Linear Model Selection and Regularization

Chapter 6

March 6, 2018

1. Subset selection
2. Shrinkage methods
3. Dimension reduction methods (using derived inputs)
4. High dimension data analysis

Summary

So far we have covered

- Linear Models
  - Chap 3. regression
  - Chap 4. classification
  - Chap 5. model assessment (CV/Bootstrap)
- Now Chap 6. **model selection and regularization**
- Nonlinear models
  - Chap 7. generalized additive models (splines etc.)
  - Chap 8. trees (boosting, random forests)
  - Chap 9. kernel method (support vector machines)
  - Chap *. neural networks (deep learning)
- Chap 10. Topics in unsupervised learning

Summary

So far we have covered

- Linear Models
  - Chap 3. regression
  - Chap 4. classification
  - Chap 5. model assessment (CV/Bootstrap)
- Now Chap 6. **model selection and regularization**
- Nonlinear models
  - Chap 7. generalized additive models (splines etc.)
  - Chap 8. trees (boosting, random forests)
  - Chap 9. kernel method (support vector machines)
  - Chap *. neural networks (deep learning)
- Chap 10. Topics in unsupervised learning

Summary

So far we have covered

- Linear Models
  - Chap 3. regression
  - Chap 4. classification
  - Chap 5. model assessment (CV/Bootstrap)
- Now Chap 6. **model selection and regularization**
- Nonlinear models
  - Chap 7. generalized additive models (splines etc.)
  - Chap 8. trees (boosting, random forests)
  - Chap 9. kernel method (support vector machines)
  - Chap *. neural networks (deep learning)
- Chap 10. Topics in unsupervised learning

Interpretability vs. Prediction

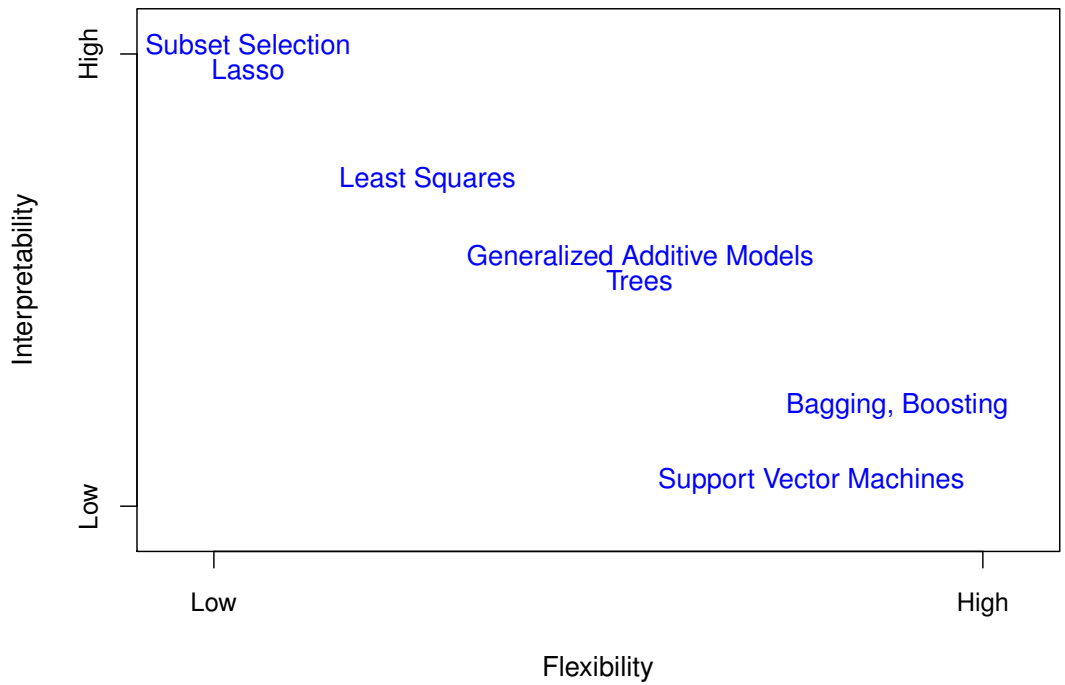

Figure: 2.7. As models become flexible, interpretability drops. **Occam Razor principle**: Everything has to be kept as simple as possible, but not simpler (Albert Einstein).

About this chapter

- Linear model already addressed in detail in Chapter 3.

$$Y = \beta_0 + \beta_1 X_1 + ... + \beta_p X_p + \epsilon$$

- Model assessment: cross-validation (prediction) error in Chapter 5.
- This chapter is about model selection for linear models.
- The model selection techniques can be extended beyond linear models.
- Details about AIC, BIC, Mallow's C_p mentioned in Chapter 3.

Feature/variable selection

- Not all existing input variables are useful for predicting the output.
- Keeping redundant inputs in model can lead to poor prediction and poor interpretation.
- We consider three ways of variable/model selection:
  1. Subset selection.
  2. Shrinkage/regularization: constraining some regression parameters to 0.
  3. Dimension reduction: (actually using the "derived inputs" by, for example, principle component approach.)

Best subset selection

- Exhaust all possible combinations of inputs.
- With p variables, there are 2^p many distinct combinations.
- Identify the best model among these models.

The algorithm of best subset selection

- Step 1. Let $\mathcal{M}_0$ be the *null model*, $Y = \beta_0 + \epsilon$. The predictor is the sample mean of response.
- Step 2. For $k = 1, 2, ..., p$,
  Fit all $\binom{p}{k} = p!/(k!(n-k)!)]$ models that contain exactly k predictors.
  Pick the best model, that with largest R^2, among them and call it $\mathcal{M}_k$.
- Step 3. Select a single best model from $\mathcal{M}_0, ..., \mathcal{M}_p$ by cross validation or AIC or BIC or C_p or adjusted R^2.

Comments

- Step 2 reduces to $p + 1$ models, using training error.
- Step 3 identifies the best model using predition error.
- Why using R^2 for step 2: they are of same complexity; the RSS is easy to compute.
- Cannot use training error in Step 3.

Recall: Definitions

- Residue

$$\hat{\epsilon}_i = y_i - \hat{\beta}_0 - \sum_{j=1}^{p} \hat{\beta}_j x_{ij}$$

- Residual Sum of Squares

$$\text{RSS} = \sum_{i=1}^{n} \hat{\epsilon}_i^2 = \sum_{i=1}^{n} (y_i - \hat{\beta}_0 - \sum_{j=1}^{p} \hat{\beta}_j x_{ij})^2$$

- R-squared

$$R^2 = \frac{\text{SS}_{reg}}{\text{SS}_{total}} = 1 - \frac{\text{SS}_{error}}{\text{SS}_{total}}$$

where $\text{SS}_{error} = \text{RSS} = \sum_{i=1}^{n} \hat{\epsilon}_i^2$ and $\text{SS}_{total} = \sum_{i=1}^{n} (y_i - \bar{y})^2$.

Example: Credit data

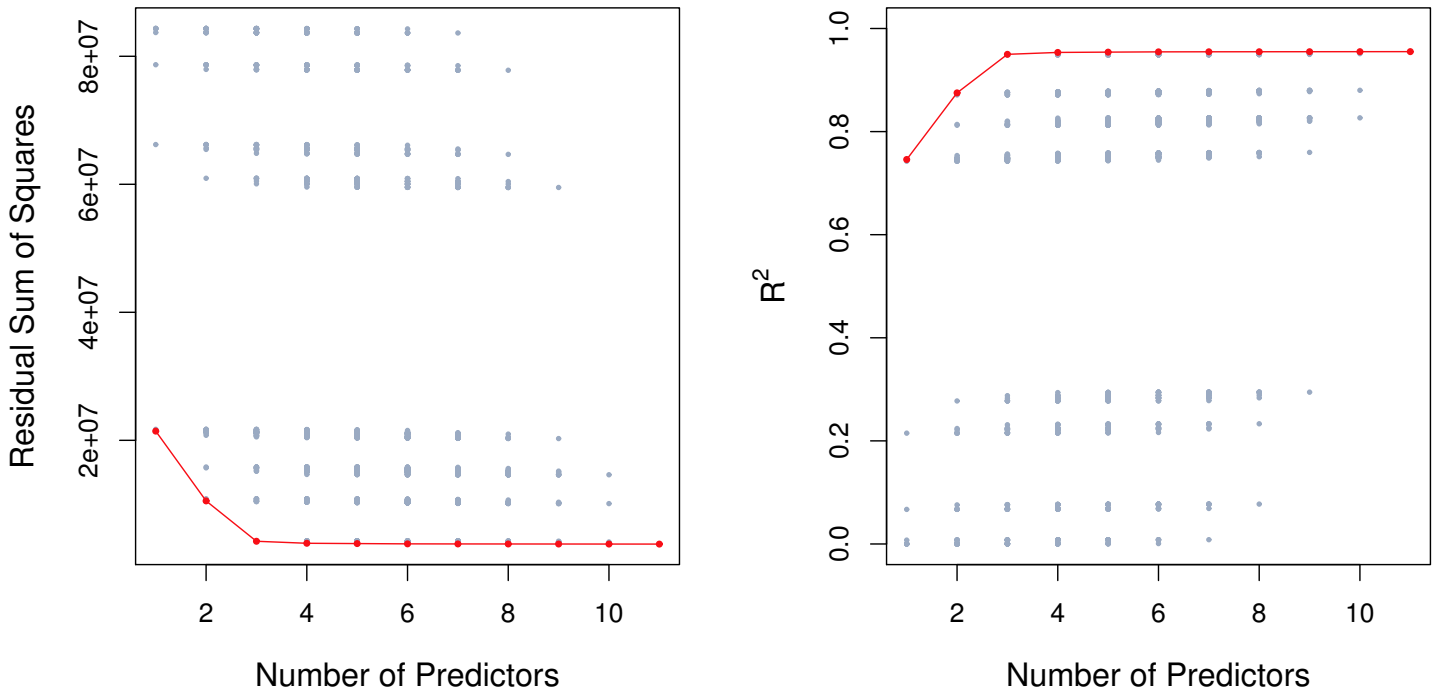

Figure: 6.1. For each possible model containing a subset of the ten predictors in the Credit data set, the RSS and R^2 are displayed. The red frontier tracks the best model for a given number of predictors, according to RSS and R^2. Though the data set contains only ten predictors, the x-axis ranges from 1 to 11, since one of the variables is categorical and takes on three values, leading to the creation of two dummy variables.

The issues of R-squared

- The R-squared is the percentage of the total variation in response due to the inputs.
- The R-squared reflects the *training error*.
- However, a model with larger R-squared is not necessarily better than another model with smaller R-squared when we consider *test error*!
- If model A has all the inputs of model B, then model A's R-squared will always be greater than or as large as that of model B.
- If model A's additional inputs are entirely uncorrelated with the response, model A contain more noise than model B. As a result, the prediction based on model A would inevitably be poorer or no better.

a). Adjusted R-squared.

- The adjusted R-squared, taking into account of the degrees of freedom, is defined as

$$\begin{aligned}
\text{adjusted } R^2 &= 1 - \frac{\text{MS}_{error}}{\text{MS}_{total}} \\
&= 1 - \frac{\text{SS}_{error}/(n-p-1)}{\text{SS}_{total}/(n-1)} \\
&= 1 - \frac{s^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2/(n-1)}
\end{aligned}$$

  With more inputs, the R^2 always increase, but the adjusted R^2 could decrease since more irrelevant inputs are penalized by the smaller degree of freedom of the residuals.

- The adjusted R-squared is preferred over the R-squared in evaluating models.

b). Mallows' C_p.

Recall that our linear model (2.1) has p covariates, and $s_p^2 = \hat{\sigma}^2 = \text{SS}_{error}/(n-p-1)$ is the unbiased estimator of σ^2.
Suppose we use only d of the p covariates with $d \leq p$.
The statistic of Mallow's C_p is defined as

$$\frac{\text{SS}_{error}(d)}{s_p^2} - 2d - n, \quad \text{or} \quad \frac{1}{n}\left(\text{SS}_{error}(d) + 2ds_p^2\right).$$

where $\text{SS}_{error}(d)$ is the residual sum of squares for the linear model with d inputs.
Mallows' C_p is an unbiased estimate of test MSE: the smaller it is, the better the model is.

c). AIC.

AIC stands for Akaike information criterion, which aims at maximizing the predictive likelihood and is defined as

$$\text{AIC} = \frac{1}{ns_p^2}\left(\text{SS}_{error}(d) + 2ds_p^2\right),$$

when Gaussian likelihood is assumed in least square regression. The model with the smallest AIC is preferred.

d). BIC.

