Sparse Models

CMPUT 466/551

Ping Jin (pjin1@ualberta.ca)

Outline

- **Introduction to Dimension Reduction**
- **Linear Regression and Least Squares (Review)**
- **Subset Selection**
- **Shrinkage Methods**
- **Beyond LASSO**

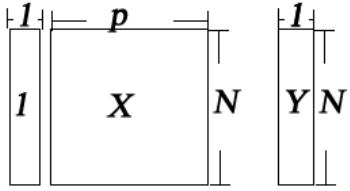
Part 1: Introduction to Dimension Reduction

1. **Introduction to Dimension Reduction**
   - **General notations**
   - **Motivations**
   - **Feature selection and feature extraction**
   - **Feature Selection**
     - **Wrapper method**
     - **Filter method**
     - **Embedded method**
   - **Feature Extraction**
     - **PCA, ICA...**
2. Linear Regression and Least Squares (Review)
3. Subset Selection
4. Shrinkage Methods
5. Beyond LASSO

General Notations

Dataset

- **X**: columnwise centered $N \times p$ matrix
  - N : # samples, p : # features
  - An intercept vector **1** is added to **X**, then **X** is $N \times (p+1)$ matrix
- **y**: $N \times 1$ vector of labels(classification) or continous values(regression)



Basic Model

- Linear Regression
  - Assumption: the regression function $E(Y|X)$ is linear

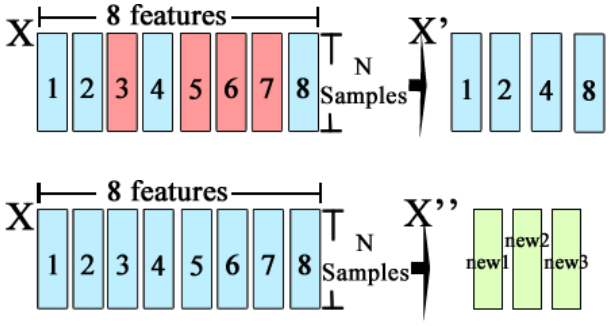
$$f(X) = X\beta$$

  - β: $(p+1) \times 1$ vector of coefficients

Motivations

- Dimension reduction is about transforming data with high dimensionality into data of much lower dimensionality
  - **Computational efficiency**: less dimensions require less computations
  - **Accuracy**: lower risk of overfitting

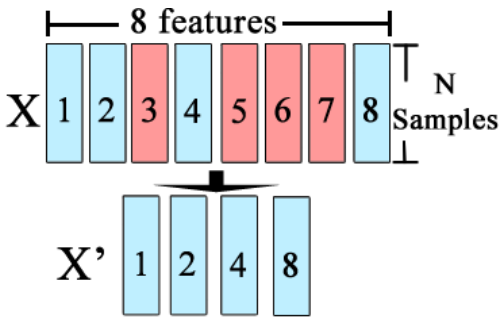
- **Categories**
  - Feature Selection:
    - chooses a subset of features from the original feature set
  - Feature Extraction:
    - transforms the original features into new ones, linearly or non-linearly
    - e.g. PCA, ICA, etc.



Feature Selection and Feature Extraction

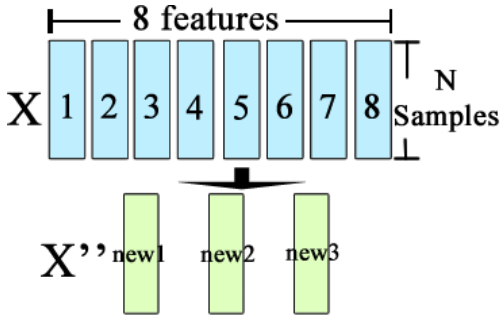
Feature Selection

- Easier to interpret
- Reduces cost: computation, budget, etc.



Feature Extraction

- More flexible. Feature selection is a special case of linear feature extraction



Feature Selection and Feature Extraction

Example 1: Prostate Cancer

- **Response**: level of prostate-specific antigen (lpsa).
- **Initial Feature Set**:

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

- **Task**:
  - predict *lpsa* from measurements of features

Feature selection

- Cost: Measuring features cost money
- Interpretation: Doctors can see which features are important

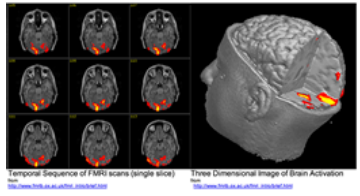
Feature Selection and Feature Extraction

Example 2: classification with fMRI data

- fMRI data are 4D images, with one dimension being time.
- Each image is ~ $50 \times 50 \times 50$ (spatial) $\times 200$(times) $= 25M$ dimensions

Feature extraction

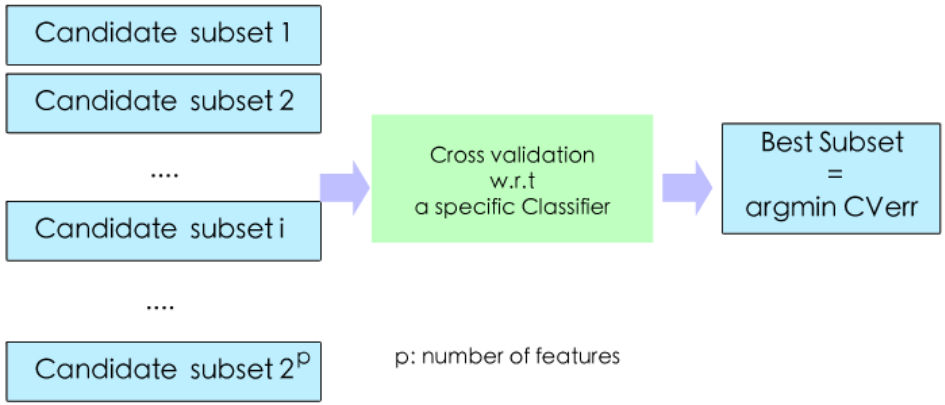
- Individual voxel-times are not important
- Cost is not correlated with #features
- Feature extraction offers more flexibility in transforming features, which potentially results in better accuracy



Feature Selection Methods

Wrapper Methods

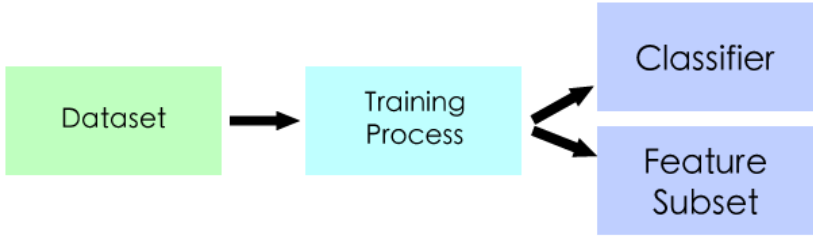
- search the space of feature subsets
- use the training/validation accuracy of a particular classifier as the measure of utility for a candidate subset



Feature Selection Methods

Embedded Methods

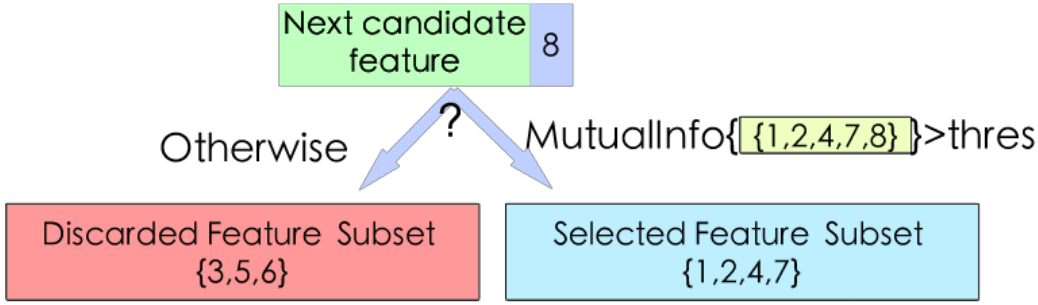
- exploit the structure of specific classes of learning models to guide the feature selection process
- embedded as part of the model construction process
  - e.g. LASSO.



Feature Selection Methods

Filter Methods

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



Feature Selection

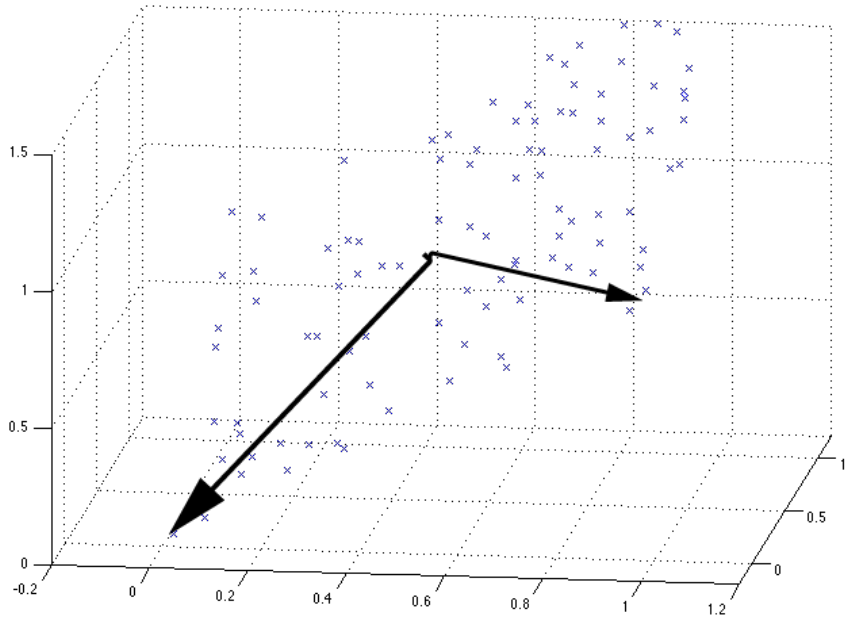
Comparison

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

Feature Extraction

Principle Components Analysis

- **A graphical explanation**
  - Each data sample has three features
  - Original features are transformed into new ones
  - Often use only the new features with largest variance
- **Example**
  - For fMRI images, we usually have millions of dimensions. PCA can project the data from millions of dimensions to only thousands of dimensions, or even less
- Other feature extraction methods: ICA, Kernel PCA , etc..



