fused lasso penalizes the L_1-norm of both the coefficients and their successive differences.
- **Formulation**

$$\hat{\beta} = argmin_{\beta} \{ \|\mathbf{X}\beta - \mathbf{y}\|_2^2 \}$$

$$s.t. \|\beta\| \leq s_1 and \sum_{j=2}^{p} |\beta_j - \beta_{j-1}| \leq s_2$$

Beyond LASSO

Fused Lasso

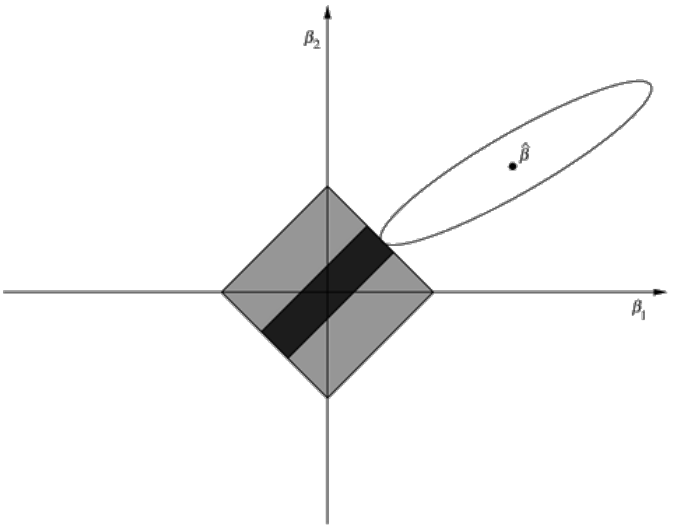


Fig. 2. Schematic diagram of the fused lasso, for the case $N > p = 2$: we seek the first time that the contours of the sum-of-squares loss function (⬭) satisfy $\Sigma_j |\beta_j| = s_1$ (◆) and $\Sigma_j |\beta_j - \beta_{j-1}| = s_2$ (◢)

Beyond LASSO

Group Lasso

- **Intuition**
  - Features are divided into L groups
  - Features within the same group should share similar coefficients
- **Example**
  - Binary dummy variables from one single discrete variable, e.g. $stage\_cancer \in \{1, 2, 3\}$ can be translated into three binary dummy variables $(stage1, stage2, stage3)$
- **Formulations**

$$obj = \|\mathbf{y} - \sum_{l=1}^{L} \mathbf{X}_l \beta_l\|_2^2 + \lambda_1 \sum_{l=1}^{L} \|\beta_l\|_2 + \lambda_2 \|\beta\|_1$$

Beyond LASSO

l_1-l_p penalization

- **Applies to multi-task learning**, where the goal is to estimate predictive models for several related tasks.
- **Examples**
  - **Example 1**: recognize speech of different speakers, or handwriting of different writers,
  - **Example 2**: learn to control a robot for grasping different objects or drive in different landscapes, etc.
- **Assumptions about the tasks**
  - sufficiently different that learning a specific model for each task results in improved performance
  - similar enough that they share some common underlying representation that should make simultaneous learning beneficial.
  - In particular, we focus on the scenario where the different tasks share a subset of relevant features to be selected from a large common space of features.

Beyond LASSO

l_1-l_p penalization

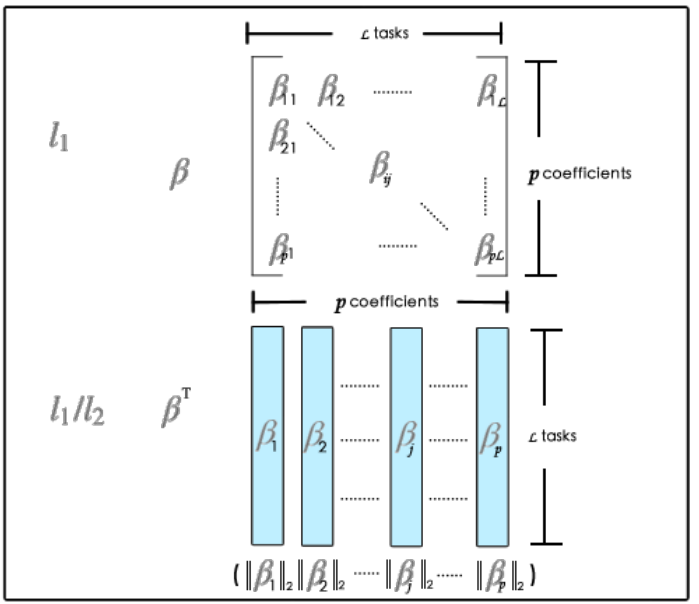
- Formulation
  - $\mathbf{X}_l$: $N \times p$ input matrix for task l and L is the total number of tasks
  - β: $p \times L$ coefficient matrix
  - $\mathbf{y}$: $N \times L$ output matrix
  - objective function

$$obj = \sum_{l=1}^{L} loss(\beta_{:l}, \mathbf{X}_l, \mathbf{y}_{:l}) + \lambda \sum_{j=1}^{p} \|\beta_{:j}\|_2$$

where $loss()$ is some loss function and $\sum_{j=1}^{p} \|\beta_{:j}\|_2$ is equavalent to get the l_1 norm of vector $(\|\beta_{:1}\|_2, \|\beta_{:2}\|_2, \ldots, \|\beta_{:p}\|_2)$.