- BIC stands for Schwarz's Bayesian information criterion, which is defined as

$$\text{BIC} = \frac{1}{ns_p^2}\left(\text{SS}_{error}(d) + ds_p^2(\log n)\right),$$

for a linear model with p inputs. Again, the model with the smallest BIC is perfered. The derivation of BIC results from Bayesian statistics and has Bayesian interpretation. It replaces $2ds_p^2$ in AIC by $(\log n)ds_p^2$, so for $\log n > 2$ or $n > 7$, BIC penalizes more heavily the models with more number of inputs.

Example

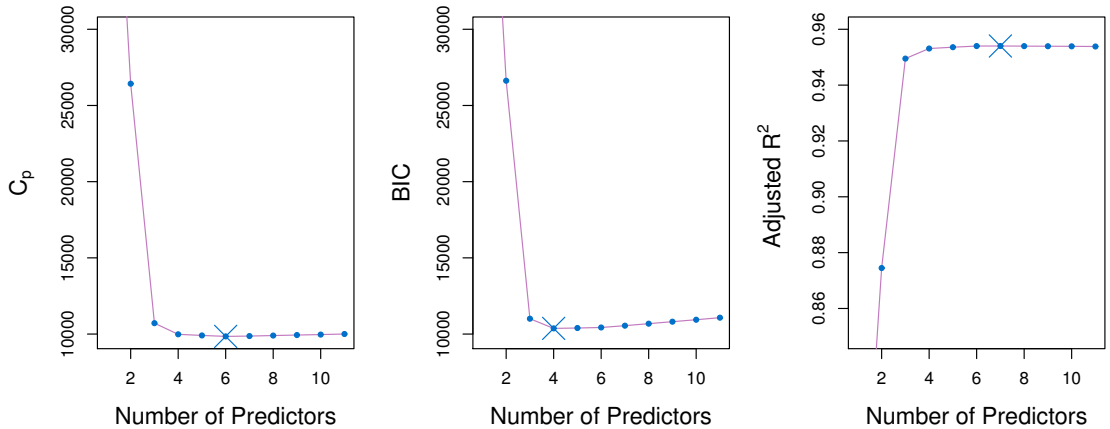

Figure: 6.2. C_p, BIC, and adjusted R^2 are shown for the best models of each size for the Credit data set (the lower frontier in Figure 6.1). C_p and BIC are estimates of test MSE. In the middle plot we see that the BIC estimate of test error shows an increase after four variables are selected. The other two plots are rather flat after four variables are included.

Pros and Cons of best subset selection

- Seems straightforward to carry out.
- Conceptually clear.
-
- The search space too large (2^p models), may lead to overfit.
- Computationally infeasible: too many models to run.
- if $p = 20$, there are $2^{20} > 1000,000$ models.

Forward stepwise selection

- Start with the null model.
- Find the best one-variable model.
- With the best one-varialbe model, add one more variable to get the best two-variable model.
- With the best two-varialbe model, add one more variable to get the best three-variable model.
- ....
- Find the best among all these best k-variable models.

The algorithm of forward stepwise selection

- Step 1. Let $\mathcal{M}_0$ be the null model, $Y = \beta_0 + \epsilon$. The predictor is the sample mean of response.
- Step 2. For $k = 0, 1, ..., p-1$,
  Consider all $p - k$ models that augment the predictors in $\mathcal{M}_k$ with one additional predictor.
  Choose the best among these $p - k$ models, and call it $\mathcal{M}_{k+1}$.
  Here best is defined as having smallest RSS or highest R^2.
- step 3. Select a single best model from $\mathcal{M}_0, ..., \mathcal{M}_p$ by cross validation or AIC or BIC or C_p or adjusted R^2.

Pros and Cons of forward stepwise selection

- Less computation
- Less models ($\sum_{k=0}^{p-1}(p-k) = 1 + p(p+1)/2$ models).
- (if $p = 20$, only 211 models, compared with more than 1 million models for best subset selection).
- No problem for first n-steps if $p > n$.
- Once an input is in, it does not get out.

Example: credit dataset

Variables	Best subset	Forward stepwise
one	rating	rating
two	rating, income	rating, income
three	rating, income, student	rating, income, student
four	cards, income, student, limit	rating, income, student, limit

TABLE 6.1. The first four selected models for best subset selection and forward stepwise selection on the Credit data set. The first three models are identical but the fourth models differ.

Backward stepwise selection

- Start with the largest model (all p inputs in).
- Find the best $(p-1)$-variable model, by reducing one from the largest model
- Find the best $(p-2)$-variable model, by reducing one variable from the best $(p-1)$-variable model.
- Find the best $(p-3)$-variable model, by reducing one variable from the best $(p-2)$-variable model.
- ....
- Find the best 1-varible model, by reducing one variable from the best 2-variable model.
- The null model.

The algorithm of backward stepwise selection

- Step 1. Let $\mathcal{M}_p$ be the full model.
- Step 2. For $k = p, p-1, ..., 1$,
  Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$ for a total of $k-1$ predictors
  Choose the best among these k models, and call it $\mathcal{M}_{k-1}$. Here best is defined as having smallest RSS or highest R^2.
- Step 3. Select a single best model from $\mathcal{M}_0, ..., \mathcal{M}_p$ by cross validation or AIC or BIC or C_p or adjusted R^2.

Pros and Cons of backward stepwise selection

- Less computation
- Less models ($\sum_{k=0}^{p-1}(p-k) = 1 + p(p+1)/2$ models).
- (if $p = 20$, only 211 models, compared with more than 1 million models for best subset selection).
- Once an input is out, it does not get in.
- No applicable to the case with $p > n$.

Find the best model based on prediction error.

- Validation/cross-validation approach (addressed in Chapter 5).
- Use Adjusted R^2, AIC, BIC or C_p.

Example

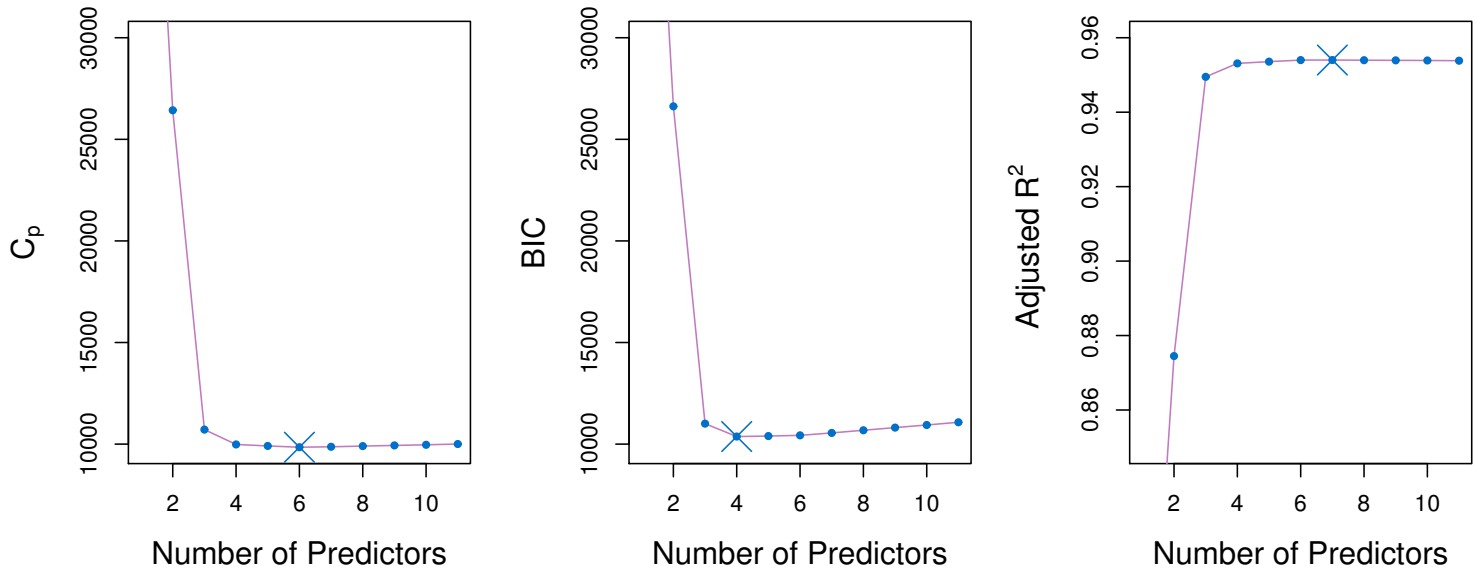

Figure: 6.2. C_p, BIC, and adjusted R^2 are shown for the best models of each size for the Credit data set (the lower frontier in Figure 6.1). C_p and BIC are estimates of test MSE. In the middle plot we see that the BIC estimate of test error shows an increase after four variables are selected. The other two plots are rather flat after four variables are included.

Example

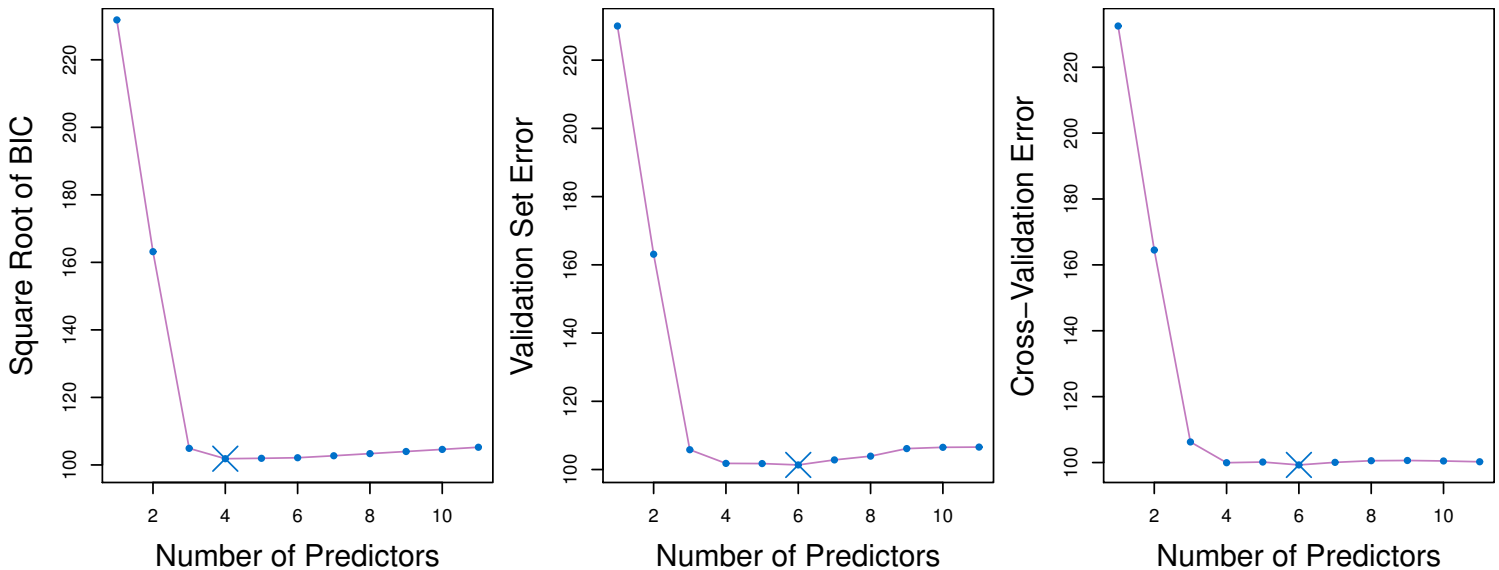

Figure: 6.3. For the Credit data set, three quantities are displayed for the best model containing d predictors, for d ranging from 1 to 11. The overall best model, based on each of these quantities, is shown as a blue cross. Left: Square root of BIC. Center: Validation set errors (75% training data). Right: 10-fold Cross-validation errors.

The one standard deviation rule

- In the above figure, model with 6 inputs do not seem to be much better than model with 4 or 3 inputs.
- Keep in mind the Occam's razor: Choose the simplest model if they are similar by other criterion.

The one standard deviation rule

- Calculate the standard error of the estimated test MSE for each model size,
- Consider the models with estimated test MSE of one standard deviation within the smallest test MSE.
- Among them select the one with the smallest model size.
- (Apply this rule to the Example in Figure 6.3 gives the model with 3 variable.)

Ridge Regression

- The least squares estimator $\hat{\beta}$ is minimizing

$$\text{RSS} = \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2$$

- The ridge regression $\hat{\beta}_\lambda^R$ is minimizing

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\beta_j^2$$

  where $\lambda \geq 0$ is a tuning parameter.
- The first term measuares goodness of fit, the smaller the better.
- The second term $\lambda\sum_{j=1}^{p}\beta_j^2$ is called shrikage penalty, which *shrinks* β_j towards 0.
- The shrinkage reduces variance (at the cost increased bias)!

Tuning parameter λ.