Part 2: Linear Regression and Least Squares (Review)

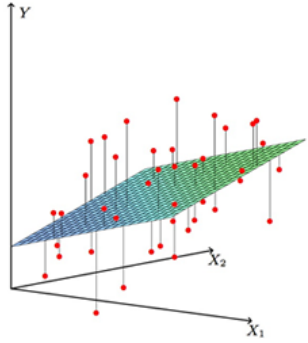
1. Introduction to Dimension Reduction
2. **Linear Regression and Least Squares (Review)**
   - **Least Square Fit**
   - **Gauss Markov**
   - **Bias-Variance tradeoff**
   - **Problems**
3. Subset Selection
4. Shrinkage Methods
5. Beyond LASSO

Linear Regression and Least Squares (Review)

Least Squares Fit

$$RSS(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta)$$

$$\frac{\partial RSS}{\partial \beta} = -2\mathbf{X}^T(\mathbf{y} - \mathbf{X}\beta) = 0 \quad \Rightarrow \quad \hat{\beta}^{ls} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

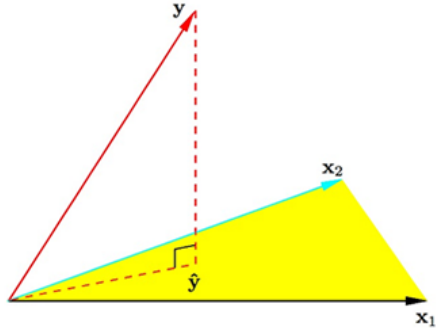


Gauss Markov Theorem

The least squares estimates $\hat{\beta}^{ls}$ of the parameters β have the smallest variance among all linear unbiased estimates.

Question

Is it good to be unbiased?



Linear Regression and Least Squares (Review)

Bias-Variance tradeoff

$$\begin{aligned} MSE(\hat{\mathbf{y}}) &= E[(\hat{\mathbf{y}} - Y)^2] \\ &= Var(\hat{\mathbf{y}}) + [E[\hat{\mathbf{y}}] - Y]^2 \end{aligned}$$

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

Problems of Least Squares

- **Prediction accuracy**: unbiased, but high variance compared to many biased estimators, overfitting noise and sensitive to outlier
- **Interpretation**: $\hat{\beta}$ involves all of the features. Better to have SIMPLER linear model, that involves only a few features...
- Recall that $\hat{\beta}^{ls} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$
  - $(\mathbf{X}^T\mathbf{X})$ may be **not invertible** and thus no closed form solution

Part 3: Subset Selection Methods

1. Introduction to Dimension Reduction
2. Linear Regression and Least Squares (Review)
3. **Subset Selection**
   - **Best-subset selection**
   - **Forward stepwise selection**
   - **Forward stagewise selection**
   - **Problems**
4. Shrinkage Methods
5. Beyond LASSO

Subset Selection Methods

Best-subset selection

- Best subset regression finds for each $k \in \{0, 1, 2, \ldots, p\}$ the subset of features of size k that gives smallest RSS.
- Then cross validation is utilized to choose the best k
- An efficient algorithm, the leaps and bounds procedure (Furnival and Wilson, 1974), makes this feasible for p as large as 30 or 40.

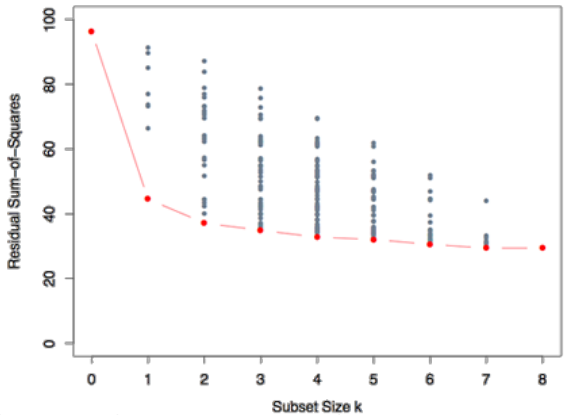


FIGURE 3.5. *All possible subset models for the prostate cancer example. At each subset size is shown the residual sum-of-squares for each model of that size.*

Subset Selection Methods

Forward-STEPWISE selection

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

- a **sequential greedy** algorithm.

Forward-Stepwise Selection builds a model sequentially, adding one variable at a time.

- Initialization
  - Active set $\mathcal{A} = \emptyset$, $\mathbf{r} = \mathbf{y}$, $\beta = 0$
- At each step, it
  - identifies the best variable (with the highest correlation with the residual error)

$$\mathbf{k} = argmax_j(|correlation(\mathbf{x}_j, \mathbf{r})|)$$

  - $A = A \cup \mathbf{k}$
  - then updates the least squares fit β, $\mathbf{r}$ to include all the active variables

Subset Selection Methods

Forward-STAGEWISE Regression

- Initialize the fit vector $\mathbf{f} = 0$
- For each time step
  - Compute the correlation vector

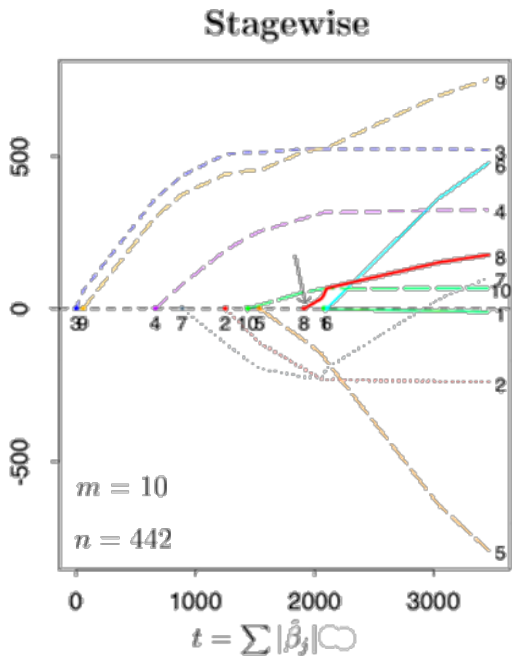
$$\mathbf{c} = (\mathbf{c}_1, .. \mathbf{c}_p)$$

    - $\mathbf{c}_j$ represents the correlation between $\mathbf{x}_j$ and the residual error
  - $k = argmax_{j \in \{1,2,..,p\}} |\mathbf{c}_j|$
  - Coefficients and fit vector are updated

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

$$\beta_k \leftarrow \beta_k + \alpha \cdot sign(\mathbf{c}_k)$$

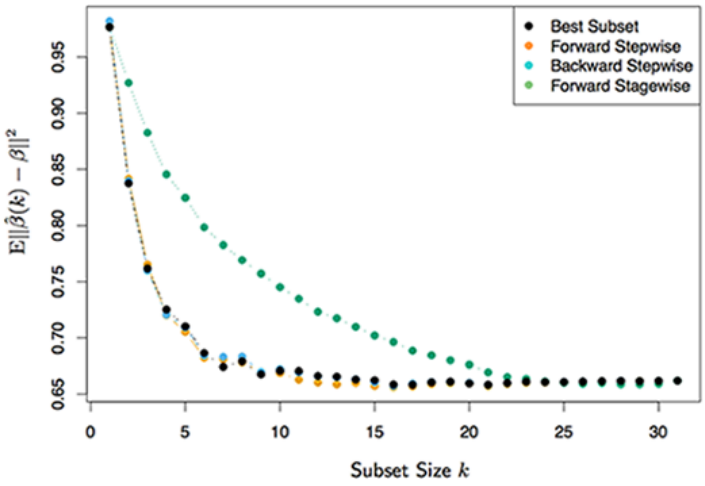
where α is the learning rate



Subset Selection Methods

Comparison

- Forward-STEPWISE selection:
  - algorithm stops in p steps
- Forward-STAGEWISE selection:
  - is a slow fitting algorithm, at each time step, only β_k is updated. Alg can take more than p steps to stop



