Linear Methods for Regression

Dept. Computer Science & Engineering,
Shanghai Jiao Tong University

Key Points in Previous Talk

- The objective of statistical learning is to identify the model with best generalization performance or with minimum training error?
- In what conditions, linear regression is the best as a classifier?
- KNN is one of implementations of the optimal decision function, but why we do not use it as a classifier in high dimensional space?
- Generalization error = Model Bias^2 + Variance

Model Complexity

Data Noise

Outline

- **The simple linear regression model**
- **Multiple linear regression**
- **Regularization**
  - Subset selection
  - Shrinkage
- **Principal component Regression**
- **Partial least squares Regression**

Objectives

- **How to use LR methods appropriately**
- **How to evaluate the performance of LR methods**
  - Confidence Interval
  - MSE / Generalization
- **How to improve the generalization performance**

Preliminaries

- Data $$(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)$$
  - x_i is the predictor (regressor, covariate, independent variable)
  - y_i is the response (dependent variable, outcome)

- We denote ***the regression function*** by

$$\eta(x) = E(Y \mid x)$$

- This is the conditional expectation of Y given x.
- The linear regression model assumes a specific linear form for

$$\eta(x) = \alpha + \beta x$$

which is usually thought of as an approximation to the truth.

Fitting by least squares

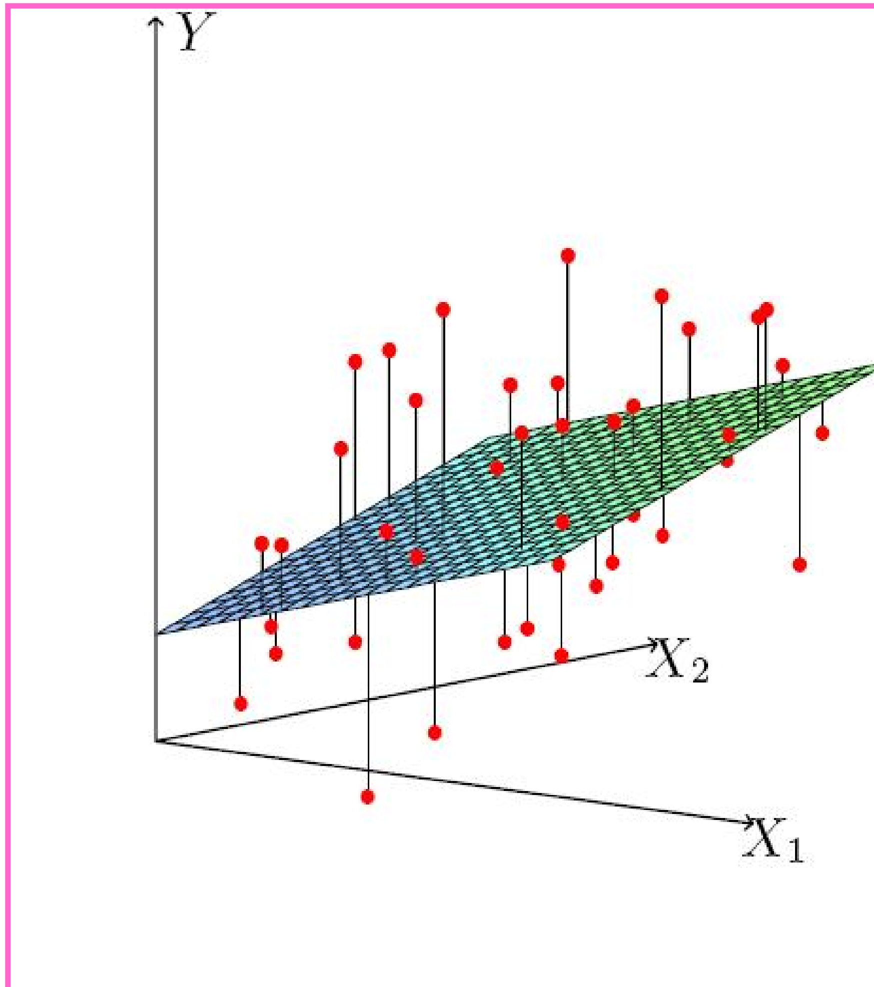
- Minimize: $\hat{\beta}_0, \hat{\beta} = \arg\min_{\beta_0,\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \beta x_i)^2$
- Solutions are

$$\hat{\beta} = \frac{\sum_{j=1}^{N} (x_i - \overline{x}) y_i}{\sum_{j=1}^{N} (x_i - \overline{x})^2}$$

$$\hat{\beta}_0 = \overline{y} - \hat{\beta}\overline{x}$$

- $\hat{y}_i = \hat{\beta}_0 + \hat{\beta} x_i$： the fitted or predicted values

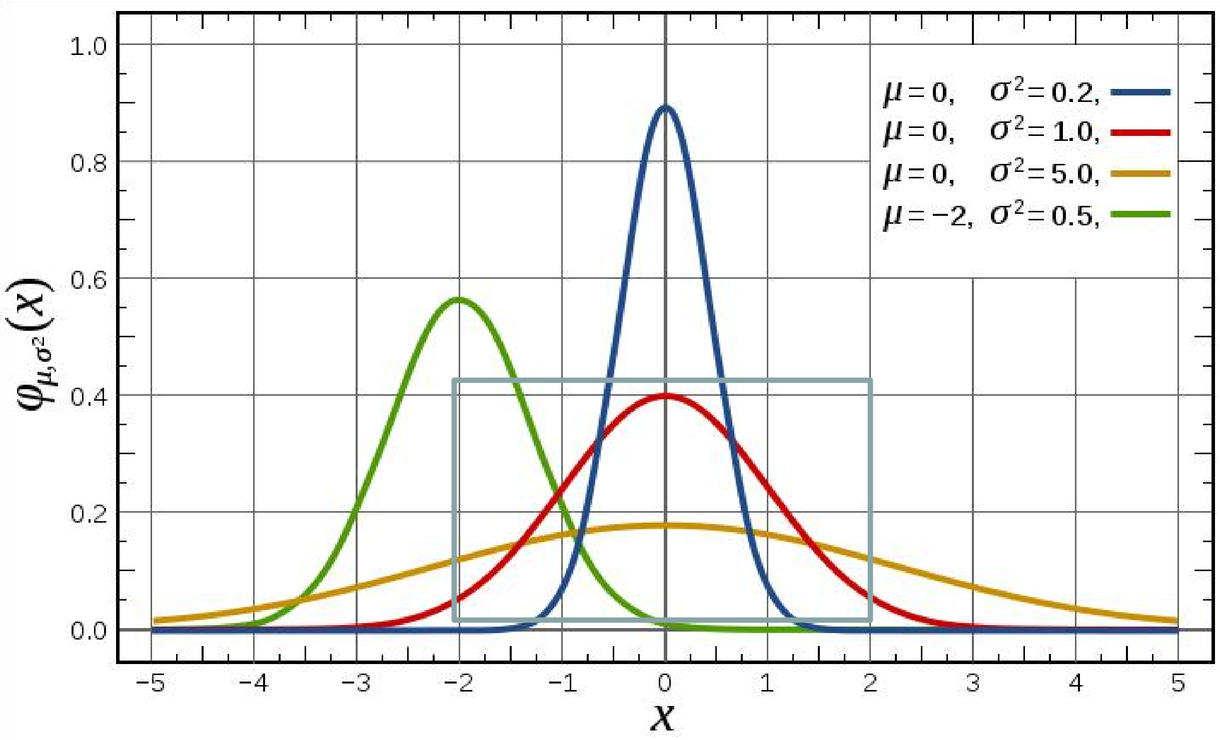
- $r_i = y_i - \hat{\beta}_0 - \hat{\beta}\, x_i$ are called the residuals