- $\lambda = 0$: no penalty, $\hat{\beta}_0^R = \hat{\beta}$.
- $\lambda = \infty$: infinity penalty, $\hat{\beta}_0^R = 0$.
- Large λ: heavy penalty, more shrinkage of the estimator.
- Note that β_0 is not penalized.

Stanardize the inputs.

- For j-th input X_j with obsevations: $(x_{1j}, ..., x_{nj})$, standarize it as

$$\tilde{x}_{ij} = \frac{x_{ij} - \bar{x}_j}{\sqrt{(1/n)\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2}}$$

  to get rid of the scale of X_j.
- Suggest to apply standardization before trying ridge regression.
- Least squares is unaffected by the scale of X_j. (i.e., $X_j\hat{\beta}_j = (cX_j)(\hat{\beta}_j/c)$)
- Ridge is affected by λ as well as the scale of the inputs.

Example

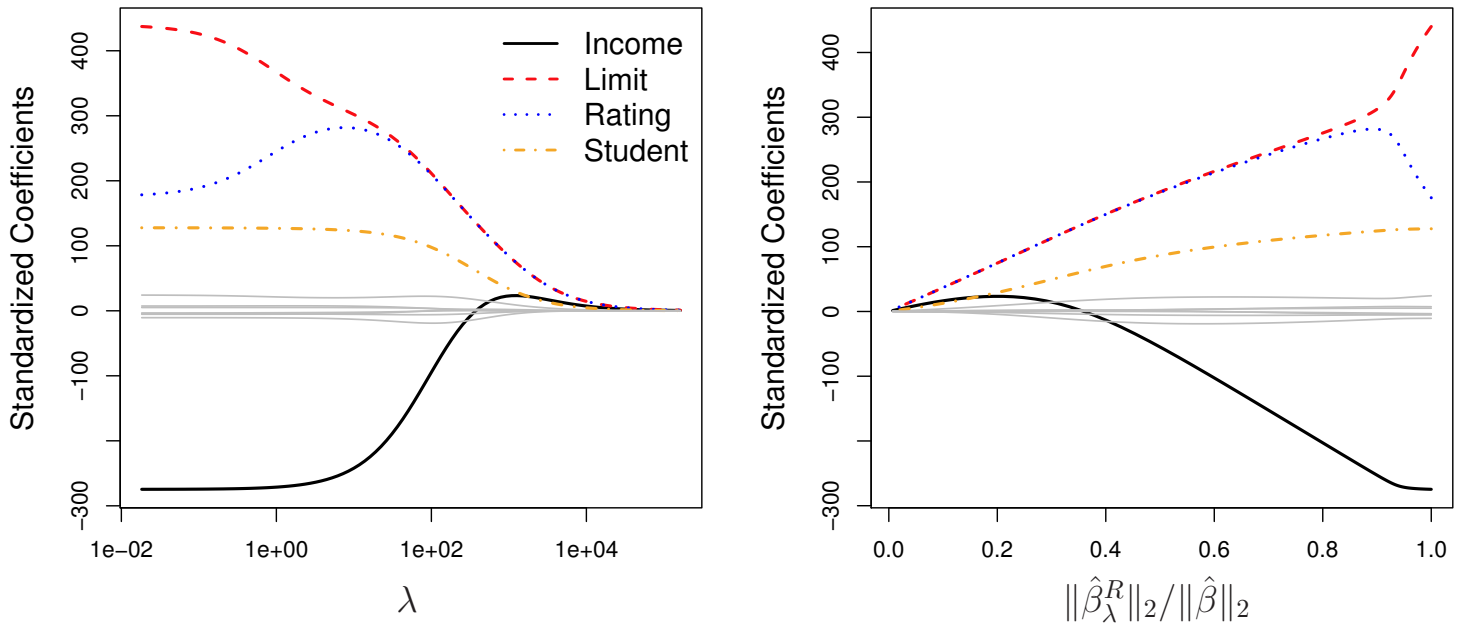

Figure: 6.4. The standardized ridge regression coefficients are displayed for the Credit data set, as a function of λ and $\|\hat{\beta}_\lambda^R\|_2/\|\hat{\beta}\|_2$. Here $\|\mathbf{a}\|_2 = \sqrt{\sum_{j=1}^{p} a_j^2}$.

Bias-variance tradeoff (why ridge improves over LSE)

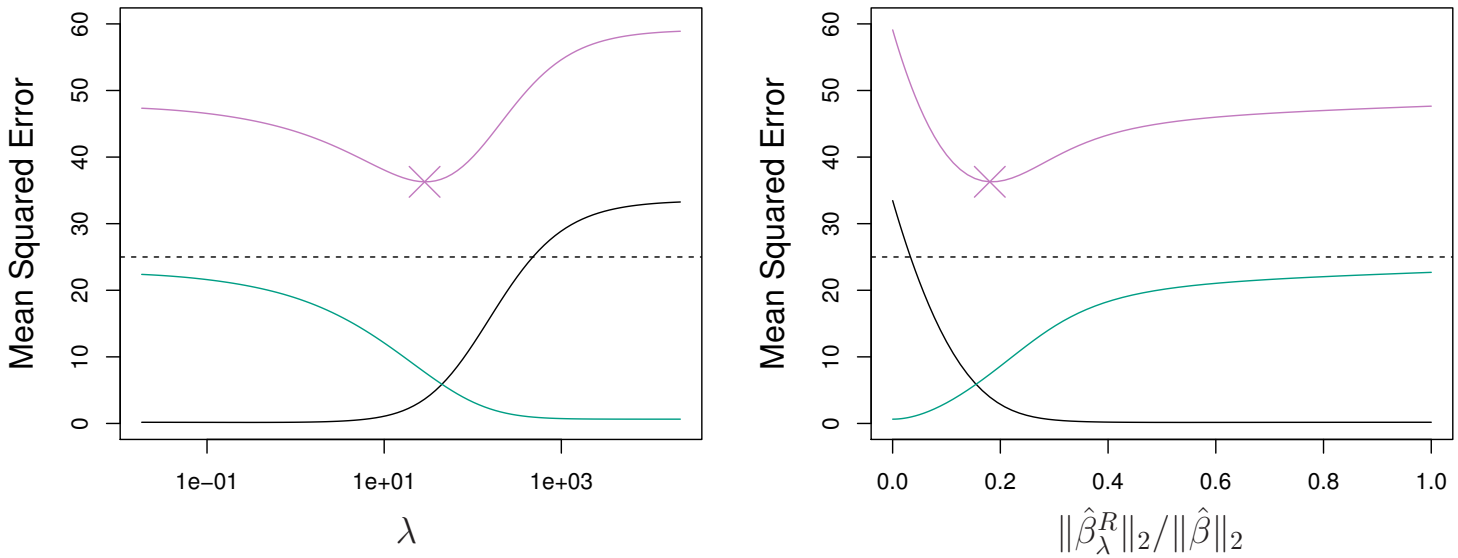

Figure: 6.5. Simulated data ($p = 45, n = 50$). Squared bias (black), variance (green), and test mean squared error (purple) for the ridge regression predictions on a simulated data set, as a function of λ and $\|\hat{\beta}^R_\lambda\|_2/\|\hat{\beta}\|_2$. The horizontal dashed lines indicate the minimum possible MSE. The purple crosses indicate the ridge regression models for which the MSE is smallest.

Remark.

- Suppose the response and the predictors is close to linear.
- the least squares estimates will have low bias.
- It may have high variance for relatively large p: small change in the training data can cause a large change in the least squares coefficient estimates.
- For large p, as in the example in Figure 6.5, the least squares estimates will be extremely variable.
- If $p > n$, then the least squares estimates do not even have a unique solution

Remark.

- If $p > n$, ridge regression can still perform well by trading off a small increase in bias for a large decrease in variance.
- Ridge regression works best in situations where the least squares estimates have high variance.
- Ridge regression also has substantial computational advantages
- $$\hat{\beta}_\lambda^R = (\mathbf{X}^T\mathbf{X}) + \lambda I)^{-1}\mathbf{X}^T\mathbf{y}$$

  where I is $p+1$ by $p+1$ diagnal with diagonal elements $(0, 1, 1, ..., 1)$.

The Lasso

- Lasso stands for Least Absolute Shrinkage and Selection Operator.
- The Lasso estimator $\hat{\beta}_\lambda^L$ is the minimizer of

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p}|\beta_j|$$

- We may use $\|\beta\|_1 = \sum_{j=1}^{p}|\beta_j|$, which is the l_1 norm.
- LASSO often shrinks coefficients to be identically 0. (This is not the case for ridge)
- Hence it performs variable selection, and yields sparse models.

Example: Credit data.

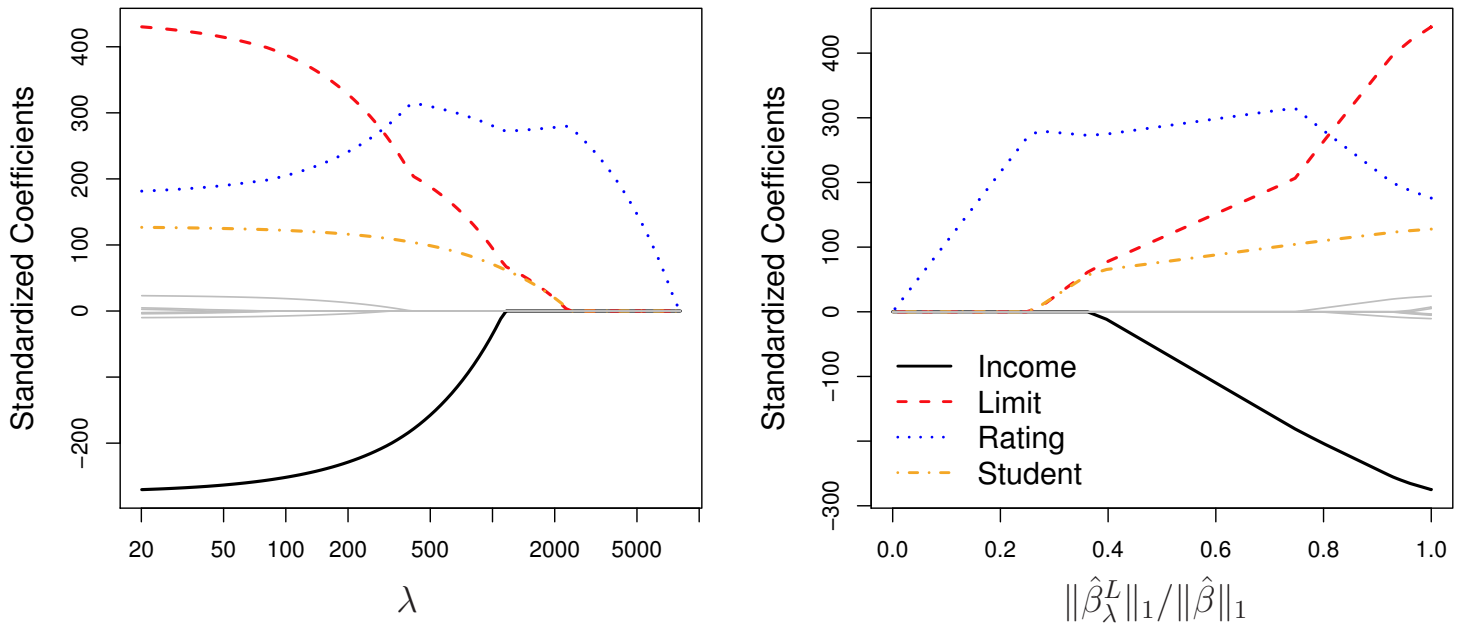

Figure: 6.6. The standardized lasso coefficients on the Credit data set are shown as a function of λ and $\|\hat{\beta}^L_\lambda\|_1/\|\hat{\beta}\|_1$.

Another formulation

- For Lasso: Minimize

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 \text{ subject to } \sum_{j=1}^{p}|\beta_j| \leq s$$

- For Ridge: Minimize

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 \text{ subject to } \sum_{j=1}^{p}\beta_j^2 \leq s$$

- For l_0: Minimize

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 \text{ subject to } \sum_{j=1}^{p}I(\beta \neq 0) \leq s$$

l_0 method penalizes number of non-zero coefficients. A difficult (NP-hard) problem for optimization.