- $N = 300$ Observations, $p = 31$ features
- averaged over 50 simulations

Summary of Subset Selection Methods

Advantages w.r.t Least Squares

- More interpretable result
- More compact model

Disadvantages

- It is a discrete process, and thus has high variance and sensitivity to changes in the dataset
  - If the dataset changes a little, the feature selection result may be very different
- Thus may not lower prediction error as much

Part 4: Shrinkage Methods

1. Introduction to Dimension Reduction
2. Linear Regression and Least Squares (Review)
3. Subset Selection
4. **Shrinkage Methods**
   - **Ridge Regression**
     - **Formulations and closed form solution**
     - **Singular value decomposition**
     - **Degree of Freedom**
   - LASSO
5. Beyond LASSO

Ridge Regression

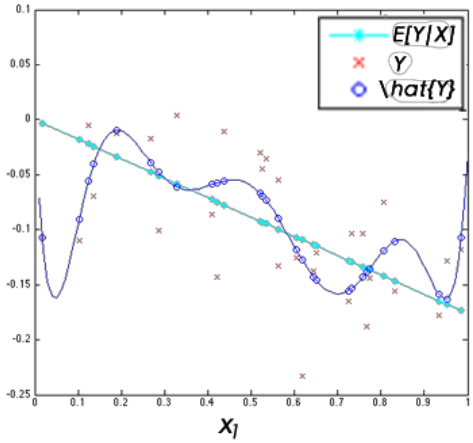
- Least squares with quadratic constraints

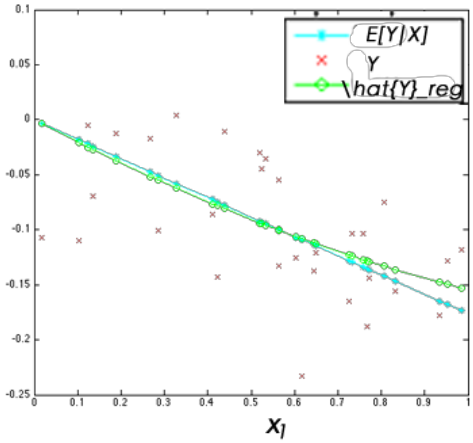
$$\hat{\beta}^{ridge} = argmin_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \mathbf{x}_{ij}\beta_j)^2, \quad s.t. \quad \sum_{j=1}^{p} \beta_j^2 \leq t$$

- Simulation Experiment
  - $N = 30$
  - $\mathbf{x}_1 \sim N(0, 1)$, $\mathbf{x}_2 = \mathbf{x}_1^2$
  - $\beta \sim U(-0.5, 0.5)$
  - $Y = (\mathbf{x}_1, \mathbf{x}_2) \times \beta$
  - $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_1^2, \ldots, \mathbf{x}_1^8)$

Ridge Regression

- Simulation Experiment





Ridge Regression

- **Least squares with quadratic constraints**

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

- **Its Lagrange 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, our solution as the posterior means

- **Solution**

$$RSS(\beta) = (\mathbf{y} - \mathbf{X}\beta)^T (\mathbf{y} - \mathbf{X}\beta) + \lambda \beta^T \beta$$
$$\partial RSS(\beta)/\partial \beta = 0 \quad \Rightarrow \quad \hat{\beta}^{ridge} = (\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$$

Ridge Regression

Singular Value Decomposition (SVD)

SVD offers some additional insight into the nature of ridge regression.

- **The SVD of X**:

$$\mathbf{X} = \mathbf{UDV}^T$$

  - **U**: $N \times p$ **orthogonal** matrix with columns spanning the column space of **X**.
    - $\mathbf{u}_j$ is the jth column of **U**
  - **V**: $p \times p$ **orthogonal** matrix with columns spanning the row space of **X**.
    - $\mathbf{v}_j$ is the jth column of **V**
  - **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 **X**

$$\underset{X}{\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}} = \underset{U}{\begin{bmatrix} 0.85 & -0.52 \\ 0.53 & 0.85 \end{bmatrix}} \underset{D}{\begin{bmatrix} 1.62 & 0 \\ 0 & 0.62 \end{bmatrix}} \underset{V}{\begin{bmatrix} 0.53 & -0.85 \\ 0.85 & 0.53 \end{bmatrix}}$$

$$X \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Ridge Regression

Singular Value Decomposition (SVD)

- **For least squares**

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

- **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 squares, we have an additional shrinkage term

$$\frac{d_j^2}{d_j^2+\lambda},$$

the smaller d_j is and the larger λ is, the more shrinkage we have.

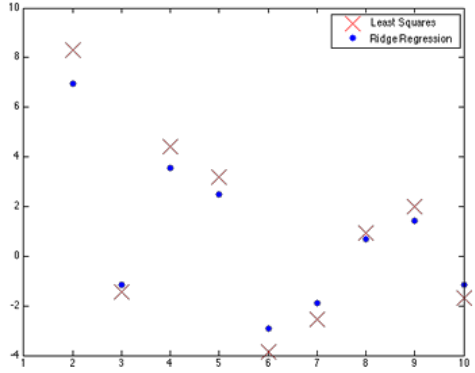
Ridge Regression

Singular Value Decomposition (SVD)

- $N = 100, p = 10$

$$\mathbf{X}\hat{\beta}^{ls} = u_1 \times (u_1^T y) + u_2 \times (u_2^T y) + \cdots + u_3 \times (u_p^T y)$$

$$\mathbf{X}\hat{\beta}^{ridge} = u_1 \times (u_1^T y) \frac{d_1^2}{d_1^2 + \lambda} + u_2 \times (u_2^T y) \frac{d_2^2}{d_2^2 + \lambda} + \cdots + u_3 \times (u_p^T y) \frac{d_p^2}{d_p^2 + \lambda}$$



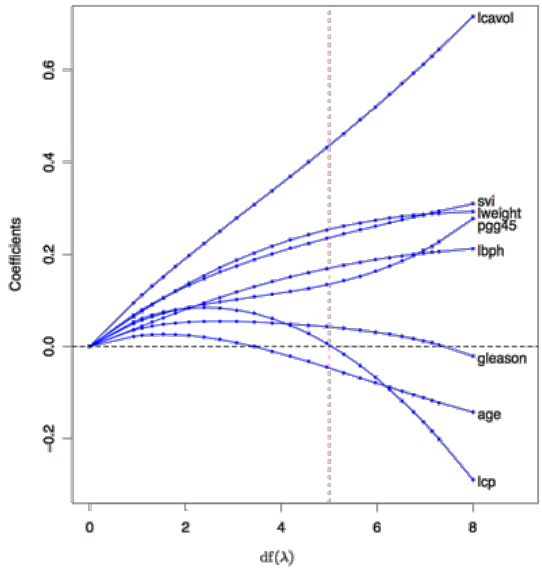
Ridge Regression

Degree of Freedom

- The number of degrees of freedom is the number of values in the final calculation of a statistic that are free to vary. The degree of freedom of ridge estimate is related to λ, thus defined as $df(\lambda)$.
- Computation

$$df(\lambda) = tr[\mathbf{X}(\mathbf{X}^{\mathbf{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathbf{T}}]$$
$$= \sum_{j=1}^{p} \frac{d_j^2}{d_j^2 + \lambda}$$

- [larger λ] → [smaller $df(\lambda)$] → [more constrained model]
- The redline gives the best $df(\lambda)$ identified from cross validation



Ridge Regression

Advantages

- w.r.t. Least Squares
  - $(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})$ is always inveritible and thus the closed form solution always exist
  - Ridge regression controls the complexity with regularization term via λ, which is less prone to overfitting compared with least squares fit,
    - e.g. sometimes a wildly large coefficient on one variable can be cancelled by another wildly large coefficient of a correlated variable
  - Possibly higher prediction accuracy, as the estimates of ridge regression trade a little bias for less variance
- w.r.t. Subset Selection Methods
  - Ridge regression is a continous shrinkage method which has less variance than subset selection methods

Disadvantages w.r.t. Subset Selection Methods

- Interpretability and compactness: Though coefficients are shrunk, but not to zero. Unlike methods that select part of the features, ridge regression may encounter efficiency issue and offer little interpretations in high dimensional problems.