Gaussian Distribution

- **The normal distribution with mean *μ*, and variance *σ*²**.

$$f(x)=\frac{1}{\sqrt{2\pi}\,\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)=\frac{1}{\sigma}\phi\left(\frac{x-\mu}{\sigma}\right)$$



Standard errors & confidence intervals

- Assume further that $y_i = \beta_0 + \beta x_i + \varepsilon_i$

  where $E(\varepsilon_i) = 0$ and $Var(\varepsilon_i) = \sigma^2$ .Then

$$se(\hat{\beta}) = \left[\frac{\sigma^2}{\sum (x_i - \bar{x})^2} \right]^{1/2}$$

  Estimate σ^2 by $\hat{\sigma}^2 = \sum (y_i - \hat{y}_i)^2 / (N-2)$.

- Under additional assumption of normality for ε_is , a 95% confidence interval for β is:

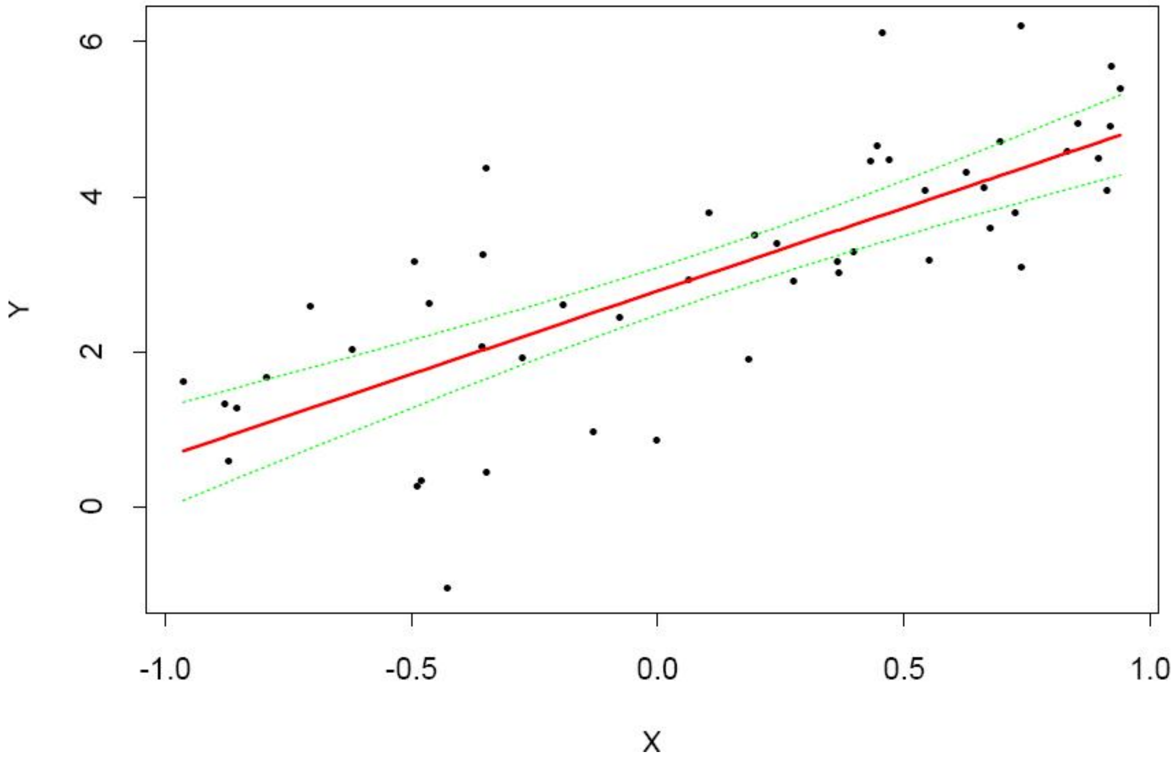
$$\hat{\beta} \pm 1.96 s\hat{e}(\hat{\beta}), \quad s\hat{e}(\hat{\beta}) = \left[\frac{\hat{\sigma}^2}{\sum (x_i - \bar{x})^2} \right]^{1/2}$$

Fitted Line and Standard Errors

- **Fitted regression line with pointwise standard errors:**

$$\hat{\eta}(x) = \hat{\beta}_0 + \hat{\beta}x$$
$$= \bar{y} + \hat{\beta}(x - \bar{x})$$

$$se[\hat{\eta}(x)] = \left[\mathrm{var}(\bar{y}) + \mathrm{var}(\hat{\beta})(x - \bar{x})^2\right]^{1/2}$$
$$= \left[\frac{\sigma^2}{n} + \frac{\sigma^2 (x - \bar{x})^2}{\sum (x_i - \bar{x})^2}\right]^{1/2}$$



$$\hat{\eta}(x) \pm 2\,\mathrm{se}\left|\hat{\eta}(x)\right|$$

Multiple linear regression

- Statistical Model $y = \beta_0 + \mathbf{x}^T \beta + \varepsilon$
- Model is $y_i = \beta_0 + \mathbf{x}_i^T \beta + \varepsilon_i, \quad i = 1, \cdots, \mathrm{N}$

Equivalently in matrix notation:

$$\mathbf{y} = \mathbf{X}\beta + \boldsymbol{\varepsilon}$$

- $\mathbf{y}$ is N-vector of predicted values
- $\mathbf{X}$ is N $\times$ p matrix of regresses, with ones in the first column
- β is a p-vector of parameters

Estimation by least squares

$$\hat{\beta} = \arg\min \sum_i (y_i - \beta_0 - \sum_{j=1}^{p-1} x_{ij}\beta_j)^2$$