Variable selection property for Lasso

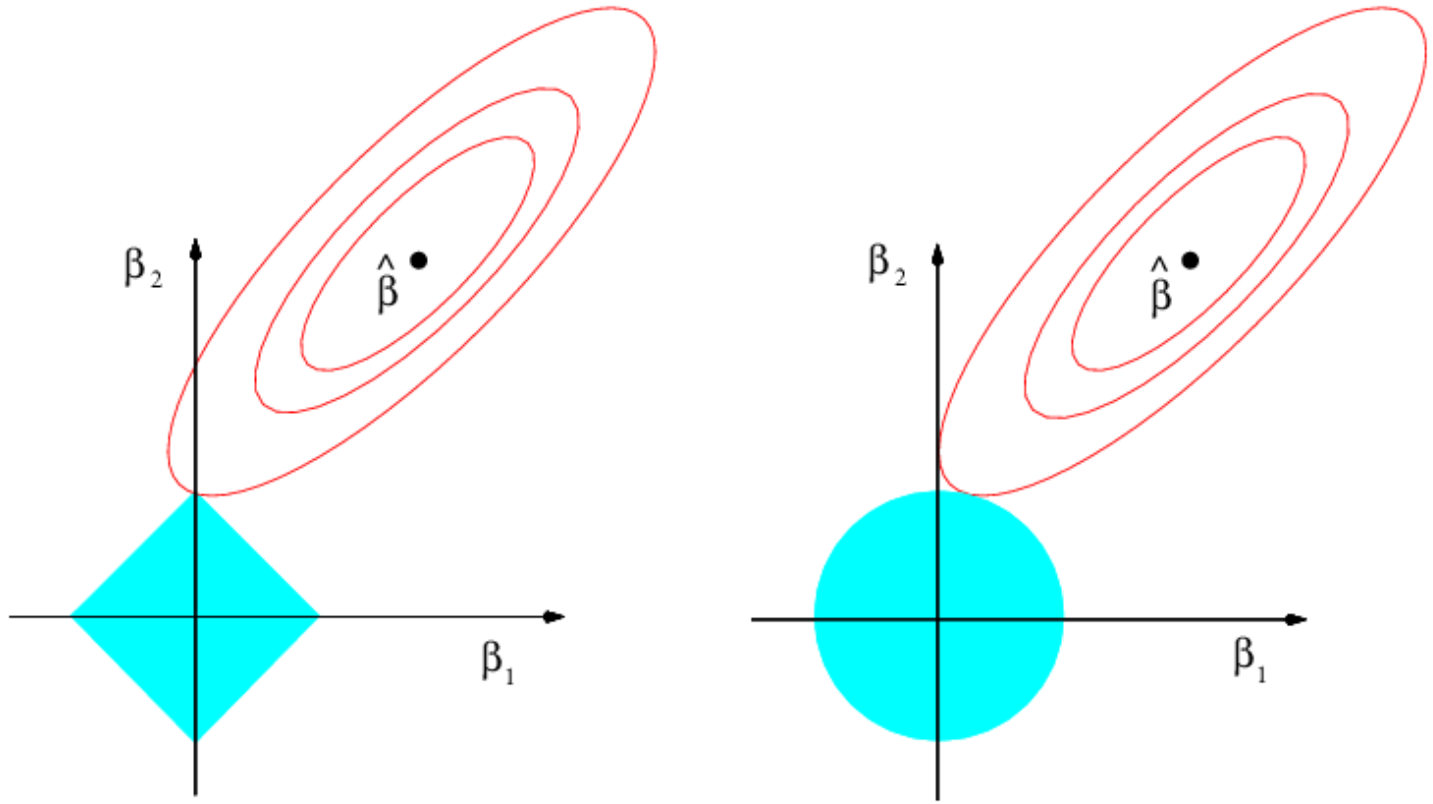

Figure: 6.7. Contours of the error and constraint functions for the lasso (left) and ridge regression (right). The solid blue areas are the constraint regions, $|\beta_1| + |\beta_2| \leq s$ and $\beta_1^2 + \beta_2^2 \leq s$, while the red ellipses are the contours of the RSS.

Simulated data as in Figure 6.5 for comparing Lasso and ridge

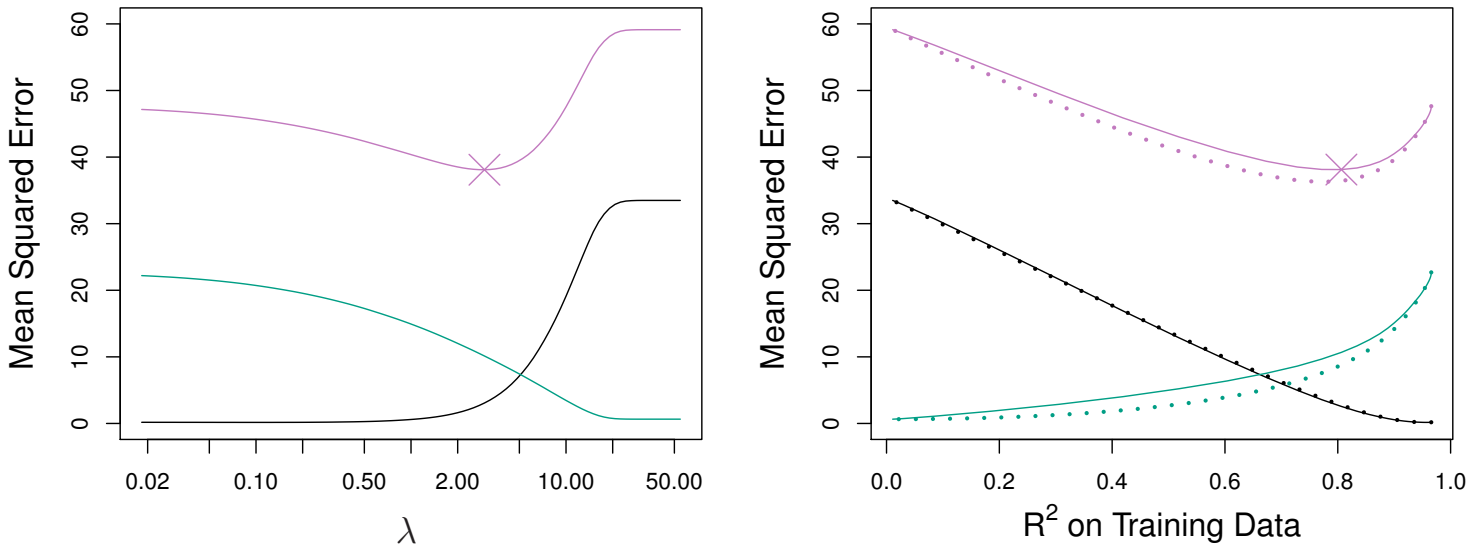

Figure: 6.8. Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso on a simulated data set. Right: Comparison of squared bias, variance and test MSE between lasso (solid) and ridge (dotted). Both are plotted against their R2 on the training data, as a common form of indexing. The crosses in both plots indicate the lasso model for which the MSE is smallest.

The Lasso

- In the previous example, ridge is slightly better than Lasso.
- The data in Figure 6.8 were generated in such a way that all 45 predictors were related to the response.
- None of the true coefficients $\beta_1, ..., \beta_{45}$ equaled zero.
- The lasso implicitly assumes that a number of the coefficients truly equal zero.
- Not surprising that ridge regression outperforms the lasso in terms of prediction error in this setting.
- In the next figure, only 2 out of 45 predictors are related with the response.

Comparing Lasso and ridge, different data

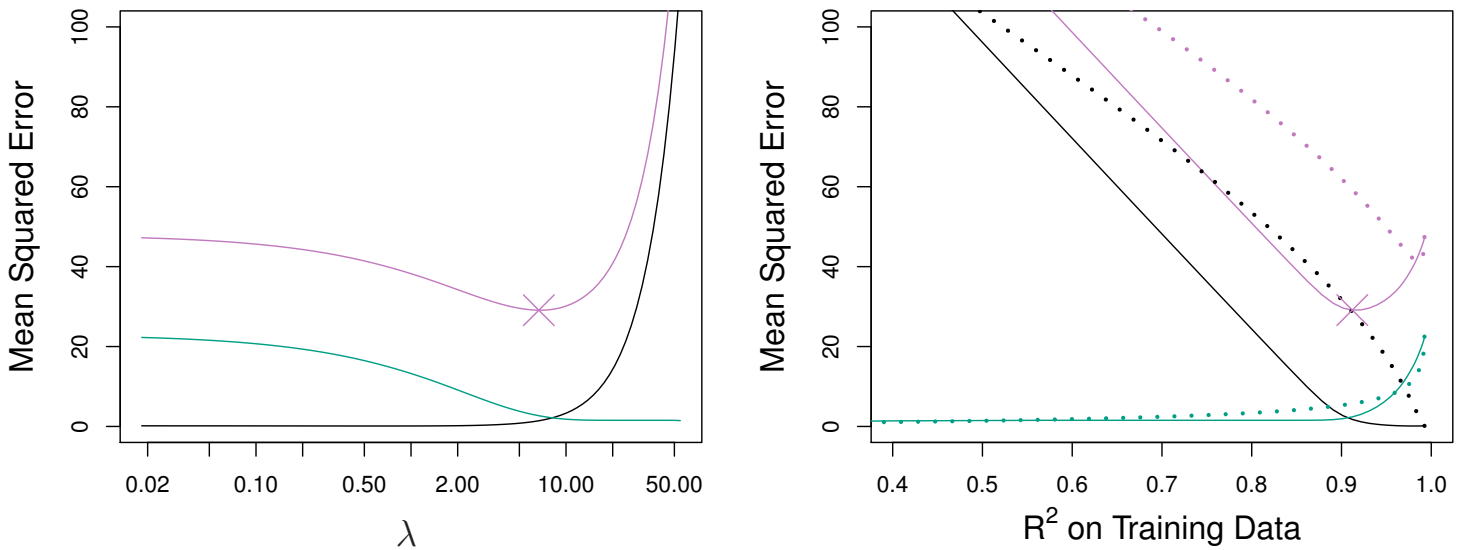

Figure: 6.9. Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso. The simulated data is similar to that in Figure 6.8, except that now only two predictors are related to the response. Right: Comparison of squared bias, variance and test MSE between lasso (solid) and ridge (dotted). Both are plotted against their R^2 on the training data. The crosses in both plots indicate the lasso model for which the MSE is smallest.

Summary remark

- Both ridge and Lasso can improve over the traditional least squares by trade off variance with bias.
- There are significant improvement when the variance of the least squares is large, mostly with small n and large p.
- Lasso has feature selection, while ridge does not.
- Use cross validation to determine which one has better prediction.

Simple cases

- Ridge has closed form solution. Lasso generally does not have closed form solution.
- Consider the simple model $y_i = \beta_i + \epsilon_i$, $i = 1, ..., n$ and $n = p$. Then,
  The least squares $\hat{\beta}_j = y_j$; the ridge $\hat{\beta}_j^R = y_j/(1+\lambda)$
  The Lasso $\hat{\beta}_j^L = \text{sign}(y_j)(|y_j| - \lambda/2)_+$.
- Slightly more generally, suppose input columns of the $\mathbf{X}$ are standarized to be mean 0 and variance 1 and are orthogonal.

$$\hat{\beta}_j^R = \hat{\beta}_j^{\text{LSE}}/(1+\lambda)$$

$$\hat{\beta}_j^L = \text{sign}(\hat{\beta}_j^{\text{LSE}})(|\hat{\beta}_j^{\text{LSE}}| - \lambda/2)_+$$

for $j = 1, ..., p$.

Example

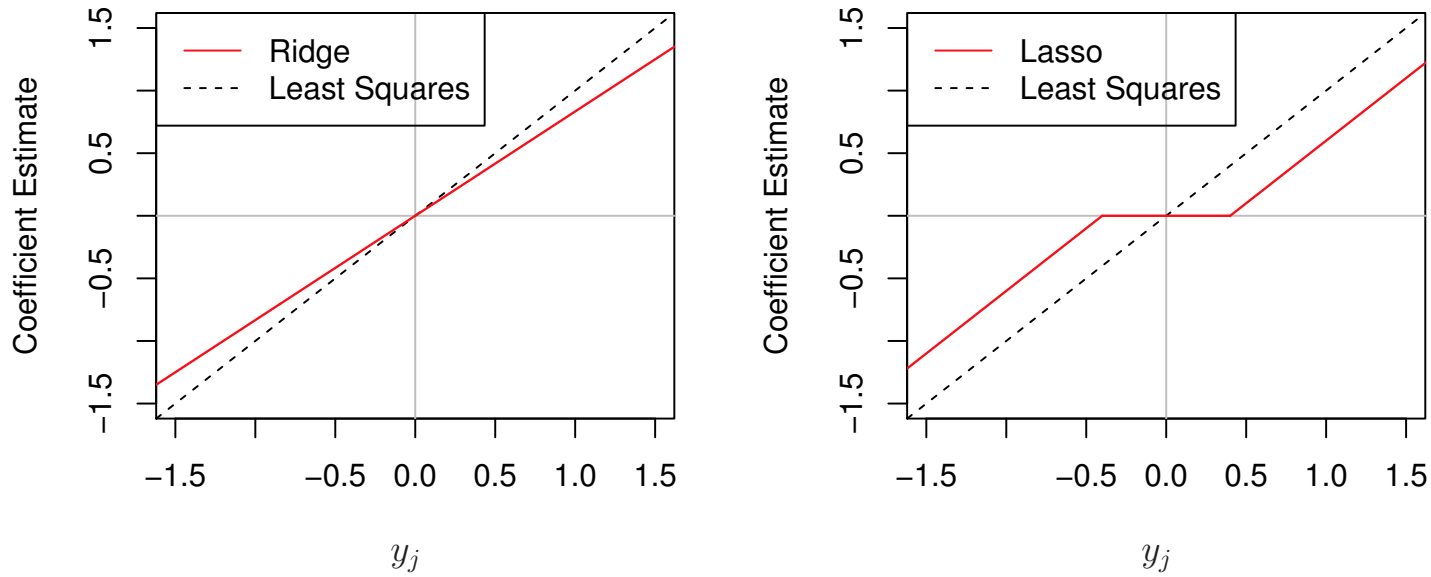

Figure: 6.10. The ridge regression and lasso coefficient estimates for a simple setting with $n = p$ and X a diagonal matrix with 1s on the diagonal. Left: The ridge regression coefficient estimates are shrunken proportionally towards zero, relative to the least squares estimates. Right: The lasso coefficient estimates are soft-thresholded towards zero.