Part 4: Shrinkage Methods - LASSO

1. Introduction to Dimension Reduction
2. Linear Regression and Least Squares (Review)
3. Subset Selection
4. **Shrinkage Methods**
   - Ridge Regression
   - **LASSO**
     - **Formulations**
     - **Comparisons with ridge regression and subset selection**
     - **Quadratic Programming**
     - **Least Angle Regression**
     - **Viewed as approximation for l_0-regularization**
5. Beyond LASSO

LASSO

Linear regression with l_1-regularization

- **Formulations**

  - **Least squares with constraints**

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

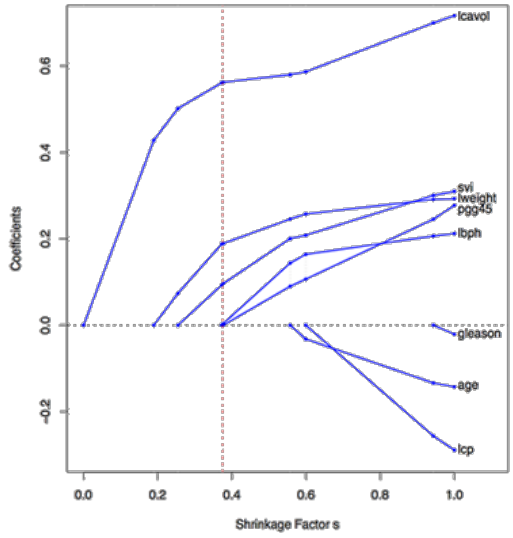
  - **Its Lagrange form**

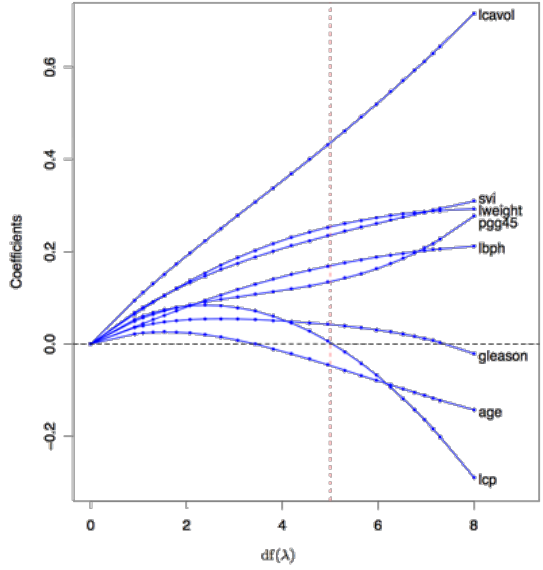
$$\hat{\beta}^{LASSO} = argmin_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} \mathbf{x_{ij}} \beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

  - The l_1-regularization can be viewed as a Laplace prior on the coefficients

LASSO

- $s = \frac{t}{\sum_{j=1}^{p} |\hat{\beta}_j|}$, where $\hat{\beta}$ is the least squares estimate
- Redlines represent the s and $df(\lambda)$ with the best cross validation error





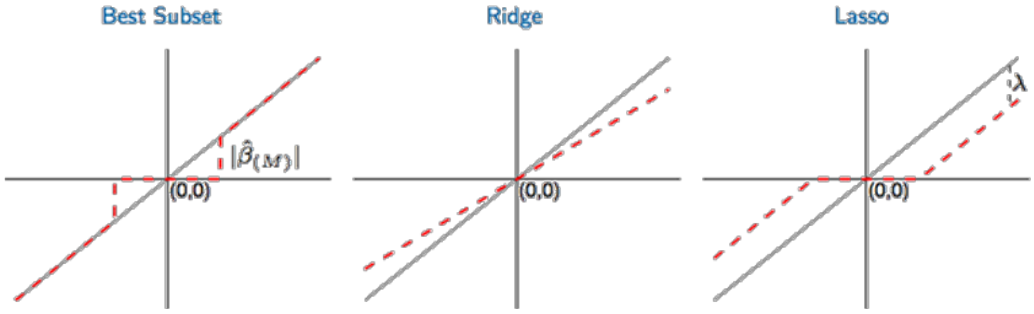
LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- Subset Selection
- **Shrinkage Methods**
  - 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

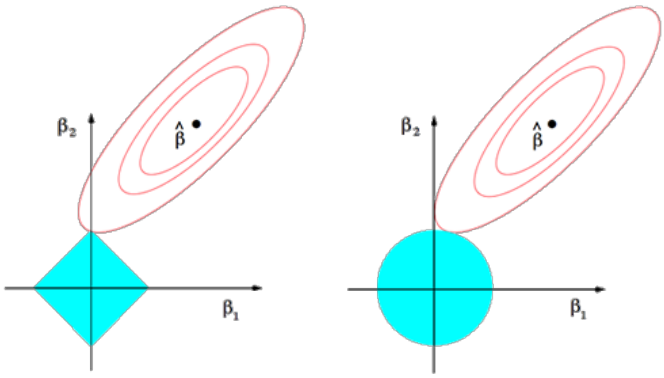
- **Orthonormal Input X**
  - **Best subset**: [Hard thresholding] keeps the top M largest coefficeints of $\hat{\beta}^{ls}$
  - **Ridge**: [Pure shrinkage] does proportional shrinkage of $\hat{\beta}^{ls}$
  - **LASSO**: [Soft thresholding] translates each coefficient of $\hat{\beta}^{ls}$ by λ towards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



LASSO

Comparison

- **Non-orthonormal Input X**



Solid blue area: the constraints

- left: $|\beta_1| + |\beta_1| \leq t$
- right: $\beta_1^2 + \beta_1^2 \leq t^2$

$\hat{\beta}$: least squares fit

LASSO

Other unit circles for different p-norms

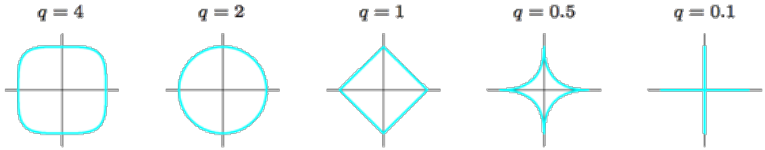


FIGURE 3.12. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for given values of* q.

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

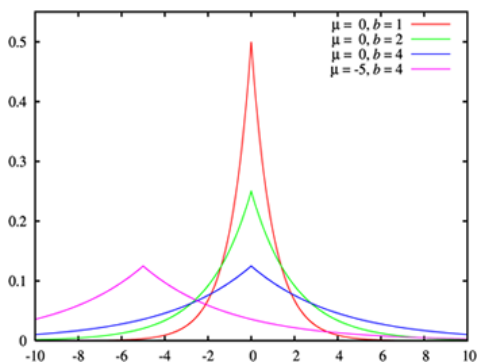
Here $q = 0$ is the pure variable selection procedure, as it is counting the **number of non-zero coefficients**.

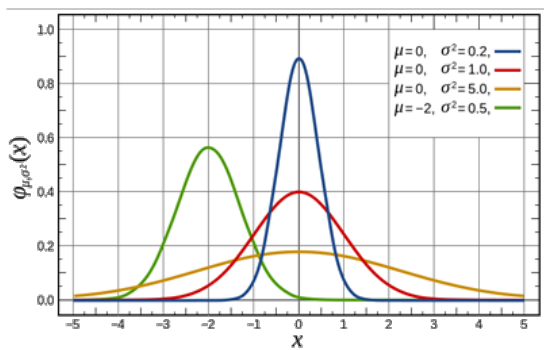
LASSO

Regularizations as priors

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

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





LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- Subset Selection
- **Shrinkage Methods**
  - 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 for a given λ.
  - Later in these slides, a method called least angle regression can solve LASSO for all λ

LASSO

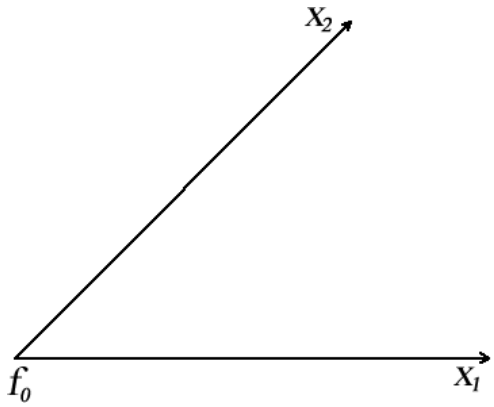
- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- Subset Selection
- **Shrinkage Methods**
  - Ridge Regression
  - **LASSO**
    - Formulations
    - Comparisons with ridge regression and subset selection
    - Quadratic Programming
    - **Least Angle Regression**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

LAR Algorithm

- Initialization:
  - Standardized all predictors s.t. $\bar{\mathbf{x}}_j = 0, \mathbf{x}_j^T \mathbf{x}_j = 1$; $\mathbf{r}_0 = \mathbf{y} - \bar{\mathbf{y}}$; $\beta = \mathbf{0}$; $\mathbf{f}_0 = \mathbf{0}$;
  - $k = argmax_j |\mathbf{x}_j^T \mathbf{r}_0|$, $\mathcal{A}_1 = \{k\}$
- Main
  - for time step $t = 1, 2, \ldots min(N-1, p)$
    - $\mathbf{r}_t = \mathbf{y} - \mathbf{X}_{\mathcal{A}_t} \beta_{\mathcal{A}_t}$, $\mathbf{f}_t = \mathbf{X}_{\mathcal{A}_t} \beta_{\mathcal{A}_t}$
    - $\delta_t = (\mathbf{X}_{\mathcal{A}_t}^{\mathbf{T}} \mathbf{X}_{\mathcal{A}_t})^{-1} \mathbf{X}_{\mathcal{A}_t}^{\mathbf{T}} \mathbf{r_t}$, $\mathbf{u}_t = \mathbf{X}_{\mathcal{A}_t} \delta_t$
    - Search α
      - $\beta_{\mathcal{A}_t}(\alpha) = \beta_{\mathcal{A}_t} + \alpha \cdot \delta_t$
      - Concurrently, $\mathbf{f}_t(\alpha) = \mathbf{f}_t + \alpha \cdot \mathbf{u}_t$
    - Until $|\mathbf{X}_{\mathcal{A}_t} \mathbf{r}_t(\alpha)| = max_{\mathbf{x}_j \in \bar{\mathcal{A}}_t} |\mathbf{x}_j^T \mathbf{r}_t(\alpha)|$
    - $k = argmax_{j \in \bar{\mathcal{A}}_t} |\mathbf{x}_j \mathbf{r}_t(\alpha)|$,
    - $\mathcal{A}_{t+1} = \mathcal{A}_t \cup \{k\}$

