Chapter 6: Linear model selection and regularization

Yu-Tzung Chang and Hsuan-Wei Lee

Department of Political Science, National Taiwan University

2018.11.22.

Outline

- Linear model selection
    - Best subset selection
    - Stepwise model selection
- Regularization
    - Ridge regression
    - Lasso regression

Linear model selection and regularization

- Recall the linear model

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

- In the lectures that follow, we consider some approaches for extending the linear model framework. In the lectures covering Chapter 7 of the textbook, we generalize the linear model in order to accommodated *non-linear*, but still *additive*, relationships.
- In the lectures covering Chapter 8, we consider even more general *non-linear* models.

In praise of linear models

- Despite its simplicity, the linear model has distinct advantages in terms of its *interpretability* and often shows good *predictive performance*.
- Hence we discuss the lecture some ways in which the simple linear model can be improved, by replacing ordinary least squares fitting with some alternative fitting procedures.

Why consider alternatives to least sqaures?

- *Prediction accuracy:* especially when $p > n$, to control the variance.
- *Model interpretability:* By removing irrelevant features – that is, by setting the corresponding coefficient estimates to zero – we can obtain a model that is more easily interpreted. We will present some approaches for automatically performing *feature selection*.

Three classes of methods

- *Subset selection.* We identify a subset of p predictors that we believe to be related to the response. We then fit a model using least squares on the reduced set of variables.
- *Shrinkage.*We fit a model involving all p predictors, but the estimated coefficients are shrunken towards zero relative to the least square estimates. This shrinkage (also known as *regularization*) has the effect of reducing variance and can also perform variable selection.
- *Dimension reduction.* We project the p predictors into a M-dimensional subspace, when $M < p$. This is achieved by computing M different *linear combinations*, or *projections*, of the variables. Then these M projections are used as predictors to fit a linear regression model by least squares.

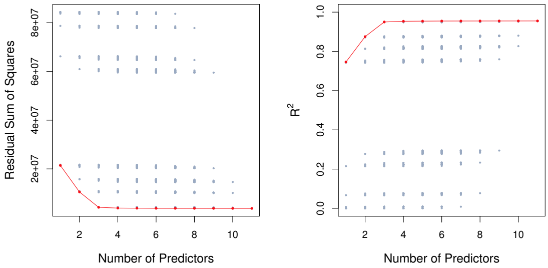
Subset selection

- Best subset selection
- Stepwise model selection

Best subset selection

1. Let $\mathscr{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.
2. For $k = 1, 2, \ldots, p$:
   1. Fit all $\binom{p}{k}$ models that contain exactly k predictors.
   2. Pick the best among these $\binom{p}{k}$ models, and call it $\mathscr{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2.
3. Select a single best model from among $\mathscr{M}_0, \mathscr{M}_1, \ldots, \mathscr{M}_p$ using cross-validated prediction error C_p, AIC, BIC, or adjected R^2.

Example – credit data set



For each possible model containing a subset of the ten predictors in the *credit card* 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 variables, leading the creation of two dummy variables.

Extension to other models

- Although we have presented best subset selection here for least squares regression, the same ideas apply to other types of models, such as logistic regression.
- The *deviance* – negative two times the maximized log-likelihood – plays the role of RSS for a broader class of models.

Stepwise selection

- For computational reasons, best subset selection cannot be applied with very large p.
- Best subset selection may also suffer from statistical problems with p is large: larger the search space, the higher the chance of finding models that look good on the training data, even though they might not have any predictive power on future data.
- Thus an enormous search space can lead to *overfitting* and high variance of the coefficient estimates.
- For both of these reasons, *stepwise* methods, which explore a far more restricted set of models, are attractive alternatives to best subset selection.

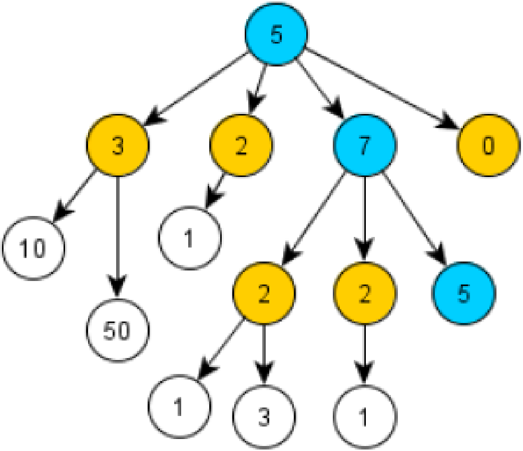
Forward stepwise selection

- Forward stepwise selection begins with a model containing no predictors, and then adds predictors, and then adds predictors to the model, one-at-a time, until all of the predictors are in the model.
- In particular, at each step the variable that gives the greatest *additional* improvement to the fit is added to the model.