Bayesian interpretation

- Suppose $\beta = (\beta_0, ..., \beta_p)$ are random variables with a prior distribution $p(\cdot)$.
- Given β and the input X, Y has conditional density $f(y|X, \beta)$.
- The posterior distribution of the parameter β is

$$p(\beta|X, Y) \propto f(Y|X, \beta)p(\beta|X) = f(Y|X, \beta)p(\beta)$$

The proporionality means a constant (not related with β) multiplier. (β and X are independent.)

Bayesian interpretation

- Now consider the linear regression model, $Y = \beta_0 + X_1\beta_1 + ... + X_p\beta_p + \epsilon$, with ϵ conditioning on X follows $N(0, \sigma^2)$.
- If the β has the normal prior, the prior of β following normal distribution with mean 0 then the posterior mode for β is ridge estimator.
- If the β has the double exponential prior:

$$f(t) = \lambda e^{-\lambda|t|}/2$$

the prior of β following normal distribution with mean 0 then the posterior mode for β is ridge estimator.

The Gaussian and double expoential curves

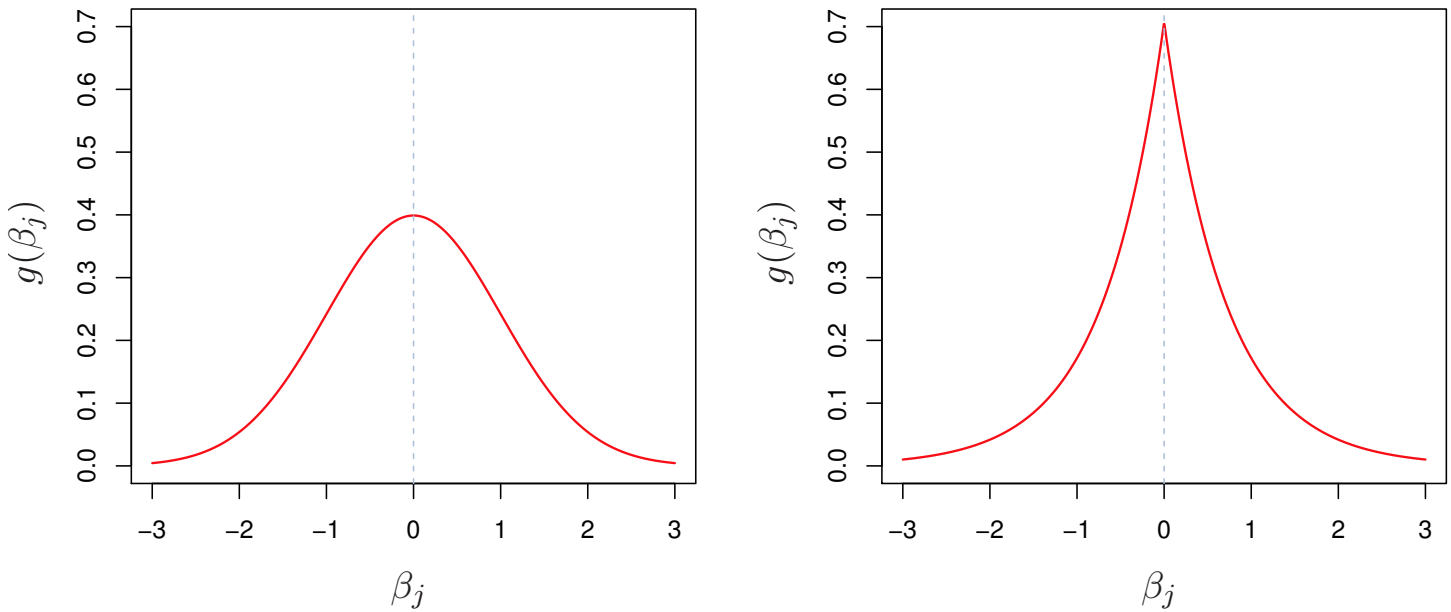

Figure: 6.11. Left: Ridge regression is the posterior mode for β under a Gaussian prior. Right: The lasso is the posterior mode for β under a double-exponential prior.

Tuning parameter selection by cross-validation: Credit data

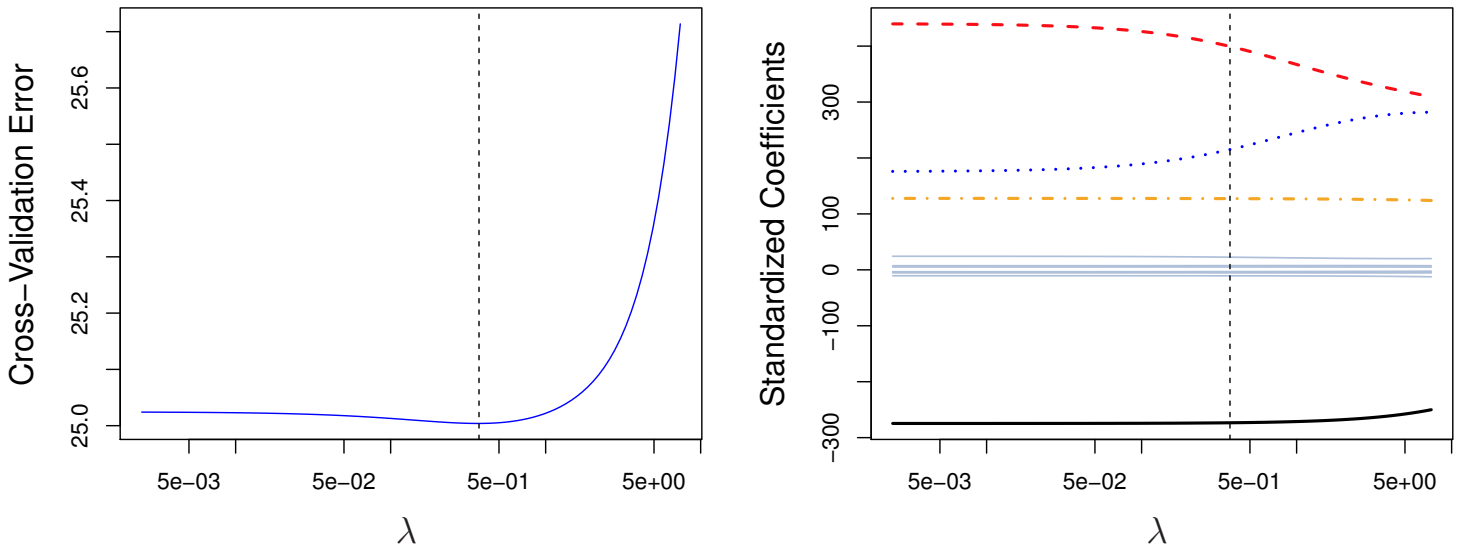

Figure: 6.12. Left: Cross-validation errors that result from applying ridge regression to the Credit data set with various value of λ. Right: The coefficient estimates as a function of λ. The vertical dashed lines indicate the value of λ selected by cross-validation.

Example

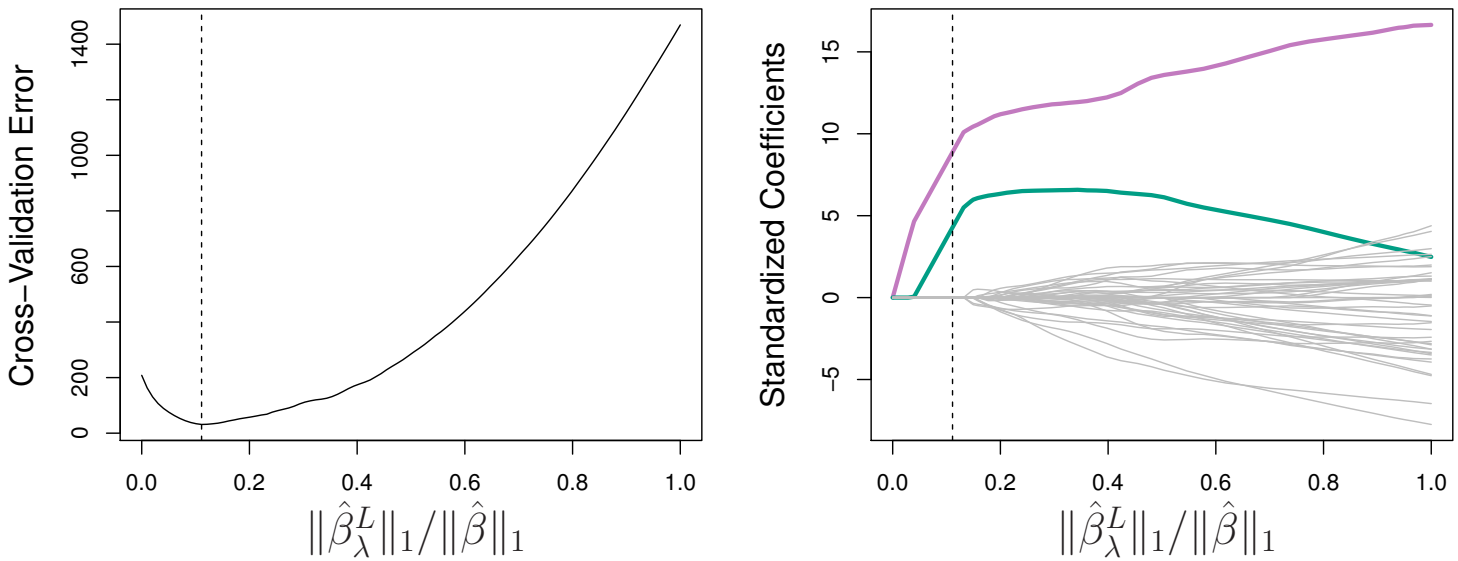

Figure: 6.13. Left: Ten-fold cross-validation MSE for the lasso, applied to the sparse simulated data set from Figure 6.9. Right: The corresponding lasso coefficient estimates are displayed. The vertical dashed lines indicate the lasso fit for which the cross-validation error is smallest.

Dimension reduction methods (using derived inputs)

- When p is large, we may consider to regress on, not the orignal inputs $X_1, ..., X_p$, but some small number of derived inputs $Z_1, ..., Z_M$ with $M < p$.

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_m + \epsilon_i, \quad i = 1, ..., n.$$

- A natural choice of Z_i is through linear combination of $X_1, ..., X_p$

$$Z_m = \sum_{j=1}^{p} \phi_{jm} X_j$$

- Essentially still a linear model with inputs $X_1, ...X_p$ but with restricted parameters

$$\beta_j = \sum_{m=1}^{M} \theta_m \phi_{jm}$$

- A key step is how to determine the linear combination.

Prinpipal Components as major statistical methodology

- Let

$$X = \begin{pmatrix} X_1 \\ \vdots \\ X_p \end{pmatrix}$$

be the random vector of p dimension that we are concerned with. For example, X may represent the returns of p stocks. As before, we use

$$\Sigma = \text{var}(X) = \begin{pmatrix} \sigma_{11} & \cdots & \sigma_{1p} \\ \vdots & \vdots & \vdots \\ \sigma_{p1} & \cdots & \sigma_{pp} \end{pmatrix}, \qquad \text{where } \sigma_{kl} = \text{cov}(X_k, X_l).$$

to denote the variance matrix X.
The mean of X plays no role in PCs, and we assume here $E(X) = 0$ for convenience.