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

Solution is $\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$

$$\hat{y} = \mathbf{X}\hat{\beta}$$

Also $Var(\hat{\beta}) = (\mathbf{X}^T\mathbf{X})^{-1}\sigma^2$

The Bias-variance tradeoff

- A good measure of the prediction performance for an estimator $\hat{f}(x)$ is the mean squared error. Let $f_0(x)$ be the true function.

$$\mathrm{MSE}[\hat{f}(x)] = \mathrm{E}[\hat{f}(x) - f_0(x)]^2$$

- This can be written as

$$\mathrm{MSE}[\hat{f}(x)] = \mathrm{Var}[\hat{f}(x)] + [E\hat{f}(x) - f_0(x)]^2$$

$$variance + \quad bias\text{^}2.$$

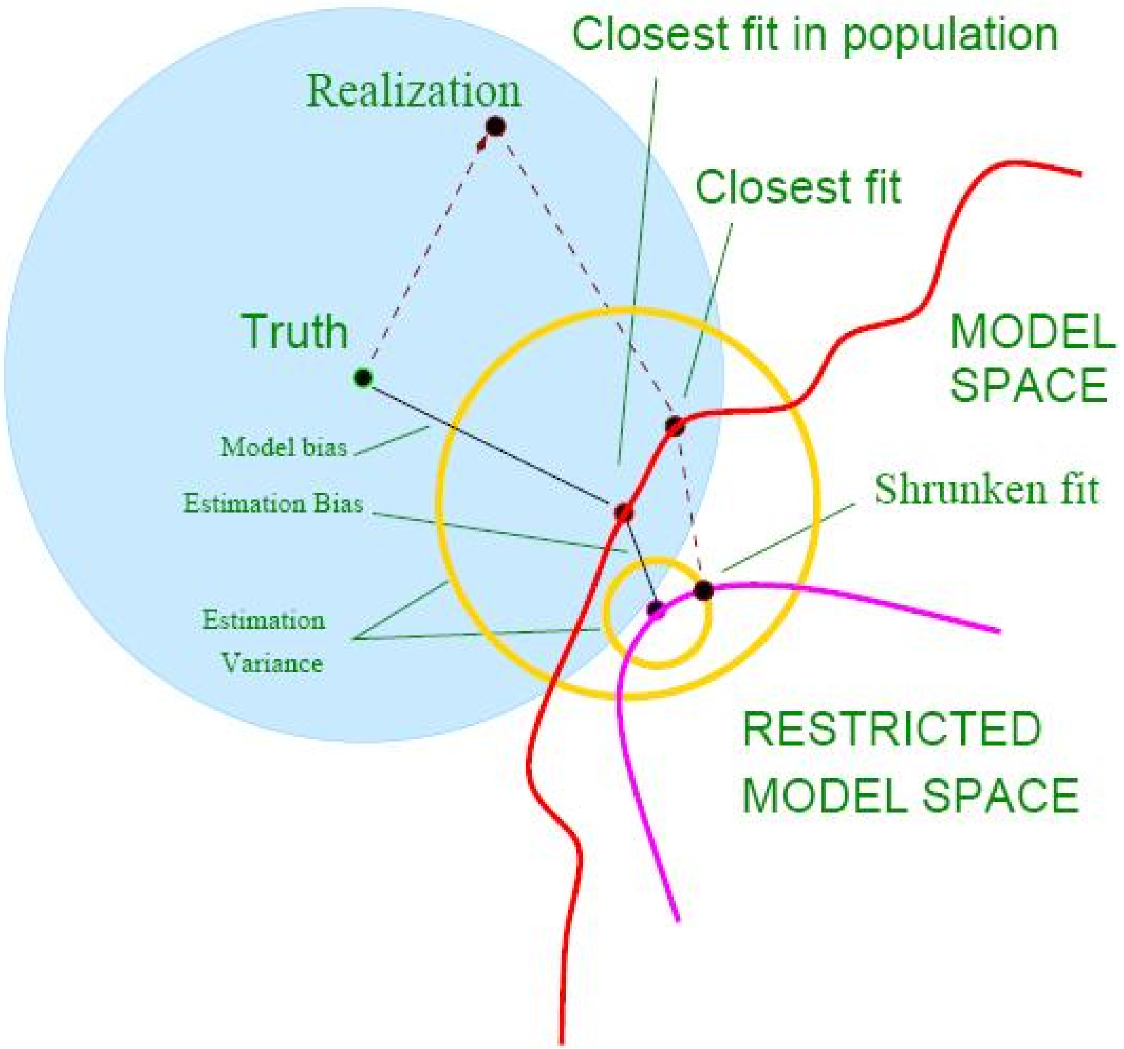
- When bias is low, variance is high and vice-versa.
  - Choose estimators ---- a tradeoff between bias and variance.

The Bias-variance tradeoff

- If the linear model is correct for a given problem, then the least squares prediction f is unbiased, and has the lowest variance among all unbiased estimators that are linear functions of y.
- Generally, by *regularization* (shrinking, dampening, controlling)
  - the estimator in some way, its variance will be reduced
  - if the corresponding increase in bias is small, this will be worthwhile.

Model Selection

- **Examples of regularization**: subset selection (forward, backward, all subsets); ridge regression, the lasso.
- In reality models are almost never correct, so there is an additional *model bias* between the closest member of the linear model class and the truth.



Question?

Assume that the true function

$$y = f(x), \quad x \in R^{10}.$$

If we use higher dimensional variables $x \in R^p, (p > 10)$ to approximate the function, can we achieve better generalization performance?