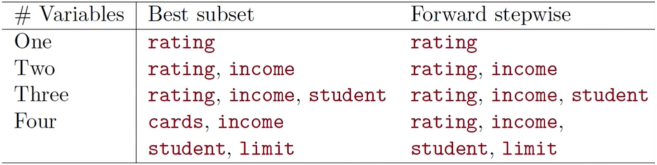
Forward stepwise selection (continued)

1. Let $\mathscr{M}_0$ denote the *null model*, which contains no predictors.
2. For $k = 1, 2, \ldots, p - 1$:
    1. Consider all $p - k$ models that augment the predictors in $\mathscr{M}_k$ with one additional predictors.
    2. Choose the *best* among these $p - k$ models, and call it $\mathscr{M}_{k+1}$. Here *best* is defined as having smallest RSS or highset R^2.
3. Select a single best model from among $\mathscr{M}_0, \mathscr{M}_1, \ldots, \mathscr{M}_p$ using cross-validated prediction error C_p, AIC, BIC, or adjusted R^2.

Greedy algorithm illustration



Credit data example



# 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`

The first four selected models for best selection and forward stepwise selection on the *credit* data set. The first three models are identical but the fourth models differ.

Backward stepwise selection

- Like forward stepwise selection, *backward stepwise selection* provides an efficient alternative to best subset selection.
- However, unlike the forward stepwise selection, it begins with the full least squares model containing all p predictors, and then iteratively removes the lease useful predictor, one-at-a-time.

Backward stepwise selection (continued)

1. Let $\mathscr{M}_p$ denote the *full model*, which contains all p predictors.
2. For $k = p, p-1, \ldots, 1$:
   1. Consider all k models that contain all but one of the predictors in $\mathscr{M}_k$, for a total of $k-1$ predictors.
   2. Choose the *best* among these k models, and call it $\mathscr{M}_{k-1}$. Here *best* is defined as having smallest RSS or highset R^2.
3. Select a single best model from among $\mathscr{M}_0, \mathscr{M}_1, \ldots, \mathscr{M}_p$ using cross-validated prediction error C_p, AIC, BIC, or adjusted R^2.

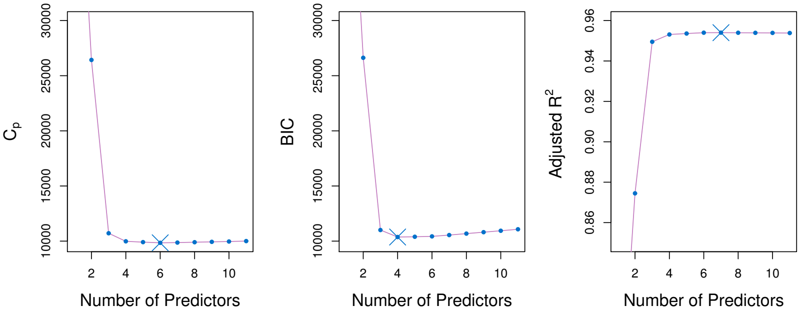
Estimating test error: two approaches

- We can indirectly estimate test error by making an *adjustment* to the training error to account for the bias due to overfitting.
- We can *directly* estimate the test error using either a validation set approach or a cross-validation approach, as discussed in the previous chapters.
- We illustrate both approaches next.

C_p, AIC, BIC, and adjusted R^2

- Idea: we only work with *training* data and *penalize* model with more variables.
- These techniques adjust the training error for the model size, and can be used to select among a set of models with different numbers of variables.
- The next figure displays C_p, AIC, BIC, and adjusted R^2 for the best model of each size produced by best subset selection on the *credit* data set.