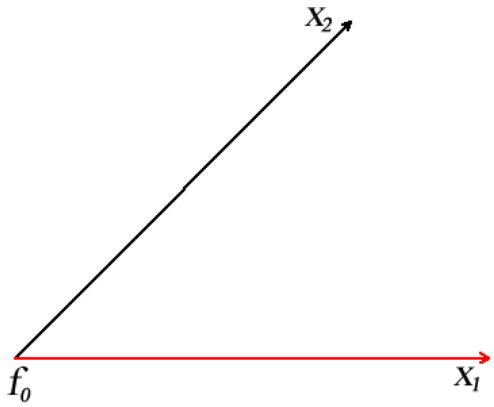
- $\mathcal{A}_t$: active set, the set indices of features we already included in the model at time step t.
  - $\bar{\mathcal{A}}_t = \{1, 2, \ldots, p\} - \mathcal{A}_t$
- α: searching parameter within a time step
- $\beta_{\mathcal{A}_t}$: coefficients vector at the beginning of time step t
- $\beta_{\mathcal{A}_t}(\alpha)$: coefficients vector in time step t w.r.t. α
- $\mathbf{f}_t$: the fit vector at the beginning of time step t, $\mathbf{f}_0 = 0$
- $\mathbf{f}_t(\alpha)$: the fit vector in time step t w.r.t. α
- $\mathbf{r}_t$: residual vector at the beginning of time step t, $\mathbf{r}_0 = \mathbf{y} - \bar{\mathbf{y}}$, where $\bar{\mathbf{y}} = average(\mathbf{y})$
- $\mathbf{r}_t(\alpha)$: residual vector in time step t, w.r.t. α

LAR - Example



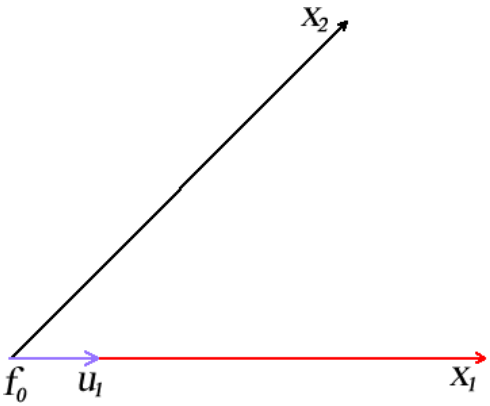
- Initialization:
  - Standardized each columns of $\mathbf{X}$
    - s.t. $\bar{\mathbf{x}}_j = 0$, $\mathbf{x}_j^T \mathbf{x}_j = 1$
  - $\mathbf{r}_0 = \mathbf{y} - \bar{\mathbf{y}}$;
  - $\beta = (0, 0)^T$;
  - $\mathcal{A}_0 = \varnothing$
  - $\mathbf{f}_0$ is the current fit at time 0
    - $\mathbf{f}_0 = (0, 0)^T$

LAR - Example



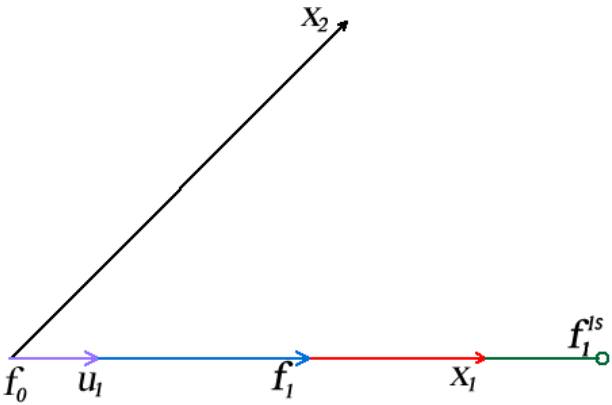
- $k = argmax_j |\mathbf{x}_j^T \mathbf{r}_0| = 1$
- $\mathcal{A}_1 = \{1\}$

LAR - Example



- $\mathbf{r}_1 = \mathbf{y} - \mathbf{X}_{\mathcal{A}_1} \beta_{\mathcal{A}_1}$
- $\delta_1 = (\mathbf{X}_{\mathcal{A}_1}^{\mathbf{T}} \mathbf{X}_{\mathcal{A}_1})^{-1} \mathbf{X}_{\mathcal{A}_1}^{\mathbf{T}} \mathbf{r}_1$
- $\mathbf{u}_1 = \mathbf{X}_{\mathcal{A}_1} \delta_1$

LAR - Example



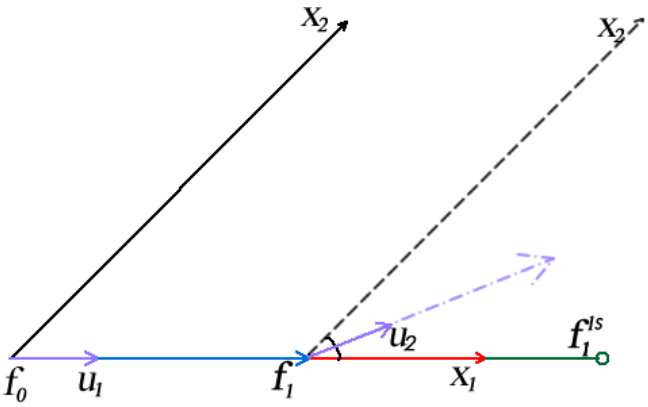
Explanations

- Search α
  - $\beta_{\mathcal{A}_1}(\alpha) = \beta_{\mathcal{A}_1} + \alpha \cdot \delta_1$,
  - $\mathbf{f}_1(\alpha) = \mathbf{f}_1 + \alpha \cdot \mathbf{u}_1$
  - $\mathbf{r}_1(\alpha) = \mathbf{y} - \mathbf{X}_{\mathcal{A}_1}\beta_{\mathcal{A}_1}(\alpha)$
- Until $|\mathbf{X}_{\mathcal{A}_1}\mathbf{r}_1(\alpha)| = max_{j\in\bar{\mathcal{A}}_1}|\mathbf{x}_j^T\mathbf{r}_1(\alpha)|$
- $2 = argmax_{j\in\bar{\mathcal{A}}_1}|\mathbf{x}_j^T\mathbf{r}_1(\alpha)|$
- $\mathcal{A}_2 = \{1, 2\}$

Comments

- $\mathbf{f}_t$ is approaching $\mathbf{f}_t^{ls}$, but never reaches it, except for the final step of LAR

LAR - Example



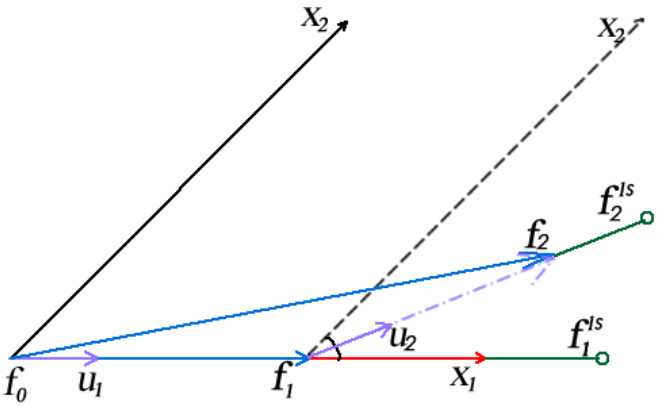
Explanations

- $\mathbf{r}_2 = \mathbf{y} - \mathbf{X}_{\mathcal{A}_2}\beta_{\mathcal{A}_2}$
- $\delta_2 = (\mathbf{X}_{\mathcal{A}_2}^{\mathbf{T}}\mathbf{X}_{\mathcal{A}_2})^{-1}\mathbf{X}_{\mathcal{A}_2}^{\mathbf{T}}\mathbf{r}_2$
- $\mathbf{u}_2 = \mathbf{X}_{\mathcal{A}_2}\delta_2$

Comments

- 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 $\mathbf{x}_j \in \mathcal{A}_k$.

LAR - Example



- If $p = 2$
  - $\mathbf{f}_2 = \mathbf{f}_2^{ls}$
- The absolute values of correlations of $\mathbf{x}_j \in \mathcal{A}_k, \forall j$ with the residual error $\mathbf{r}_t \alpha$ are tied and decrease at the same rate during searching α.