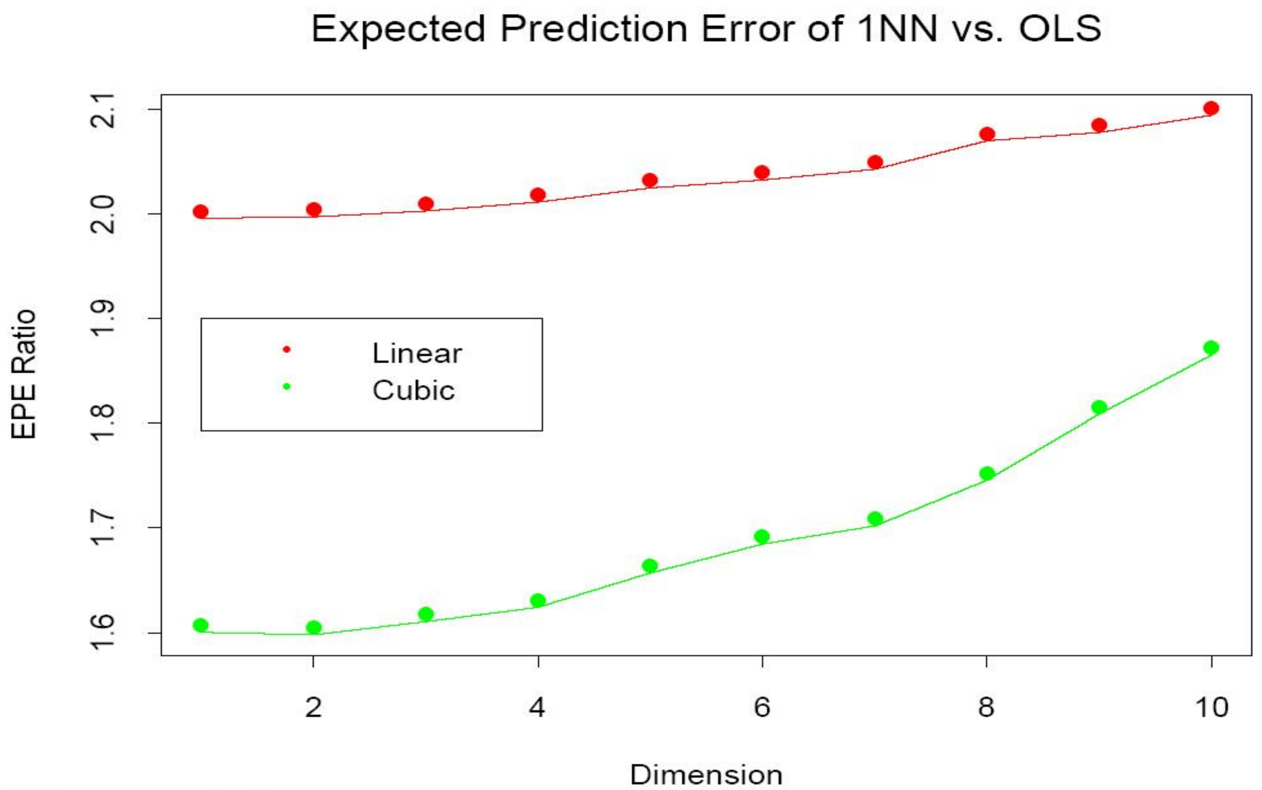


Figure 2.9: *The curves show the expected prediction error (at $x_0 = 0$) for 1-nearest neighbor relative to least squares for the model $Y = f(X) + \varepsilon$. For the red curve, $f(x) = x_1$, while for the green curve $f(x) = \frac{1}{2}(x_1 + 1)^3$.*

Variable subset selection

- **The first is prediction accuracy**
  - often have low bias, but large variance.
- **The second reason is interpretation.**
  - to determine a smaller subset that exhibit the strongest effects.

- **There are different strategies:**
  - **All subsets regression** is to find the subset of size s that gives smallest residual sum of squares.
  - The question of how to choose s involves the tradeoff between bias and variance: can use cross-validation

Hypothesis Test

Hypothesis testing: the main steps

Set null hypothesis

↓

Set study (alternative) hypothesis

↓

Carry out significance test — α -value

↓

Obtain test statistic — z - score

↓

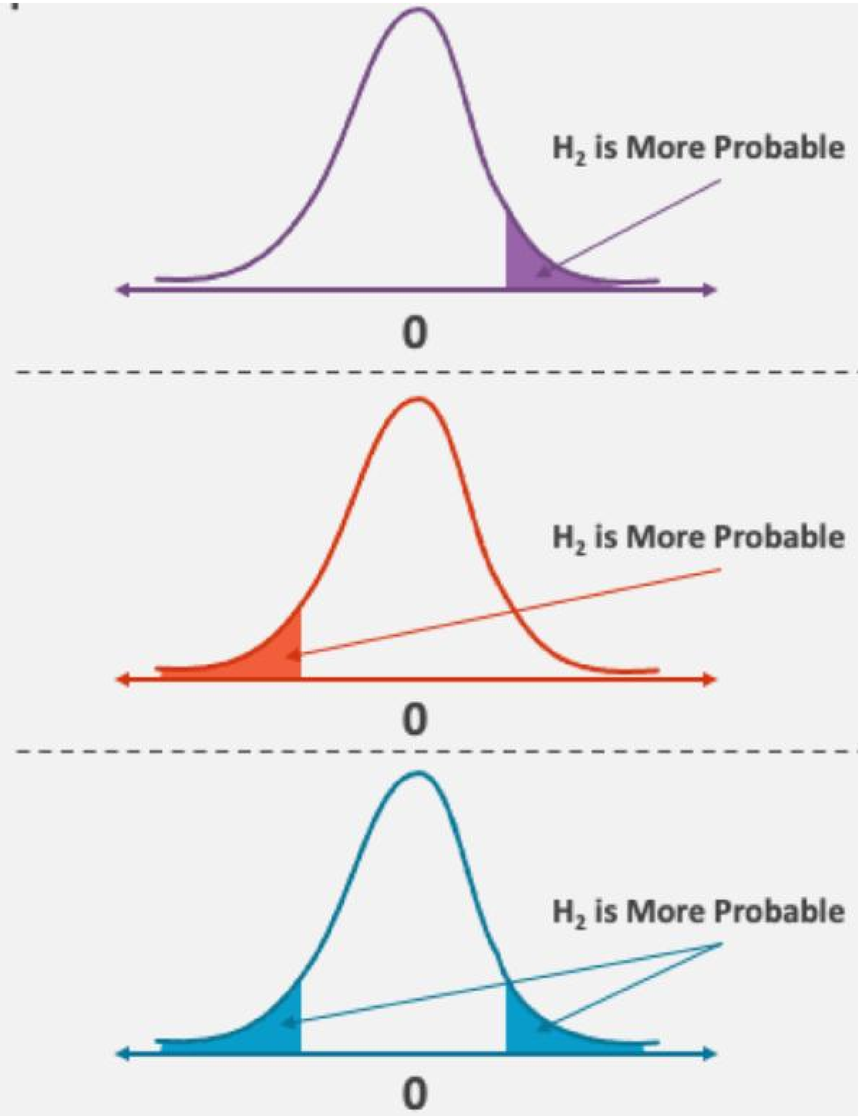
Compare test statistic to hypothesized critical value — z^{α} percentile

↓

Obtain p-value — p -value

↓

Make a decision



$$z^{0.05} = 1.96$$

Hypothesis Test

- **The linear regression model**

$$Y = \beta_0 + \sum_{j=1}^{p} X_j \beta_j + \varepsilon, \qquad \varepsilon \sim N(0, \sigma^2)$$

- The regression solutions

$$\hat{\beta} \sim N(\beta, (\mathbf{X}^T \mathbf{X})^{-1} \sigma^2), \qquad \hat{\sigma}^2 \sim \sigma^2 \chi^2_{N-p-1} / (N-p-1)$$

- To test the hypothesis that a particular coefficient $\beta_j = 0$, we take the standardized coefficient or *Z-score*

$$z_j = \hat{\beta}_j \Big/ \left(\hat{\sigma} \sqrt{v_j} \right)$$

where v_{j} is the j-th diagonal element of $(\mathbf{X}^T\mathbf{X})^{-1}$.