PCA

- By matrix singular value decomposition, we know

$$\Sigma = \mathbf{e}\Lambda\mathbf{e}'$$

where

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_p \end{pmatrix} \quad \text{with } \lambda_1 \geq \cdots \geq \lambda_p > 0$$

and

$$\mathbf{e} = (\mathbf{e}_1 \vdots \cdots \vdots \mathbf{e}_p) = \begin{pmatrix} e_{11} & \cdots & e_{1p} \\ \vdots & \vdots & \vdots \\ e_{p1} & \cdots & e_{pp} \end{pmatrix} \quad \text{is an orthonormal matrix,}$$

i.e., $\mathbf{e}\mathbf{e}' = I_p$. Note that $\mathbf{e}$ is a $p \times p$ matrix and $\mathbf{e}_k$ is its k-th column and therefore is a p-vector. And $(\lambda_k, \mathbf{e}_k)$, $k = 1, ..., p$, are the eigenvalue-eigenvector pairs of the matrix Σ.

PCA

- The variation of a one dimensional random variable can be quantified by its variance.
  For a random variable X of p-dimension, its variation, fully described by its variance matrix Σ. One commonly used quantification of the total "amount" of variation of X is the trace of Σ, $trace(\Sigma)$.
  Suppose we wish to use one single variable (1-dim) to maximumly quantify the variation of all p variables, say through linear combination of the components of X. We may try to construct it so that its variance is the largest.
  Let $Y_1 = \mathbf{a}^T X$, where $\mathbf{a}$ is a p-dimensional constant vector. Then

$$\text{var}(Y_1) = \mathbf{a}^T \text{var}(X)\mathbf{a} = \mathbf{a}^T \Sigma \mathbf{a}.$$

PCA

- We wish to identify a Y_1, so that its variance is the largest. This variance depends on the scale of $\mathbf{a}$, which can be measured by its Euclidean norm.
- A fair comparison should require $\mathbf{a}$ to be of a fixed norm, say norm 1.
- The problem becomes searching for $\mathbf{a}$ of unit norm such that $\text{var}(Y_1)$ is the largest, i.e.,

$$\mathbf{a} = \text{argmax}\{\mathbf{b}^T\Sigma\mathbf{b} : \|\mathbf{b}\| = 1\}.$$

- Recall the singular value decomposition, $\Sigma = \mathbf{e}\mathbf{\Lambda}\mathbf{e}^T$ and that $(\lambda_i, \mathbf{e}_i)$ are eigenvalue-eigenvalue pairs, such that $\lambda_1 \geq ... \geq \lambda_p$. Notice that $\mathbf{e}_i$ are orthogonal to each other with unit norms.
- It follows that the solution is $\mathbf{e}_1$. That is, $Y_1 = \mathbf{e}_1^T X$ achieves the maximum variance which is λ_1. And we call this Y_1 the first principal component.

- After finding the first principal component that is the "most important", one can mimic the procedure to find the "second most important" variable: $Y_2 = \mathbf{a}^T X$, such that

$$\mathbf{a} = \text{argmax}\{\mathbf{b}^T \Sigma \mathbf{b} : \|\mathbf{b}\| = 1, \mathbf{b} \perp \mathbf{e}_1\}$$

  Note that $\mathbf{b} \perp \mathbf{e}_1$ is equivalent to the zero correlation between Y_1 and the search space of of Y_2.
- This implies we are looking the "second most important" in an attempt to explain most of the variation in X not explained by Y_1.

PCA

- Along the line, one can formulate the second, third ... p-th principal components. Mathematically, the solutions are:

$$Y_k = \mathbf{e}_k^T X, \qquad k = 1, ..., p$$

where Y_k is the k-th principle component with variance λ_k. Note again that

$$\text{var}(\mathbf{y}) = \mathbf{\Lambda} = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_p \end{pmatrix} \tag{1}$$

It implies the principle components are orthorganal to each other, and the first being most important, second being the second most important, ..., with the importance measured by their variances.

A summary

- Set

$$Y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_p \end{pmatrix} = \mathbf{e}'(X - \mu) = \begin{pmatrix} \mathbf{e}_1' \\ \vdots \\ \mathbf{e}_p' \end{pmatrix} (X - \mu).$$

Clearly, $Y_j = \mathbf{e}_j'(X - \mu)$. By a simply calculation,

$$\text{var}(Y) = \mathbf{e}'\text{var}(X)\mathbf{e} = \mathbf{e}'\Sigma\mathbf{e} = \mathbf{e}'\mathbf{e}\Lambda\mathbf{e}'\mathbf{e} = \Lambda.$$

In particular, $\text{var}(Y_j) = \lambda_j$, $j = 1, ..., p$, and $\text{cov}(Y_k, Y_l) = 0$, for $1 \leq k \neq l \leq p$.
Then, Y_j is called the j-th population P.C. The interpretation of the P.C.s is presented in the following.
To make it clearer, we call a linear combination of X, $b'X$ with $\|b\| = 1$ a unitary linear combination.

- (1). The first P.C. Y_1 explains the most variation among all unitary linear combinations of X. Namely,

$$\text{var}(Y_1) = \lambda_1 = \max\{\text{var}(b'X) : \|b\| = 1,\ b \in R^p\}.$$

The fraction of total variation of X explained by Y_1 is

$$\frac{\text{var}(Y_1)}{\text{var}(Y_1) + \cdots + \text{var}(Y_p)} = \frac{\lambda_1}{\lambda_1 + \cdots + \lambda_p}.$$

Note that $\lambda_1 + \cdots + \lambda_p = trace(\Sigma)$ is used to measure total variation of X.

- (2). The k-th P.C. Y_k explains the most variation not explained by the previous $k-1$ P.C.s $Y_1, ..., Y_{k-1}$ among all unitary linear combination. Specifically,

$$\text{var}(Y_k) = \lambda_k = \max\{\text{var}(b'X) : \|b\| = 1,\ b'X \perp Y_1,\ ...,\ b'X \perp Y_{k-1},\ b \in$$

Here and throughout, $\perp$ means 0 correlation. The fraction of total variation of X explained by Y_k is

$$\frac{\text{var}(Y_k)}{\text{var}(Y_1) + \cdots \text{var}(Y_p)} = \frac{\lambda_k}{\lambda_1 + \cdots \lambda_p}.$$

A summary table of PCs

		eigenvalue (variance)	eigenvector (combination coefficient)	percent of variation explained	P.C.s as linear combination of $X-\mu$
1st P.C.	Y_1	λ_1	$\mathbf{e}_1$	$\lambda_1/\sum_{j=1}^{p}\lambda_j$	$Y_1=\mathbf{e}_1'(X-\mu)$
2nd P.C.	Y_2	λ_2	$\mathbf{e}_2$	$\lambda_2/\sum_{j=1}^{p}\lambda_j$	$Y_2=\mathbf{e}_2'(X-\mu)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
p-th P.C.	Y_p	λ_p	$\mathbf{e}_p$	$\lambda_p/\sum_{j=1}^{p}\lambda_j$	$Y_1=\mathbf{e}_p'(X-\mu)$

Ramarks

- Note that $\mathbf{e}_j = (e_{1j}, ..., e_{pj})'$ is the j-th column of $\mathbf{e}$. As the P.C.s are orthogonal to each other (0 correlated), the part of variation explained by each P.C.s are distinct or non-overlapping with each other.
- The relative size of the variance of a P.C. or the percentage of total variation explained measures the importance of the P.C.. Thus the 1st P.C. is the most important, the 2nd P.C. the 2nd important, and so on.

- It is often desired to reduce the number of variables, especially when the number of variables in concern are too many. But the reduction must be done without much loss of information. P.C.s provide an ideal way of such reduction. One may retain the first k P.C.s, which altogether explains

$$\frac{\lambda_1 + \cdots + \lambda_k}{\lambda_1 + \cdots + \lambda_p}$$

of the total variation.

Viewpoint from the autoencoder

- The autoencoding is an important part of the deep learning technology.
- It involves representing the variables in two-steps: encoding and decoding; and the PCA serves as a basic example.
- For X to Y (the principal components) is the encoding step and from Y back to X is the decoding step.

$$X \overset{\text{encoding}}{\Longrightarrow} Y = \mathbf{e}^T X \overset{\text{decoding}}{\Longrightarrow} X = \mathbf{e}$$

Or, specifically,

$$X \overset{\text{encoding}}{\Longrightarrow} Y_k = \mathbf{e}_k^T X,\ k = 1, ..., p. \overset{\text{decoding}}{\Longrightarrow} X = \sum_{k=1}^{p} Y_k \mathbf{e}_k$$

- The preceeding representation is only mathematically useful.
- only the first few, say r, important principle compoents are retained. And the process becomes

$$X \overset{\text{encoding}}{\Longrightarrow} Y_k = \mathbf{e}_k^T X, k = 1, ..., r. \overset{\text{decoding}}{\Longrightarrow} X^* = \sum_{j=1}^{r} Y_j \mathbf{e}_j.$$

- A simpler view is that encoding is the zipping of original variales or data, and decoding is the unzipping of the encoded variables or data.

- The population P.C.s are only theoretical, in data analysis we need to work with their sample analogues: the sample P.C.s. Suppose there are n observations of p variables presented as

$$\mathbf{X} = \left(X_{(1)} \vdots X_{(2)} \vdots \cdots \vdots X_{(p)}\right) = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \vdots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}_{n \times p}.$$

  Then $X_{(k)}$, an n-vector, contains all n observations of the k-th variable.

- Let $\mathbf{S}$ be the sample variance matrix. By decomposition,

$$\mathbf{S} = \hat{\mathbf{e}}\hat{\Lambda}\hat{\mathbf{e}}'$$

Let

$$\begin{aligned}\mathbf{Y}_{n\times p} &= \left(Y_{(1)}\vdots Y_{(2)}\vdots\cdots\vdots Y_{(p)}\right)\\ &= \left(X_{(1)}-\bar{X}_1\vdots X_{(2)}-\bar{X}_2\vdots\cdots\vdots X_{(p)}-\bar{X}_p\right)\hat{\mathbf{e}}\end{aligned}$$

where $\bar{X}_k = (1/n)\sum_{i=1}^{n} x_{ik}$ is the sample average of the n observations of the k-th variable.

A summary of sample P.C.s

		eigenvalue (variance)	eigenvector (combination coefficient)	percent of variation explained	P.C.s as linear combination of $X - \mu$
1st P.C.	$Y_{(1)}$	$\hat{\lambda}_1$	$\hat{\mathbf{e}}_1$	$\hat{\lambda}_1 / \sum_{j=1}^{p} \hat{\lambda}_j$	$Y_{(1)} = \sum_{j=1}^{p} \hat{e}_{j1}(X_{(j)} - \bar{X}_1$
2nd P.C.	$Y_{(2)}$	$\hat{\lambda}_2$	$\hat{\mathbf{e}}_2$	$\hat{\lambda}_1 / \sum_{j=1}^{p} \hat{\lambda}_j$	$Y_{(2)} = \sum_{j=1}^{p} \hat{e}_{j2}(X_{(j)} - \bar{X}_1$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
p-th P.C.	$Y_{(p)}$	$\hat{\lambda}_p$	$\hat{\mathbf{e}}_1$	$\hat{\lambda}_p / \sum_{j=1}^{p} \hat{\lambda}_j$	$Y_{(p)} = \sum_{j=1}^{p} \hat{e}_{jp}(X_{(j)} - \bar{X}_1$

- Interpretations analogous to the population P.C.s applies to the sample P.C.s. We omit the details.

Principle component regression.

- Key assumption: a small number of principal components suffice to explain most of the variability in the data, as well as the relationship with the response.
- Choose $Z_1, .., Z_M$ as the first M principle components.
- This assumption may not hold !!!