LAR

More Comments

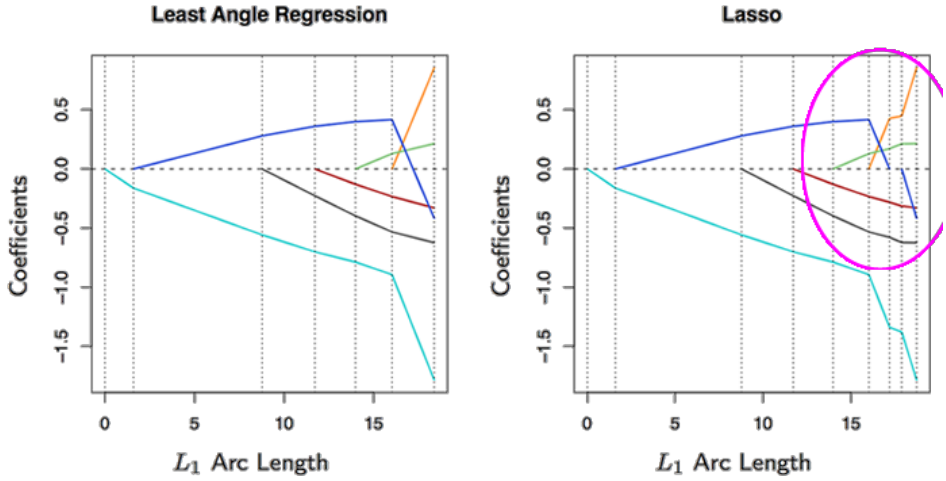
- The procedure of searching is approaching the least-squares coefficients of fitting $\mathbf{y}$ on $\mathcal{A}_k$
- LAR solves the subset selection problem for all $t, s.t. \|\beta\| \leq t$
- Actually, α can be computed instead of searching
- LAR algorithm ends in $min(p, N-1)$ steps

LAR

Result compared with LASSO

Observations

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

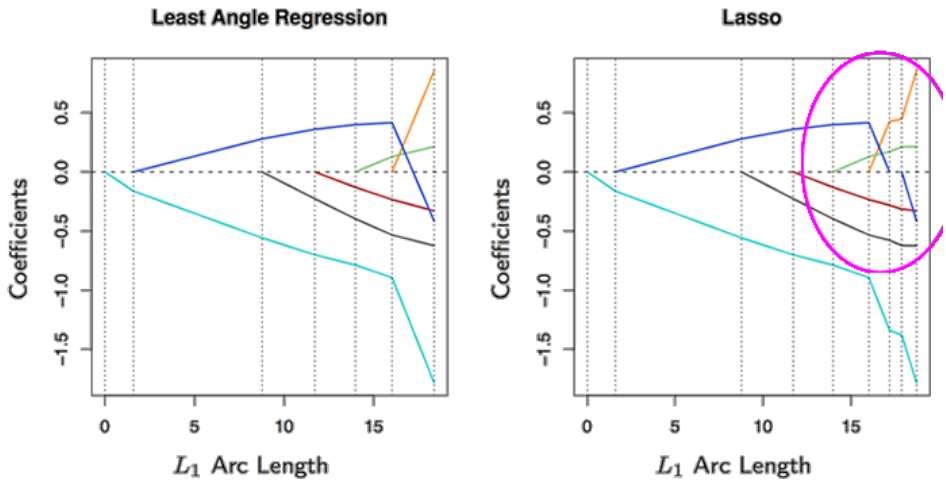


LAR

Result compared with LASSO

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



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

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

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

- Consider LASSO for a fixed λ. Let $\mathcal{B}$ be the set of indices of non-zero coefficients, then we differentiate the objective function w.r.t. those coefficients in $\mathcal{B}$ and set the gradient to zero. We have

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

- 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 $\mathcal{B}$.

LAR

Some heuristic analysis

- For LAR, we have

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

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

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

- LAR and LASSO match for variables with zero coefficients too.

LASSO

- Introduction to Dimension Reduction
- Linear Regression and Least Squares (Review)
- Subset Selection
- **Shrinkage Methods**
  - 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

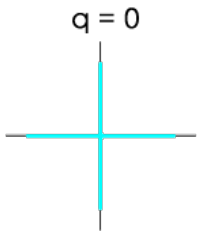
Viewed as approximation for l_0-regularization

Pure variable selection

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

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

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



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)

Part 5: Beyond LASSO

1. Introduction to Dimension Reduction
2. Linear Regression and Least Squares (Review)
3. Subset Selection
4. Shrinkage Methods
5. **Beyond LASSO**
   - **Elastic-Net**
   - **Fused LASSO**
   - **Group LASSO**
   - $l_1 - lp$ **norm**
   - **Graph-guided LASSO**

Beyond LASSO - Elastic Net

Problems with LASSO

- LASSO tends to rather arbitrarily select one of a group of highly correlated variables (see how LAR works). Sometimes, it is better to select **ALL** the relevant varibles in a group
- LASSO selects at most N variables, when $p > N$, which may be undisirable when $p >> N$
- The performance of Ridge dominates that of LASSO, when $N > p$ and variables are correlated

Elastic Net

- **Penalty Term**

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

which is a compromise between ridge regression and LASSO and $\alpha \in [0, 1]$.

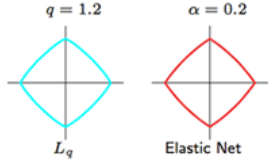


FIGURE 3.13. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for* $q = 1.2$ *(left plot), and the elastic-net penalty* $\sum_j (\alpha\beta_j^2 + (1-\alpha)|\beta_j|)$ *for* $\alpha = 0.2$ *(right plot). Although visually very similar, the elastic-net has sharp (non-differentiable) corners, while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

Beyond LASSO - Elastic Net

Advantages of E-Net

- solves above problems
- selects variables like LASSO, and shrinks together the coefficients of correlated predictors like ridge.
- has considerable computational advantages over the l_q penalties.
  - See 18.4 [Elements of Statistical Learning]

Elastic Net

- **Penalty Term**

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

which is a compromise between ridge regression and LASSO and $\alpha \in [0, 1]$.

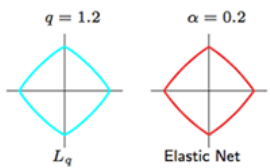


FIGURE 3.13. *Contours of constant value of* $\sum_j |\beta_j|^q$ *for* $q = 1.2$ *(left plot), and the elastic-net penalty* $\sum_j (\alpha\beta_j^2 + (1-\alpha)|\beta_j|)$ *for* $\alpha = 0.2$ *(right plot). Although visually very similar, the elastic-net has sharp (non-differentiable) corners, while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

Elastic Net - A simple illustration

- Two independent "hidden" factors $\mathbf{z}_1$ and $\mathbf{z}_2$

$$\mathbf{z}_1 \sim U(0, 20), \quad \mathbf{z}_2 \sim U(0, 20),$$

- Generate the response vector $\mathbf{y} = \mathbf{z}_1 + 0.1\mathbf{z}_2 + N(0, 1)$
- Suppose the observed features are

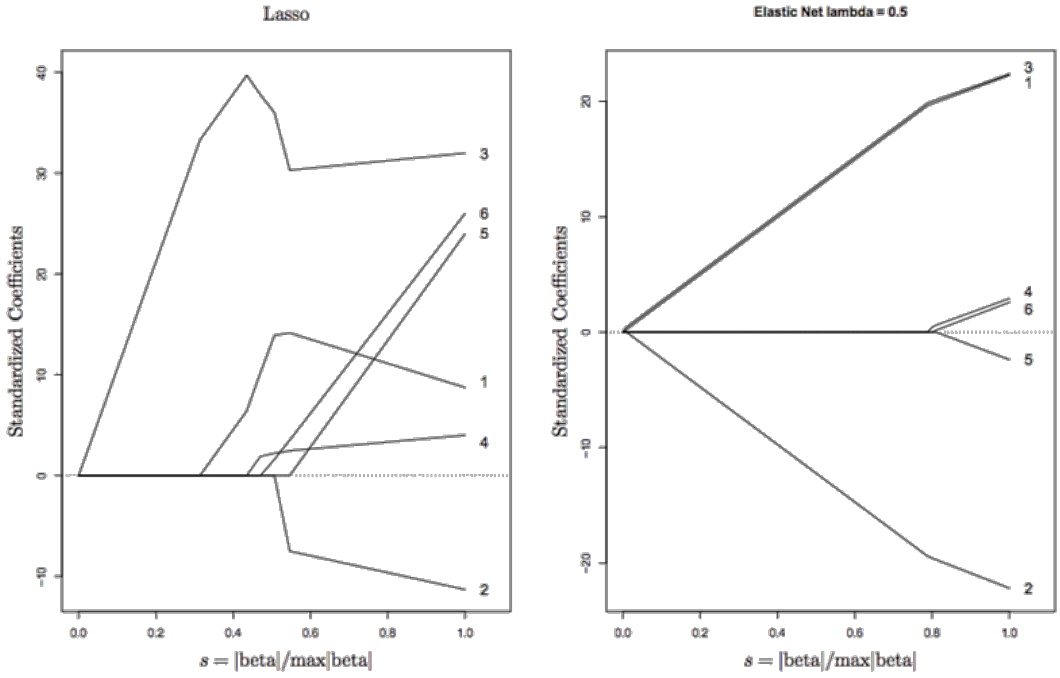
$$\mathbf{x}_1 = \mathbf{z}_1 + \epsilon_1, \quad \mathbf{x}_2 = -\mathbf{z}_1 + \epsilon_2, \quad \mathbf{x}_3 = \mathbf{z}_1 + \epsilon_3$$

$$\mathbf{x}_4 = \mathbf{z}_2 + \epsilon_4, \quad \mathbf{x}_5 = -\mathbf{z}_2 + \epsilon_5, \quad \mathbf{x}_6 = \mathbf{z}_2 + \epsilon_6$$

  where ϵ is $i.i.d.$ random noise.