Hypothesis Test（example：Prostate Cancer)

- **Lcavol:** log cancer volume,
- **lweight:** log prostate weight,
- **age**,
- **Lbph:** log of the amount of benign prostatic hyperplasia(良性前列腺增生量)
- **Svi:** seminal vesicle invasion,
- **Lcp:** log of capsular penetration,
- **Gleason:** Gleason score,
- **pgg45:** percent of Gleason scores 4 or 5

to fit **lpsa**: the log of prostate-specific antigen(前列腺特异抗原)

- Training samples: 67
- Test samples: 30

TABLE 3.1. *Correlations of predictors in the prostate cancer data.*

	lcavol	lweight	age	lbph	svi	lcp	gleason
lweight	0.300						
age	0.286	0.317					
lbph	0.063	0.437	0.287				
svi	0.593	0.181	0.129	−0.139			
lcp	$\boxed{0.692}$	0.157	0.173	−0.089	0.671		
gleason	0.426	0.024	0.366	0.033	0.307	0.476	
pgg45	0.483	0.074	0.276	−0.030	0.481	0.663	0.757

Hypothesis Test（example：Prostate Cancer）

- Roughly a *Z-score* larger than **2** in absolute value is significantly nonzero at the p = 0.05 level.

- **Significant**
  - Lcavol; lweight
  - Lbph; svi

- **Non-significant**
  - Age; lcp
  - Gleason; pgg45

Term	Coefficient	Std. Error	Z Score
Intercept	2.46	0.09	27.60
lcavol	0.68	0.13	5.37
lweight	0.26	0.10	2.75
age	−0.14	0.10	−1.40
lbph	0.21	0.10	2.06
svi	0.31	0.12	2.47
lcp	−0.29	0.15	−1.87
gleason	−0.02	0.15	−0.15
pgg45	0.27	0.15	1.74

Variable subset selection

- Backward stepwise selection starts with the full OLS model, and sequentially deletes variables.
- There are also hybrid stepwise selection strategies which add in the best variable and delete the least important variable, in a sequential manner.
- Each procedure has one or more *tuning parameters*:
  - subset size
  - *P-values* for adding or dropping terms

Model Assessment

- **Objectives:**
    1. Choose a value of a tuning parameter for a model family
    2. Estimate the prediction performance of a given model

- For both of these purposes, the best approach is to run the procedure on an independent test set, if one is available

- If possible one should use different test data for (1) and (2) above: a *validation set* for (1) and a *test set* for (2)

- Often there is insufficient data to create a separate validation or test set. In this instance *Cross-Validation* is useful.

K-Fold Cross-Validation

- Primary method for estimating a tuning parameter (such as subset size)
- Divide the data into K roughly equal parts (typically K=5 or 10)

1	2	3	4	5
Train	Train	Test	Train	Train

K-Fold Cross-Validation

- For each k = 1, 2, . . .K, fit the model with parameter to the other K − 1 parts, giving $\hat{\beta}^{-k}(\lambda)$ and compute its error in predicting the *k*-th part:

$$E_k(\lambda) = \sum_{i \in kth\ part} (y_i - \mathbf{x}_i^T \hat{\beta}^{-k}(\lambda))^2$$

- This gives the cross-validation error

$$CV(\lambda) = \frac{1}{K} \sum_{k=1}^{K} E_k(\lambda)$$

- Model selection by

$$\min CV(\lambda)$$

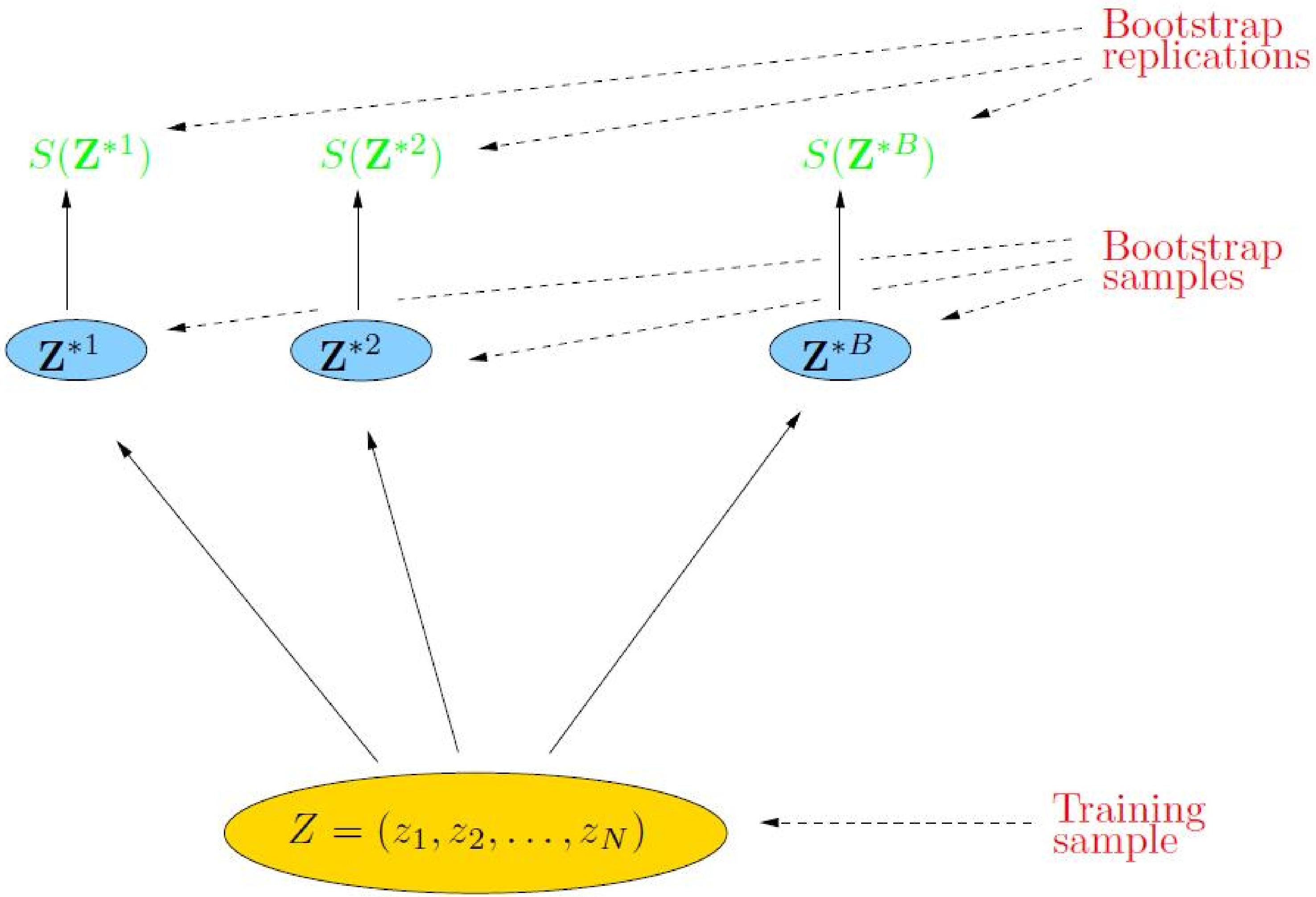
K-Fold Cross-Validation

- In our **variable subsets** example, λ is the subset size
- $\hat{\beta}^{-k}(\lambda)$ are the coefficients for the best subset of size , found from the training set that leaves out the k-th part of the data
- $E_k(\lambda)$ is the estimated test error for this best subset.