Example

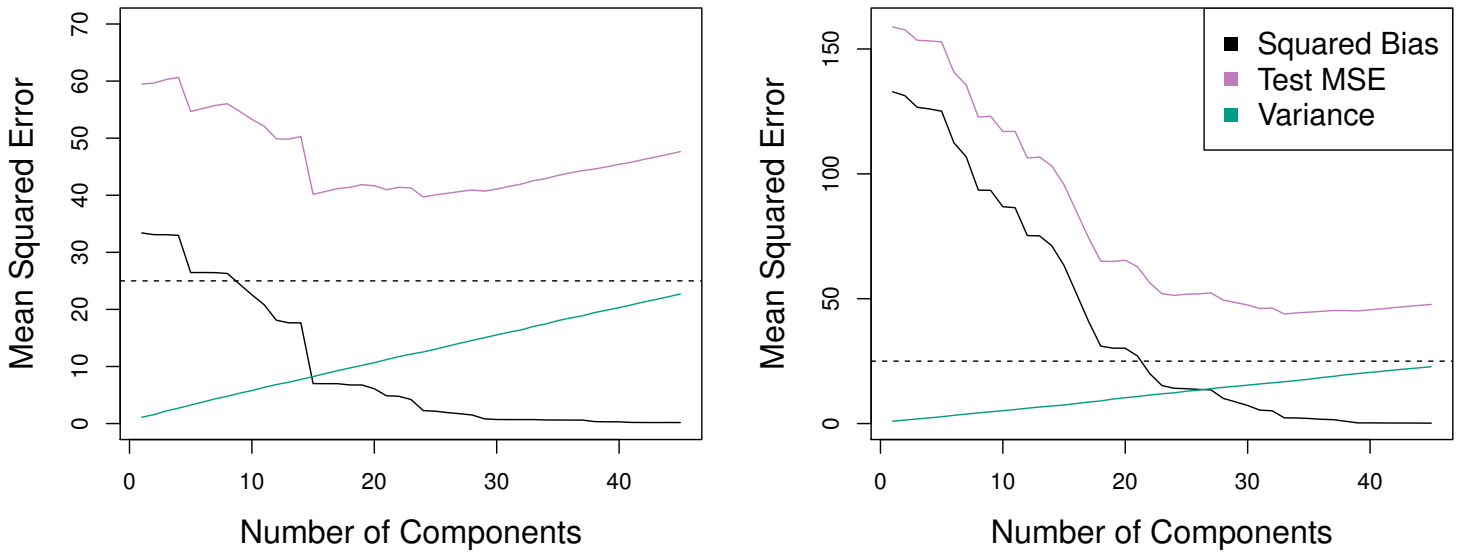

Figure: 6.18. PCR was applied to two simulated data sets. Left: Simulated data from Figure 6.8. Right: Simulated data from Figure 6.9. In comparison with figures 6.5, 6.8, 6.9, PCR does not perform as well as ridge or Lasso.

Example

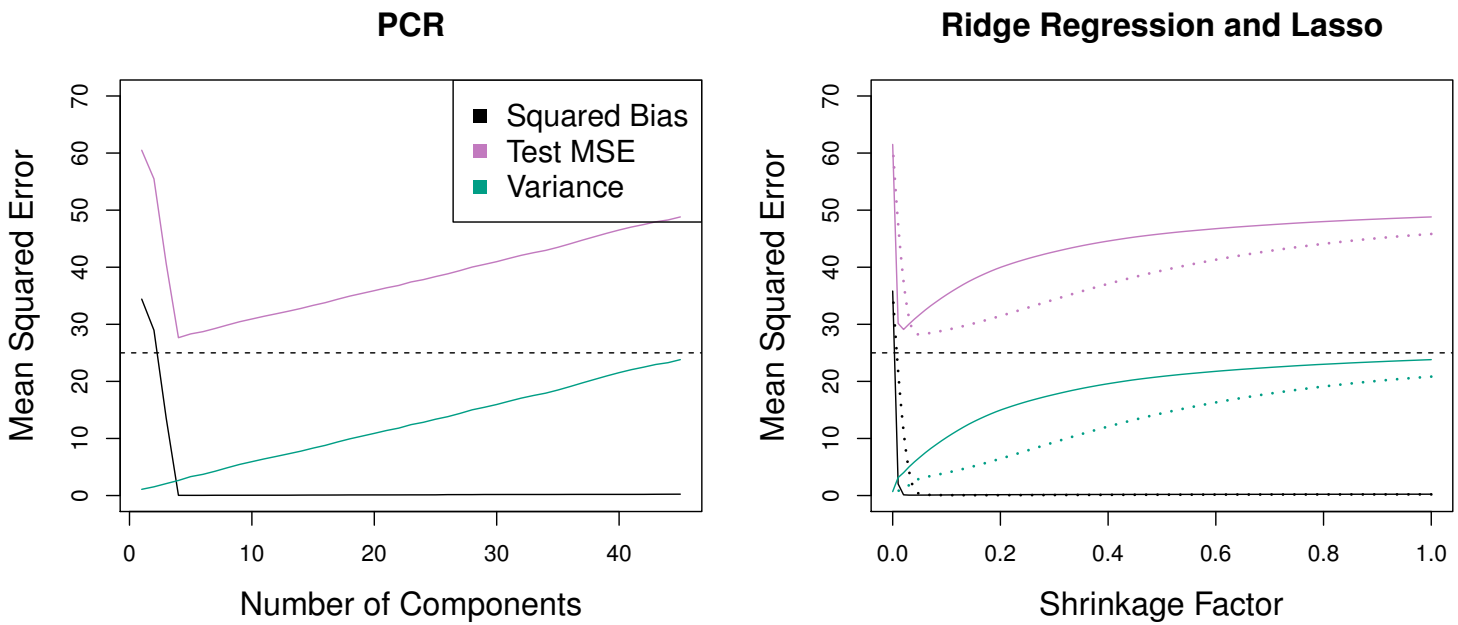

Figure: 6.19. PCR, ridge regression, and the lasso were applied to a simulated data set in which the first five principal components of X contain all the information about the response Y . In each panel, the irreducible error var(ϵ) is shown as a horizontal dashed line. Left: Results for PCR. Right: Results for lasso (solid) and ridge regression (dotted). The x-axis displays the shrinkage factor of the coefficient estimates, defined as the l_2 norm of the shrunken coefficient estimates divided by the l_2 norm of the least squares estimate.

Example

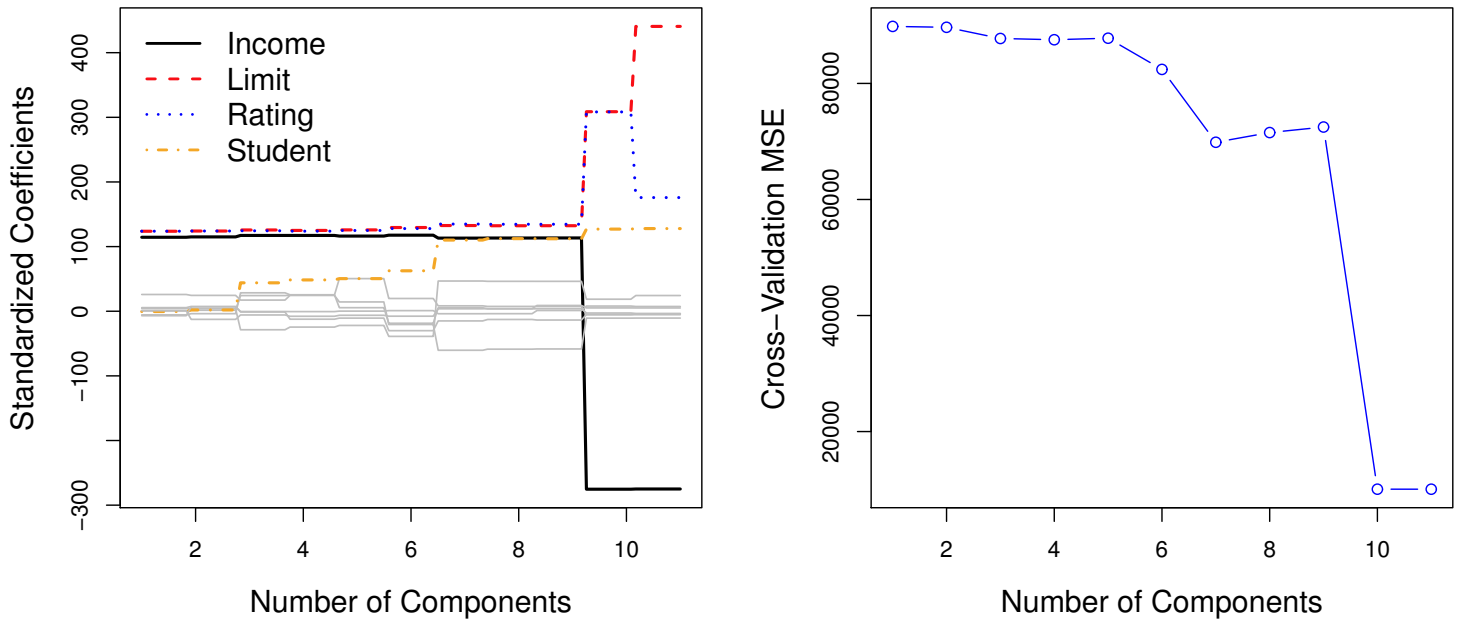

Figure: 6.20. Left: PCR standardized coefficient estimates on the Credit data set for different values of M. Right: The ten-fold cross validation MSE obtained using PCR, as a function of M.

Partial least squares approach

- Principal components are designed to explain variation within X, not the relation of X with Y.
- The key assumption with principal components regression may not hold.
- Partial least squares approach avoids this shortcoming.

Partial least squares approach

- standardize each input $\mathbf{x}_j$ to have mean 0 and variance 1. set $\hat{y}^{(0)} = \bar{y}\mathbf{1}$ and $\mathbf{x}_j^{(0)} = \mathbf{x}_j$.
- For $m = 2, ..., p$,
  $\mathbf{z}_m = \sum_{j=1}^{p} \hat{\phi}_{mj}\mathbf{x}_j^{m-1}$, where $\hat{\phi}_{mj} = \mathbf{y}^T\mathbf{x}_j^{(m-1)}$.
  $\hat{\theta}_m = \mathbf{z}_m^T\mathbf{y}/\mathbf{z}_m^T\mathbf{z}_m$
  $\hat{\mathbf{y}}^{(m)} = \hat{\mathbf{y}}^{(m-1)} + \hat{\theta}_m\mathbf{z}_m$.
  $\mathbf{x}_j^{(m)} = \mathbf{x}_j^{(m-1)} - s_{jm}\mathbf{z}_m$, where $s_{jm} = \mathbf{z}_m^T\mathbf{x}_j^{(m-1)}/\mathbf{z}_m^T\mathbf{z}_m$
- $\hat{\beta}^{\text{pls}}(m) = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}^{(m)}$

Example

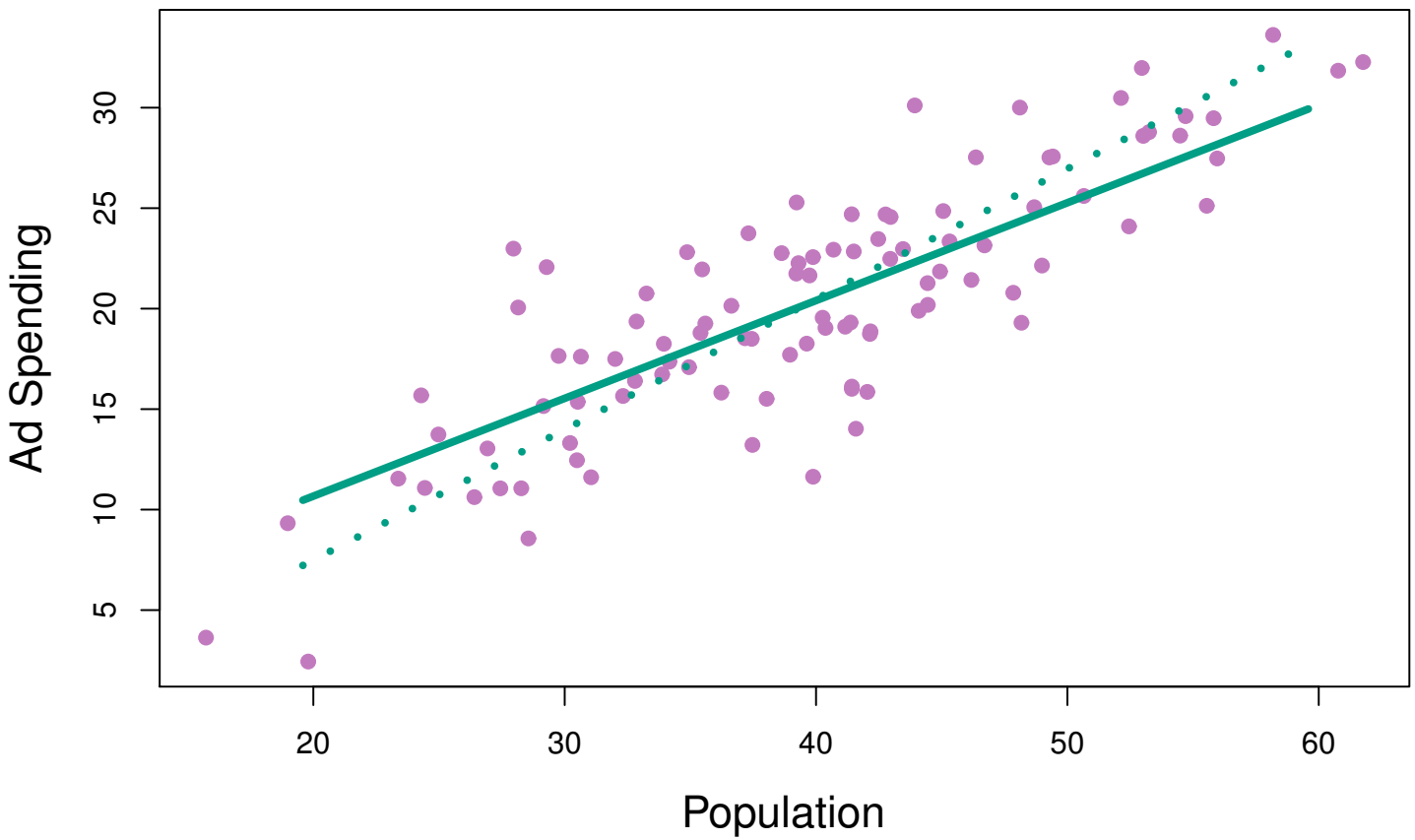

Figure: 6.21. For the advertising data, the first PLS direction (solid line) and first PCR direction (dotted line) are shown.