- Fit the model on data $(\mathbf{X}, \mathbf{y})$
- A good model should identify that only $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ are important

Elastic Net - A simple illustration



Elastic Net - A simple illustration

Method	MSE(SE)
Ridge	4.49 (0.46)
Lasso	3.06 (0.31)
Elastic Net	2.51 (0.29)

- MSE(standard error)

Beyond LASSO - Fused LASSO

Fused LASSO

- **Intuition**
  - Fused LASSO is designed for problems with features that can be ordered in some meaningful way, where "adjacent features" should have similar importance.
  - The fused LASSO penalizes the L_1-norm of both the coefficients and their successive differences.
- **Example**
  - Classification with fMRI data: each voxel has about 200 measurements over time. The coeefficients for adjacent voxels should be similar
- **Formulation**

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

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

Beyond LASSO - Fused LASSO

Fused LASSO

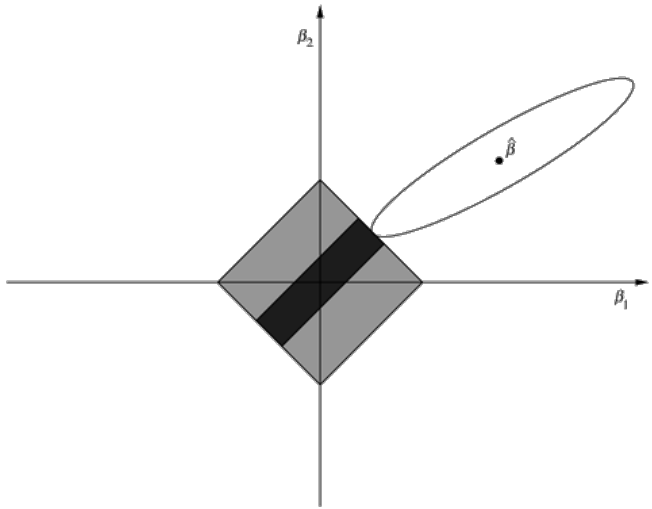
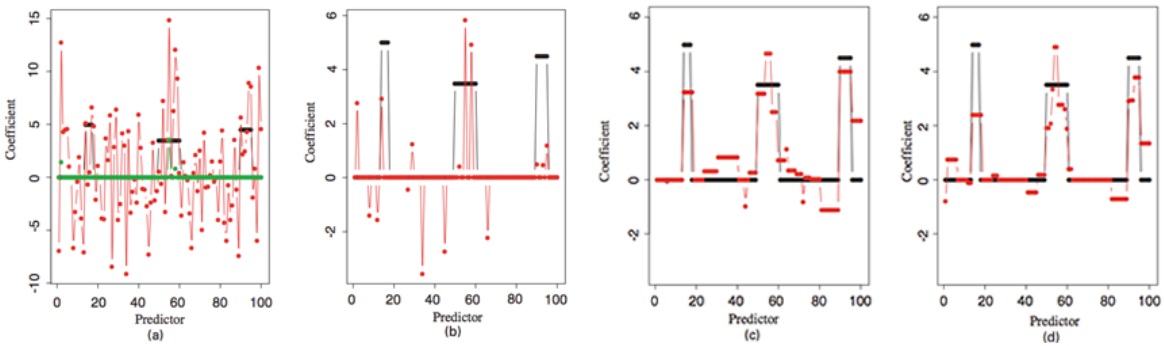


Fig. 2. Schematic diagram of the fused lasso, for the case $N > p = 2$: we seek the first time that the contours of the sum-of-squares loss function (⬭) satisfy $\Sigma_j |\beta_j| = s_1$ (◆) and $\Sigma_j |\beta_j - \beta_{j-1}| = s_2$ (◢)

Fused LASSO - Simulation results



- $p = 100$. Black lines are the true coefficients.
- (a) Univeriate regression coefficients (red), a soft threshold version of them (green)
- (b) LASSO solution (red), $s_1 = 35.6, \quad s_2 = \infty$
- (c) Fusion estimate, $s_1 = \infty, \quad s_2 = 26$
- (d) Fused LASSO, $s_1 = \sum |\beta_j|, \quad s_2 = \sum |\beta_j - \beta_{j-1}|$

Beyond LASSO - Group LASSO

Group LASSO

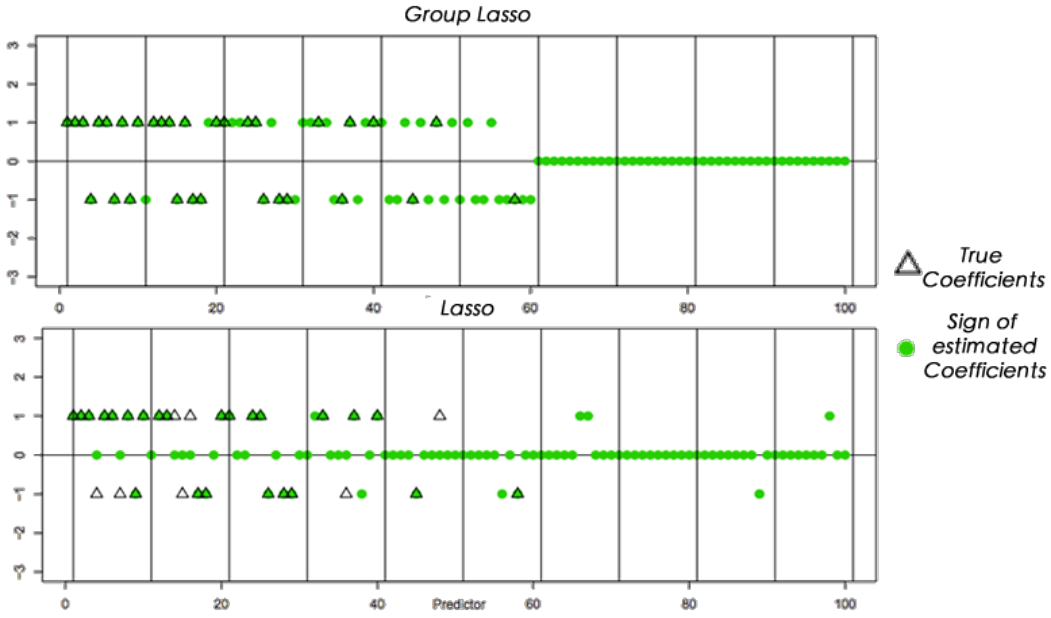
- **Intuition**
  - Features are divided into L groups
  - Features within the same group should share similar coefficients
- **Example**
  - Binary dummy variables from one single discrete variable, e.g. $stage\_cancer \in \{1, 2, 3\}$ can be translated into three binary dummy variables $(stage1, stage2,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

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

Group LASSO - Simulation Results

- Generate $n = 200$ observations with $p = 100$, divided into ten blocks equally
- The number of non-zero coefficients in blocks are
  - block 1: 10 out of 10
  - block 2: 8 out of 10
  - block 3: 6 out of 10
  - block 4: 4 out of 10
  - block 5: 2 out of 10
  - block 6-10: 0 out of 10
- The coefficients are either -1 or +1, with the sign being chosen randomly.
- The predictors are standard Gaussian with correlation 0.2 within a group and zero otherwise
- A Gaussian noise with standard deviation 4.0 was added to each observation

Group LASSO - Simulation Results



Beyond LASSO - l_1-l_p penalization

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
  - **Example 3**:learn to control a robot for driving in different landscapes
- **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.
  - different tasks share a subset of relevant features selected from a large common space of features.

Beyond LASSO - l_1-l_p penalization

l_1-l_p penalization

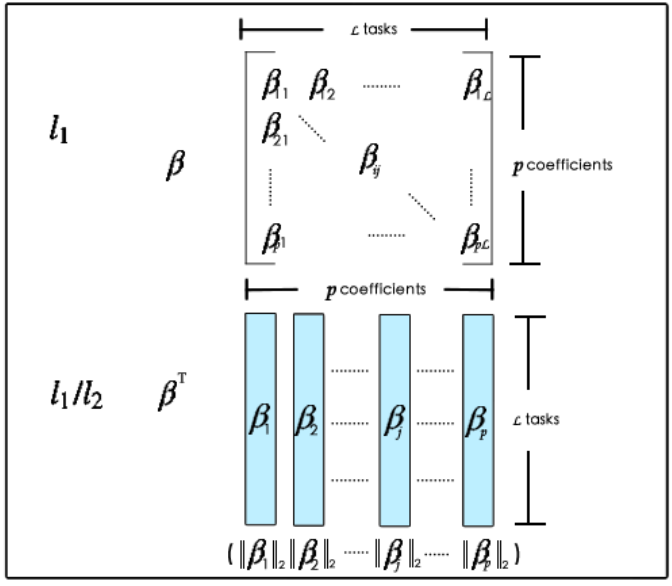
- **Formulation**
  - $\mathbf{X}_l$: $N \times p$ input matrix for task $l = 1..L$
    - L is the total number of tasks
  - β: $p \times L$ coefficient matrix
  - $\mathbf{y}$: $N \times L$ output matrix
  - objective function

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

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

Beyond LASSO - $l_1 - l_p$ penalization

$l_1 - l_p$ penalization -Coefficient matrix



$l_1 - l_p$ penalization - Experiment Result

- **Dataset**: handwritten words dataset collected by Rob Kassel
  - Contains writings from more than 180 different writers.
  - For each writer, the number of each letter we have is between 4 and 30
  - The letters are originally represented as 8×16
- **Task**: build binary classiers that discriminate between pairs of letters. Specically concentrat on the pairs of letters that are the most difficult to distinguish when written by hand.
- **Experiment**: learned classications of 9 pairs of letters for 40 different writers

The letter *a* written by 40 different people.