Beyond LASSO

$l_1 - l_p$ **penalization -Coefficient matrix**



Beyond LASSO

$l_1 - l_p$ **penalization -Norm ball**

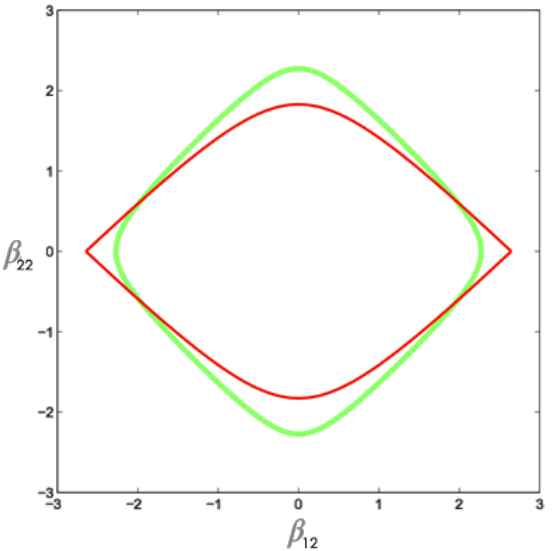


Figure 2: Norm ball induced on the coefficients (β_{12}, β_{22}) for task 2 as feature coefficients for task 1 vary: thin red contour for $(\beta_{11}, \beta_{21}) = (0, 1)$ and thick green contour for $(\beta_{11}, \beta_{21}) = (0.5, 0.5)$.

Beyond LASSO

Graph-Guided LASSO

- **Example**

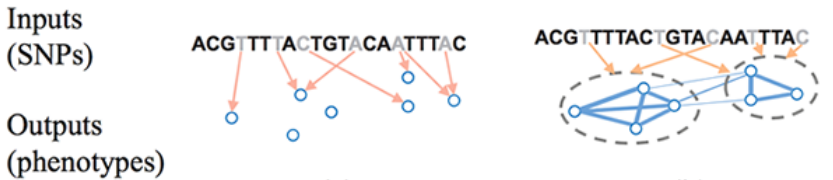


Figure 1: Illustrations of (a) lasso, (b) graph-guided fused lasso.

- **Formulation** Graph-Guided Lasso applies to multi-task settings

$$obj = \sum_{l=1}^{L} loss(\beta_{:l}, \mathbf{X}_l, \mathbf{y}_{:l}) + \lambda \|\beta\|_1 + \gamma \sum_{e=(a,b)\in E}^{p} \tau(r_{ab}) \sum_{j=1}^{p} |\beta_{ja} - sign(r_{a,b})\beta_{jb}|$$

where $r_{a,b} \in \mathbb{R}$ denotes the weight of the edge and $\tau(r)$ can be any positive monotonically increasing function of $|r|$, e.g. $\tau(r) = |r|$.

Beyond LASSO

Graph-Guided LASSO

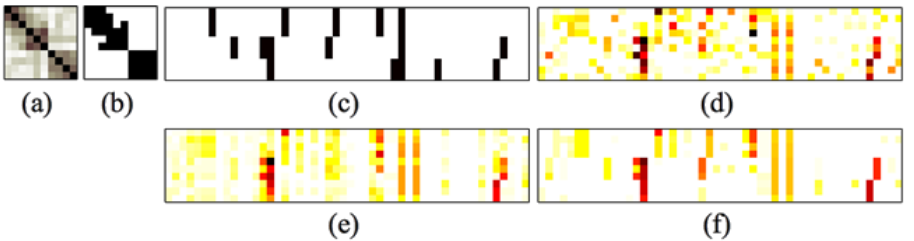


Figure 2: Regression coefficients estimated by different methods based on a single simulated dataset. $b = 0.8$ and threshold $\rho = 0.3$ for the output correlation graph are used. Red pixels indicate large values. (a) The correlation coefficient matrix of phenotypes, (b) the edges of the phenotype correlation graph obtained at threshold 0.3 are shown as white pixels, (c) the true regression coefficients used in simulation. Absolute values of the estimated regression coefficients are shown for (d) lasso, (e) ℓ_1/ℓ_2-regularized multi-task regression, (f) GFlasso. Rows correspond to outputs and columns to inputs.

Sparse Models

Thank You!