Partial least squares approach

- Partial least squares puts more weights on the variables with higher correlation with the response.
- It seeks the directions that have high variance and have high correlation with the reposne (while PCR only seeks those direction with high variance.)
- But it generally has lower bias but higher variance than PCR
- Popular in chemometrics.
- Many studies show it does not outperform other regularization methods such as ridge or Lasso or even PCR.

General remark

- Digitization of the society brings big data.
- Many of the datasets contain large number of varaibles.
- It is common that $p >> n$.
- Example: predition of blood pressure.
  Response: blood pressure.
  Inputs: SNPs; (Individual DNA mutations).
  n may be of hundreds, but p can be of millions.

The trouble

- Large p makes our linear regression model too flexible (or too large).
- It can easily lead to overfit.
- If $p > n$, the LSE is not even uniquely determined.
- A common phenomenon: small training error, but large test error.

Example

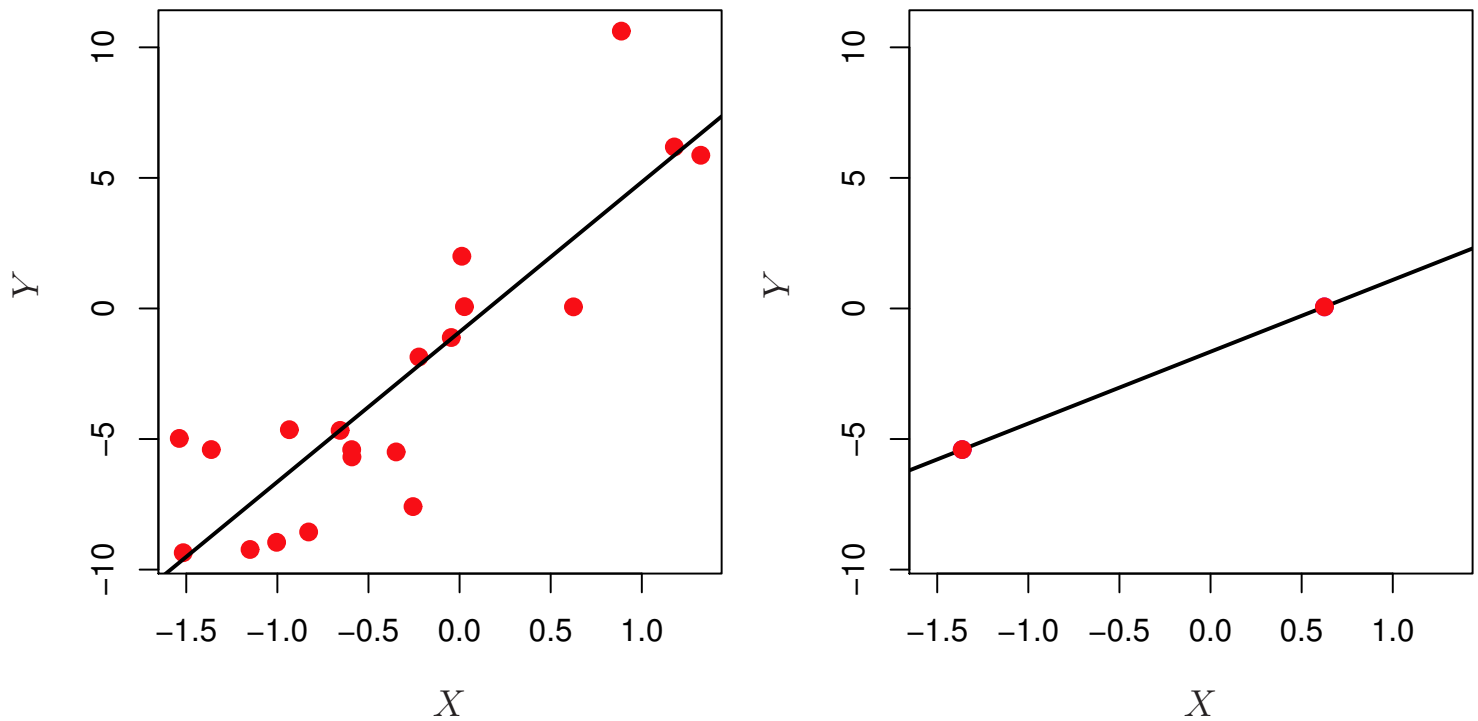

Figure: 6.22. Left: Least squares regression in the low-dimensional setting. Right: Least squares regression with $n = 2$ observations and two parameters to be estimated (an intercept and a coefficient).

Example

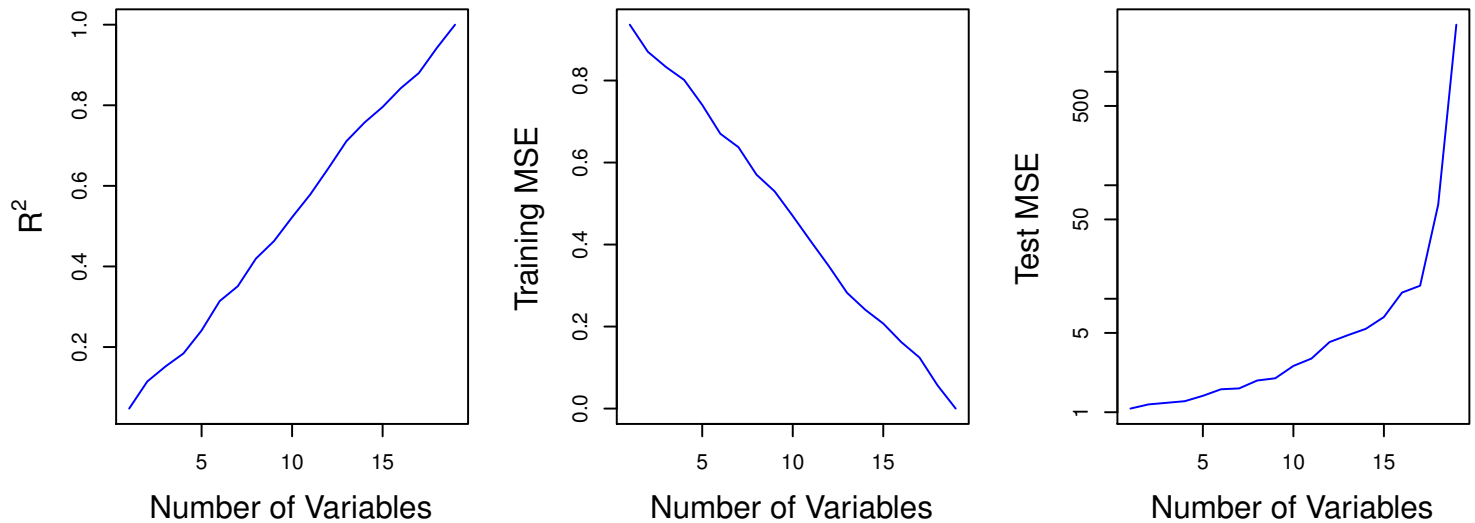

Figure: 6.23. On a simulated example with n = 20 training observations, features that are completely unrelated to the outcome are added to the model. Left: The R2 increases to 1 as more features are included. Center: The training set MSE decreases to 0 as more features are included. Right: The test set MSE increases as more features are included.

Deal with high dimensional data

- Fit less flexible models to avoid overfit.
- Use forward stepwise selection, ridge regression, the lasso, and principal components regression
- Regularization or shrinkage plays important role.
- Tuning parameter choice

Example for curse of dimensionality

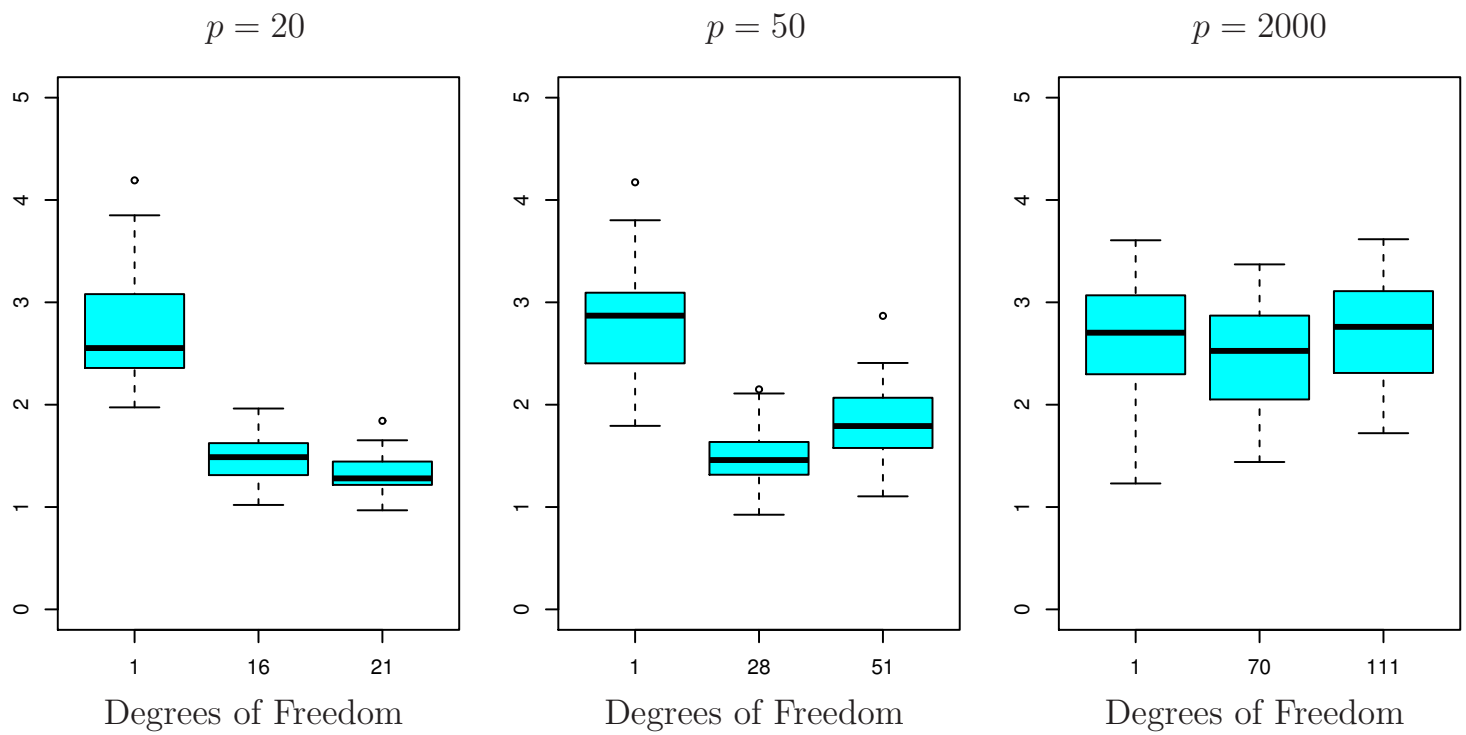

Figure: 6.24. see next page

Figure 6.24. The lasso was performed with $n = 100$ observations and three values of p, the number of features. Of the p features, 20 were associated with the response. The boxplots show the test MSEs that result using three different values of the tuning parameter λ in (6.7). For ease of interpretation, rather than reporting , the degrees of freedom are reported; for the lasso this turns out to be simply the number of estimated non-zero coefficients. When $p = 20$, the lowest test MSE was obtained with the smallest amount of regularization. When $p = 50$, the lowest test MSE was achieved when there is a substantial amount of regularization. When $p = 2,000$ the lasso performed poorly regardless of the amount of regularization, due to the fact that only 20 of the 2,000 features truly are associated with the outcome.

Caution when $p > n$.

- Extreme multicollinearity.
- Refrain from over-statement. (What we find may be one of many possible models.)
- Avoid using sum of squares, p-values, R^2, or other traditional measures of model on training as evidence of good fit.
- Place more emphasis on test error or cross validation error.

Exercises

Exercise 6.8 of ISLR: Problems 2-7.

End of Chapter 6.