Credit data example



C_p, AIC, BIC, and adjusted R^2 (continued)

- *Mallow's C_p*:
$$C_p = \frac{1}{n}(RSS + 2d\hat{\sigma}^2),$$
where d is the total number of parameters used and $\hat{\sigma}^2$ is an estimate of the variable of the error ϵ associated with each response measurement.
- The *AIC* criterion is defined for a large class of models fit by maximum likelihood:
$$AIC = -2logL + 2d$$
where L is the maximized value of the likelihood function for the estimated model.
- In the case of the linear model with Gaussian errors, maximum likelihood and least squares are the same thing, and C_p and AIC are equivalent.

C_p, AIC, BIC, and adjusted R^2 (continued)

- $$BIC = \frac{1}{n}(RSS + log(n)d\hat{\sigma}^2)$$
- Like C_p, the BIC will tend to take on a small value for a model with a low test error, and so generally we select the model that has the lowest BIC value.
- Notice that BIC replaces the $2d\hat{\sigma}^2$ used by C_p with a $log(n)d\hat{\sigma}^2$ term, where n is the number of observation.
- Since $logn > 2$ when $n > 7$, the BIC statistic generally places a heavier penalty on models with many variables, and hence results in the selection of smaller methods than C_p.

C_p, AIC, BIC, and adjusted R^2 (continued)

- For a least squares model with d variables, the adjusted R^2 statistic is calculated as

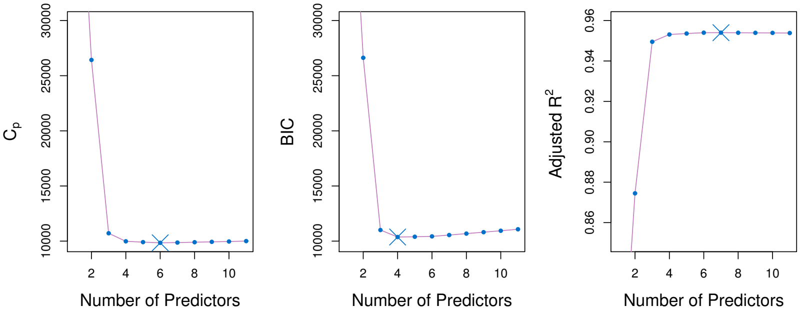
$$\text{Adjusted } R^2 = 1 - \frac{RSS/(n-d-1)}{TSS/(n-1)},$$

  where TSS is the total sum of squares.
- Unlike C_p, AIC, and BIC, for which a *small* value indicates a model with a low test error, a *large* value of adjusted R^2 indicates a model with a small test error.
- Maximizing the adjusted R^2 is equivalent to minimizing $\frac{RSS}{n-d-1}$. While RSS always decreases as the number of variables in the model increases, $\frac{RSS}{n-d-1}$ may increase or decrease, due to the presence of d in the denominator.
- Unlike the R^2 statistic, the adjusted R^2 statistic *pays a price* for the inclusion of unnecessary variables in the model.

Validation and cross-validation

- Each of the procedures returns a sequence of models $\mathscr{M}_k$ indexed by model size $k = 0, 1, 2, \ldots$. Our job is to select $\hat{k}$, once selected, we will return model $\mathscr{M}_k$.

Credit data example



Details of the previous figure

- The validation errors were calculated by randomly selecting three-quarters of the observations as the training set, and the remainder as the validation set.
- The cross-validation errors were computed using $k = 10$ folds. In this case, the validation and cross-validation methods both result in a six-variable method.
- However, all three approaches suggest that the four-, five-, and six-variable models are roughly equivalent in terms of their test errors.
- In this setting, we can select a model using the *one-standard-error rule*. We first calculate the standard error of the estimated test MSE for each model size, and then select the smallest model for which the estimated test error is within one standard error of the lower point on the curve.
- The simpler the better!

Shrinkage methods

- The subset selection methods use least squares to fit a linear model that contains a subset of the predictors.
- As an alternative, we can fit a model containing all p predictors using a technique that *constrains* or *regularizes* the coefficients, or equivalently, that *shrinks* the coefficient estimates towards zero.
- It may not be immediately obvious why such a constraint should improve the fit, but it turns out that shrinking the coefficient estimates can significantly reduce their variance.