- Minimizing: $$CV(\lambda) = \frac{1}{K}\sum_{k=1}^{K} E_k(\lambda)$$

The Bootstrap approach

- Bootstrap works by sampling B times with replacement from training set to form a “bootstrap” data set.
- This process is repeated many times and the results are averaged. Bootstrap most useful for estimating standard errors of predictions.



Shrinkage Methods

- Ridge regression
- Lasso regression
- PCA regression
- Partial least squares

Shrinkage methods

Ridge regression

- Preprocessing：Data centering

$$x_{ij} <= x_{ij} - \bar{x}_j, \beta_0 = \bar{y} = \frac{1}{N}\sum_i y_i$$

- The ridge estimator is defined by

$$\hat{\beta}^{ridge} = \arg\min(\mathrm{Y} - \mathbf{X}\beta)^T (Y - \mathbf{X}\beta) + \lambda\beta^T\beta$$

Shrinkage Methods

Ridge regression

- The ridge estimator is defined by

$$\hat{\beta}^{ridge} = \arg\min(\mathrm{Y} - \mathbf{X}\beta)^T (Y - \mathbf{X}\beta) + \lambda\beta^T \beta$$

- Equivalently,

$$\hat{\beta}^{ridge} = \arg\min(\mathrm{Y} - \mathbf{X}\beta)^T (\mathrm{Y} - \mathbf{X}\beta)$$

$$\text{subject to } \sum \beta_j^2 \le s$$

Shrinkage methods

- The parameter $\lambda > 0$ penalizes β_j proportional to its size β_j^2.

  Solution:

$$\hat{\beta}_\lambda = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathrm{Y}$$

  where $\mathbf{I}$ is the identity matrix, $\lambda > 0$

- Note $\lambda = 0$ gives the least squares estimator; if $\lambda \to \infty$, then

$$\hat{\beta} \to 0$$

Ridge regression

- Ridge solution:

$$\hat{\beta}_\lambda = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathrm{Y}$$

- Singular value Decomposition:

$$\mathbf{X}{=}\mathbf{U}\mathbf{D}\mathbf{V}^T; \quad \mathbf{D} \text{ is a diagonal matrix with}$$

$$d_1 \ge d_2 \ge d_3 \ge \ldots \ge d_p \ge 0$$

- For ordinary Regression

$$\mathbf{X}\hat{\beta}^{\mathrm{ls}}{=}\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathrm{Y}{=}\mathbf{U}\mathbf{U}^T\mathrm{Y}$$

Ridge regression

- Ridge solution:

$$\hat{\beta}_\lambda = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathrm{Y}$$

- Singular value Decomposition:

$$\mathbf{X}{=}\mathbf{UDV}^T; \quad \mathbf{D} = \mathrm{diagonal}(d_1, d_2, ..., d_p)$$

- For Ridge Regression

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

$$= \mathrm{U}D\left(\mathbf{DD} + \lambda\mathbf{I}\right)^{-1} D\mathrm{U}^T\mathbf{Y} {=} \sum_{j=1}^{p} \mathbf{u}_j \frac{d_j^2}{d_j^2 + \lambda}\mathbf{u}_{\mathrm{j}}^T\mathbf{Y}$$

The Lasso

- The lasso (least absolute shrinkage and selection operator) is a shrinkage method like ridge, but acts in **a nonlinear manner** on the outcome y.

- The lasso is defined by

$$\hat{\beta}^{lasso} = \arg\min (\mathrm{Y} - \mathbf{X}\beta)^T (\mathrm{Y} - \mathbf{X}\beta)$$