$l_1 - l_p$ penalization - Experiment Result

- **Candidate methods**
  - Pool l_1: a classifier is trained on all data regardless of writers
  - Independent l_1 regularization: For each writer, a classifier is trained
  - l_1/l_1-regularization:

$$obj = \sum_{l=1}^{L} J(\beta_{:l}, \mathbf{X}_l, \mathbf{y}_{:l}) + \lambda \sum_{l=1}^{L} \|\beta_{:l}\|_1$$

  - l_1/l_2-regularization:

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

$l_1 - l_p$ penalization - Experiment Result

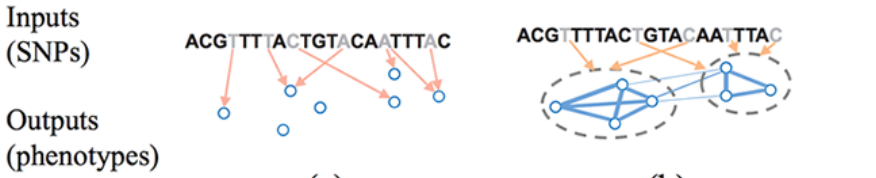
	strokes : error(%)				pixels: error (%)			
Task	ℓ_1/ℓ_2	ℓ_1/ℓ_1	sp.ℓ_1	pool	ℓ_1/ℓ_2	ℓ_1/ℓ_1	sp.ℓ_1	pool
c/e	**2.5**	3.0	3.3	3.0	**4.0**	8.5	9.0	4.5
	2.0	3.5	3.3	2.5	**3.5**	7.8	10.3	4.5
g/y	8.4	11.3	**8.1**	17.8	**11.4**	16.1	17.2	18.6
	10.3	10.3	**9.3**	16.9	11.6	**9.7**	10.9	21.4
g/s	3.3	3.8	**3.0**	10.7	**4.4**	10.0	10.3	6.9
	3.8	4.0	**2.5**	12.0	4.7	6.7	5.0	6.4
m/n	4.4	4.4	**3.6**	4.7	**2.5**	6.3	6.9	4.1
	4.1	5.8	**3.6**	5.3	**1.9**	2.8	4.1	
a/g	**1.4**	2.8	2.2	2.8	**1.3**	3.6	4.1	3.6
	0.8	1.6	1.3	2.5	**0.8**	1.7	1.4	3.9
i/j	**8.9**	9.5	9.5	11.5	12.0	14.0	14.0	**11.3**
	9.2	9.8	11.1	11.3	**10.3**	12.7	13.5	11.5
a/o	**2.0**	2.9	2.3	3.8	**2.8**	4.8	5.2	4.2
	2.7	2.7	**1.9**	4.3	**2.1**	3.1	3.5	4.2
f/t	**4.0**	5.0	6.0	8.1	**5.0**	6.7	6.1	8.2
	5.8	**4.1**	5.5	7.5	**6.4**	11.1	9.6	7.1
h/n	**0.9**	1.6	1.9	3.4	**3.2**	14.3	18.6	5.0
	0.9	0.6	**0.3**	3.7	**1.8**	3.6	5.0	5.0

- Within a cell, the first row contains results for feature selection, the second row uses random projections to obtain a common subspace (details omited, see paper: Multi-task feature selection)
- Bold: best of l_1/l_2, l_1/l_1, $sp.\,l_1$ or pooled l_1, Boxed : best of cell

Beyond LASSO - Graph-Guided Fused LASSO

Graph-Guided Fused LASSO (GFLASSO)

- **Example**



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

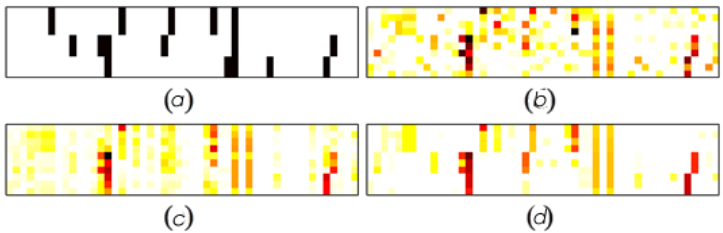
- **Formulation** Graph-Guided LASSO applies to multi-task settings

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

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 Fused LASSO

Graph-Guided Fused LASSO



- (a) The true regression coefficients
- (b) LASSO
- (c) l_1/l_2-regularized multi-task regression
- (d) GFLASSO

Summary

Outline

- **Introduction to Dimension Reduction**
- **Linear Regression and Least Squares (Review)**
- **Subset Selection**
- **Shrinkage Methods**
- **Beyond LASSO**

Summary

- Feature selection vs feature extraction
  - Feature selection: can save cost, be interpreted
  - Feature extraction: more general, often leads to better performance
- Linear models: Least Squares, Subset Selection, Ridge, LASSO:
  - LS is unbiased, but can have high variance (as includes all features)
  - Ridge (l_2 regularization): constrains parameter values, to reduce variance
  - Subset Selection, LASSO: finds subset of features (to reduce variance)
  - LASSO uses l_1 regularization
- LAR is like LASSO ((l_1 regularization), but
  - Their behaviors become different, when an coefficient hits zero
  - Modification: drops the feature, when its coefficient hits zero

Summary

- QP solves LASSO for a single λ, while LAR can solve LASSO for all λ
- Bayesian prior interpretation for Subset Selection, Ridge and LASSO
- Beyond LASSO (all use L1 regularization)
  - Elastic Net -- both L1 and L2
  - fused LASSO: coefficients of adjacent features are similar
  - group LASSO: feautures share similar coefficients within groups
  - l_1/l_2: similarities in multi-task
  - GFlasso: incorporates structure on output variables

More on the topics skipped here

- More on feature extraction methods:
  - http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf
  - Imola K. Fodor, A survey of dimension reduction techniques
  - Christopher J. C. Burges, Dimension Reduction: A Guided Tour
- Mutual-info-based feature selection:
  - Gavin Brown, Adam Pocock, Ming-Jie Zhao, Mikel Luján; Conditional Likelihood Maximisation: A Unifying Framework for Information Theoretic Feature Selection
  - Howard Hua Yang, John Moody. Feature Selection Based on Joint Mutual Information
  - Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy
- Beyond LASSO
  - http://webdocs.cs.ualberta.ca/~mahdavif/ReadingGroup/
- ELEN E6898 Sparse Signal Modeling
  - https://sites.google.com/site/eecs6898sparse2011/home

Sparse Models

Thank You!

Reference

- Trevor Hastie, Robert Tibshirani and Jerome Friedman. Elements of Statistical Learning [p7, p15, p16, p18, p19, p21-22, p26-27, p29-30, p33, p35-37, p42-p43, p50-p54, p56, p59]
- Temporal Sequence of FMRI scans (single slice): from http://www.midwest-medical.net/mri.sagittal.head.jpg [p8]
- Three Dimensional Image of Brain Activation from http://www.fmrib.ox.ac.uk/fmri_intro/brief.html [p8]
- http://en.wikipedia.org/wiki/Feature_selection [p10-12]
- http://en.wikipedia.org/wiki/Singular_value_decomposition [p27]
- http://en.wikipedia.org/wiki/Normal_distribution [p38]
- http://en.wikipedia.org/wiki/Laplacian_distribution [p38]
- http://webdocs.cs.ualberta.ca/~mahdavif/ReadingGroup/Papers/larS.pdf [p20]
- Bradley Efron, Trevor Hastie, Iain Johnstone and Robert Tibshirani. Least Angle Regression [p20]

Reference

- Kevin P. Murphy. Machine Learning A Probabilistic Perspective[p59]
- Prof.Schuurmans' notes on LASSO [p40]
- Conditional Likelihood Maximisation: A Unifying Framework for Information Theoretic Feature Selection [p8]
- Hui Zou and Trevor Hastie. Regularization and Variable Selection via the Elastic Net [p59-62]
- http://www.stanford.edu/~hastie/TALKS/enet_talk.pdf [p59-62]
- Robert Tibshirani and Michael Saunders, Sparsity and smoothness via the fused LASSO [P63-p65]
- Jerome Friedman Trevor Hastie and Robert Tibshirani. A note on the group LASSO and a sparse group LASSO [p66-68]
- Guillaume Obozinski, Ben Taskar, and Michael Jordan. Multi-task feature selection [p69-70, p72-p75]
- Xi Chen, Seyoung Kim, Qihang Lin, Jaime G. Carbonell, Eric P. Xing. Graph-Structured Multi-task Regression and an Efficient Optimization Method for General Fused LASSO [p76-77]