$$\text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \le t$$

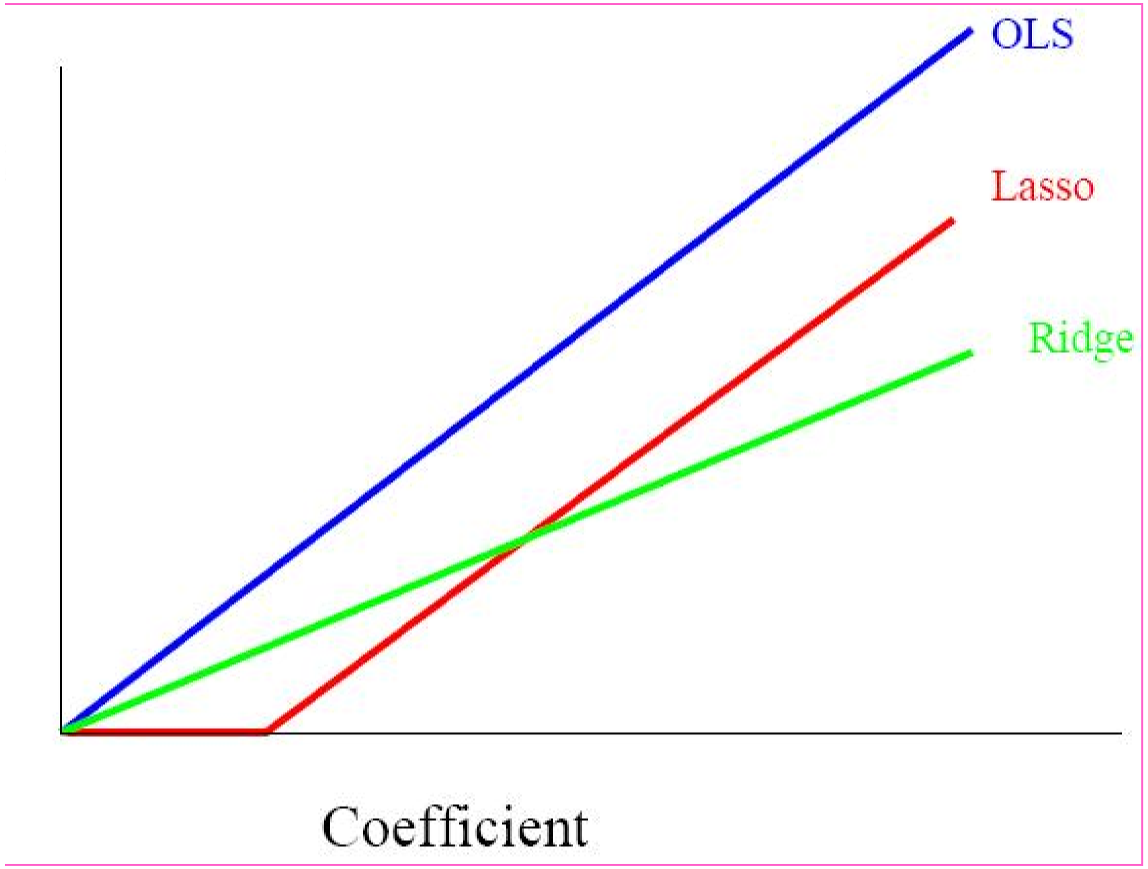
- No constraint on β_0

The Lasso

- Notice that ridge penalty $\sum \beta_j^2$ is replaced by $\sum |\beta_j|$
- This makes the solutions nonlinear in y, and a quadratic programming algorithm is used to compute them.
- Because of the nature of the constraint, if t is chosen small enough then the lasso will set some coefficients exactly to zero. Thus the lasso does a kind of continuous model selection.

The Lasso

- The parameter t should be adaptively chosen to minimize an estimate of expected, using say cross-validation
- ***Ridge vs Lasso:*** **if inputs are orthogonal,**
  - ridge *multiplies* least squares coefficients by a constant < 1,
  - lasso *translates* them towards zero by a constant, truncating at zero.



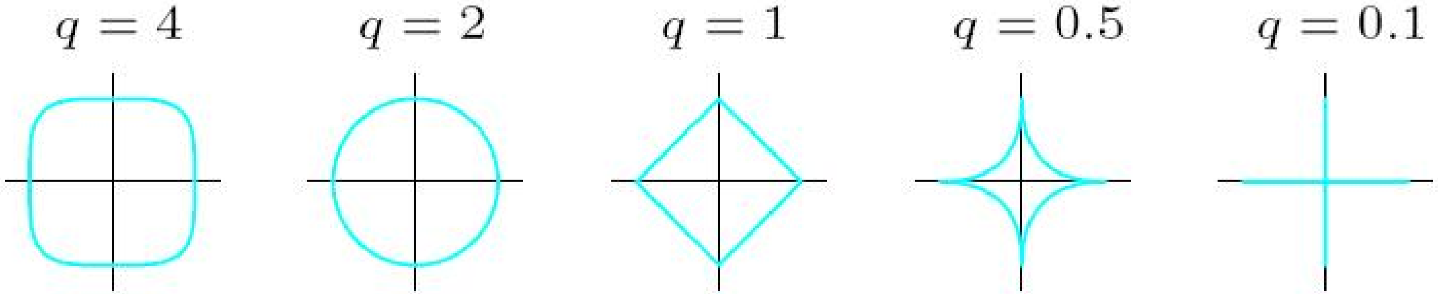
A family of shrinkage estimators

- Consider the criterion

$$\beta = \arg\min_{\beta} (\mathrm{Y} - \mathbf{X}\beta)^T (\mathrm{Y} - \mathbf{X}\beta)$$

$$\text{subject to} \quad \sum \left|\beta_j\right|^q \leq s$$

- for q >=0. The contours of constant value of $\sum_j \left|\beta_j\right|^q$ are shown for the case of two inputs.



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

Contents

Use of derived input directions

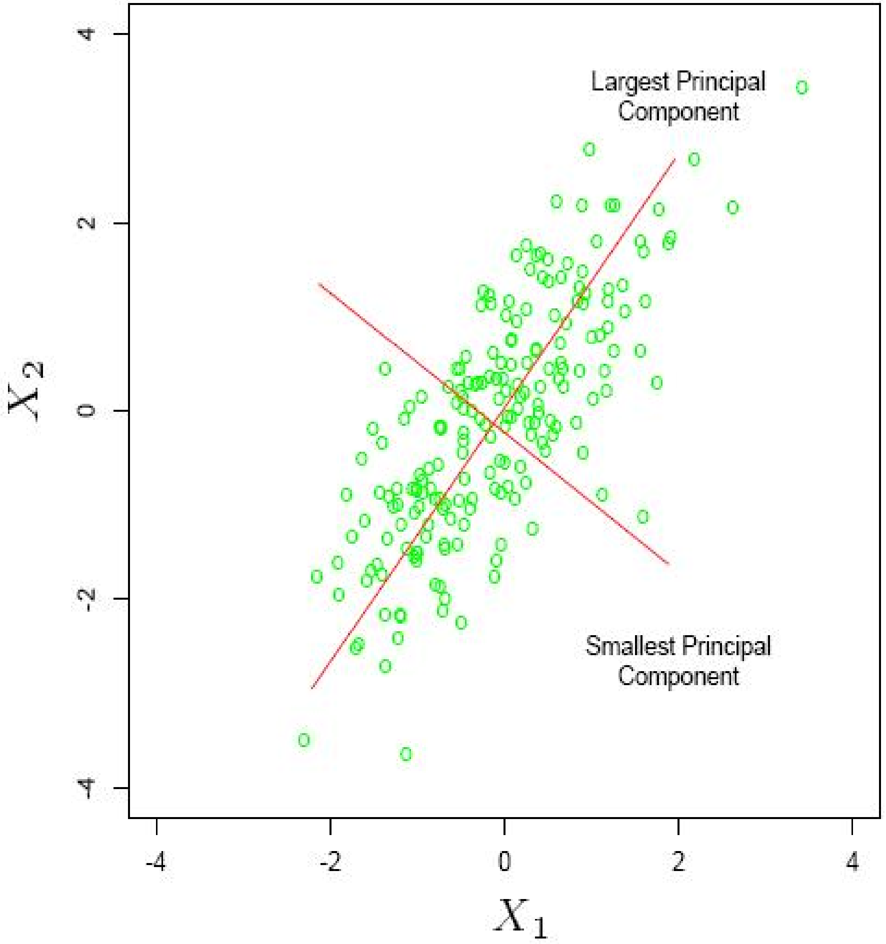
- **Principal components regression**
- Choose a set of linear combinations of the x_j s, and then regress the outcome on these linear combinations.
- **Principal components of the inputs**
  - Uncorrelated and ordered by decreasing variance.
- **If S is the sample covariance matrix of $x_1, x_2, \cdots, x_p$, then the** eigenvector equations $Sq_l = d_j^2 q_l$

  define the principal components of S.

Geometric Interpretation

- **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 y onto these components,** and then shrinks the coefficients of the low variance components more than the high-variance components.



PCA regression

- **Write $q(j)$ for the ordered principal components, ordered from largest to smallest value of d_j^2.**
- **Then principal components regression computes the derived input columns**

$$z_j = Xq(j)$$

and then regresses y on $z_1, z_2, \cdots, z_J$ for some J<=p.

PCA regression

- **Since the z_j s are orthogonal, this regression is just a sum of univariate regressions:**

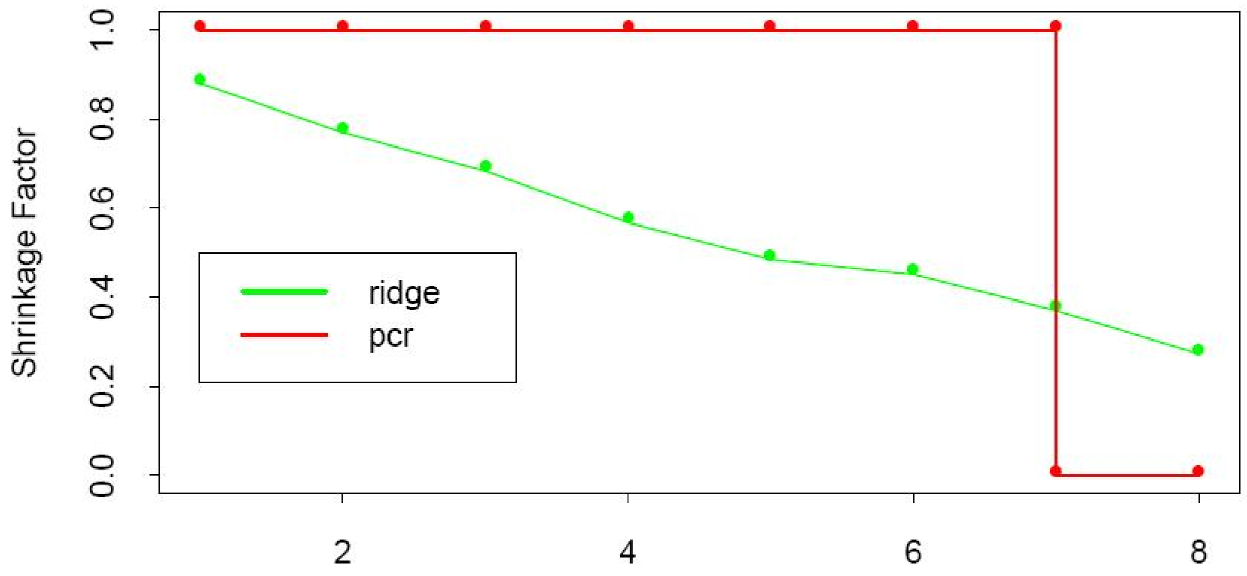
$$\hat{y}^{pcr} = \bar{y} + \sum_{j=1}^{J} \hat{\gamma}_j z_j$$

- **where $\hat{\gamma}_j$ is the univariate regression coefficient of y on z_j .**

- **Principal components regression is very similar to ridge regression:** both operate on the principal components of the input matrix.

PCA regression

- **Ridge regression shrinks the coefficients of the principal components**, with relatively more shrinkage applied to the smaller components than the larger
- **Principal components regression discards** the p-J+1 smallest eigenvalue components.



Partial least squares

- To construct a set of linear combinations of the x_j s for regression, but unlike principal components regression, it uses y (in addition to X) for this construction.
  - We assume that x is centered and begin by computing the univariate regression coefficient

$$\hat{\gamma}_j = \langle x_j, y \rangle$$

- **From this we construct the derived input**

$$z_1 = \sum \hat{\gamma}_j x_j$$

  - The first partial least squares direction.

Partial least squares

- From this we construct the derived input

$$z_1 = \sum \hat{\gamma}_j x_j$$

  – The first partial least squares direction.

- The outcome y is regressed on z_1 , giving coefficient

$$z_1 : r_1 = y - \hat{\beta}_1 z_1$$

- Orthogonalize $y, x_1, x_2, \ldots, x_p$ with respect to z_1

$$x_l^* = x_l - \hat{\theta}_l z_1$$

- Repeat the procedure

Algorithm 3.3 *Partial Least Squares.*

1. Standardize each $\mathbf{x}_j$ to have mean zero and variance one. Set $\hat{\mathbf{y}}^{(0)} = \bar{y}\mathbf{1}$, and $\mathbf{x}_j^{(0)} = \mathbf{x}_j,\ j = 1, \ldots, p$.

2. For $m = 1, 2, \ldots, p$

   (a) $\mathbf{z}_m = \sum_{j=1}^{p} \hat{\varphi}_{mj}\mathbf{x}_j^{(m-1)}$, where $\hat{\varphi}_{mj} = \langle \mathbf{x}_j^{(m-1)}, \mathbf{y}\rangle$.

   (b) $\hat{\theta}_m = \langle \mathbf{z}_m, \mathbf{y}\rangle / \langle \mathbf{z}_m, \mathbf{z}_m\rangle$.

   (c) $\hat{\mathbf{y}}^{(m)} = \hat{\mathbf{y}}^{(m-1)} + \hat{\theta}_m \mathbf{z}_m$.

   (d) Orthogonalize each $\mathbf{x}_j^{(m-1)}$ with respect to $\mathbf{z}_m$: $\mathbf{x}_j^{(m)} = \mathbf{x}_j^{(m-1)} - [\langle \mathbf{z}_m, \mathbf{x}_j^{(m-1)}\rangle / \langle \mathbf{z}_m, \mathbf{z}_m\rangle]\mathbf{z}_m,\ j = 1, 2, \ldots, p$.

3. Output the sequence of fitted vectors $\{\hat{\mathbf{y}}^{(m)}\}_1^p$. Since the $\{\mathbf{z}_\ell\}_1^m$ are linear in the original $\mathbf{x}_j$, so is $\hat{\mathbf{y}}^{(m)} = \mathbf{X}\hat{\beta}^{\text{pls}}(m)$. These linear coefficients can be recovered from the sequence of PLS transformations.

Ridge vs PCR vs PLS vs Lasso

- Recent study has shown that ridge and PCR outperform PLS in prediction, and they are simpler to understand.
- Lasso outperforms ridge when there are a moderate number of sizable effects, rather than many small effects. It also produces more interpretable models.
- These are still topics for ongoing research.

Summary

- How to use LR methods appropriately.
- How to evaluate the performance of LR methods
  - Confidence Interval
  - MSE / Generalization
- How to improve the generalization performance
  - Data: Feature selection (based on p-value)；Cross-validation
  - Shrinkage: Ridge; Lasso; PC regression; Partial least squares

- What is the purpose for imposing constraints on models?

The End of Talk