Ridge regression

- Recall that the least square fitting procedure estimates $\beta_0, \beta_1, \dots, \beta_p$ using the values that minimize

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

- In contrast, the ridge regression coefficient estimates $\hat{\beta}^R$ are the values that minimize

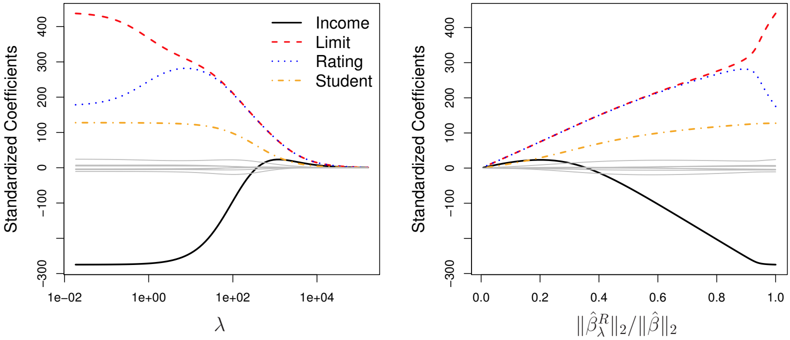
$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = RSS + \lambda \sum_{j=1}^{p} \beta_j^2,$$

where $\lambda \geqslant 0$ is a *tuning parameter*, to be determined separately.

Ridge regression (continued)

- As with least squares, ridge regression seeks coefficient estimates that fit the data well by making the RSS small.
- However, the second term, $\lambda \sum_{j=1}^{p} \beta_j^2$, called a *shrinkage penalty*, is small when $\beta_1, \ldots, \beta_p$ are close to zero, and so it has the effect of *shrinking* the estimates of β_j towards zero.
- The tuning parameter λ serves to control the relative impact of these two terms on the regression coefficient estimates.
- Selecting a good value for λ is critical; cross-validation is used for this.

Credit data example



where $||\hat{\beta}||_2 = \sqrt{\beta_1^2 + \beta_2^2 + \dots \beta_p^2}$

Details of the previous figure

- In the left-hand panel, each curve corresponds to the ridge regression coefficient estimate for one of the ten variables, plotted as a function of λ.
- The right-hand panel displays the same ridge coefficient estimates as the left-hand panel, but instead of displaying λ on the x-axis, we now display $||\hat{\beta}_\lambda^R||_2/||\hat{\beta}||_2$, where $\hat{\beta}$ denotes the vector of least squares coefficient estimates.
- The notation $||\beta||_2$ denotes the l_2 norm of a vector, and is defined as $||\beta||_2 = \sqrt{\sum_{j=1}^p \beta_j^2}$.

Ridge regression: scaling of predictors

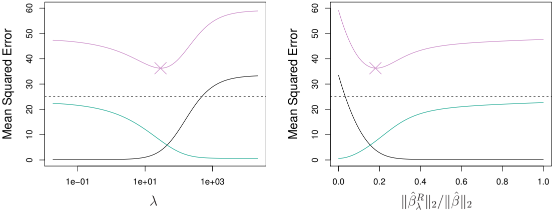
- The standard least squares coefficient estimates are *scale equivalent*: multiplying X_j by a constant c simply leads to a scaling of the least squares coefficient estimates by a factor of $1/c$. In other words, regardless of how the jth predictor is scaled, $X_j\hat{\beta}_j$ will remain the same.
- In contrast, the ridge regression coefficient estimates can change *substantially* when multiplying a given predictor by a constant, due to the sum of squared coefficients term in the penalty part of the ridge regression objective function.
- Therefore, it is best to apply ridge regression after *standardizing the predictors*, using the formula

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

- Standardizing the predictors makes the quantities *unitless*.

Why does ridge regression improve over least squares?

The bias-variance tradeoff



Simulated data with $n = 50$ observations, $p = 45$ predictors, all having nonzero coefficients. 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 $||\hat{\beta}_\lambda{}^R||_2/||\hat{\beta}||_2$. The horizontal dashed lines indicate the minimum possible MSE. The purple crosses indicate the ridge regression models for which the MSE is the smallest.

The Lasso

- Ridge regression does have one obvious disadvantage: unlike subset selection, which will generally select models that involve just a subset of the variables, ridge regression will include all p predictors in the final model.
- The *Lasso* method is a relatively recent alternative to ridge regression that overcomes this disadvantage. The Lasso coefficients, $\hat{\beta}_\lambda^L$, minimize the quantity

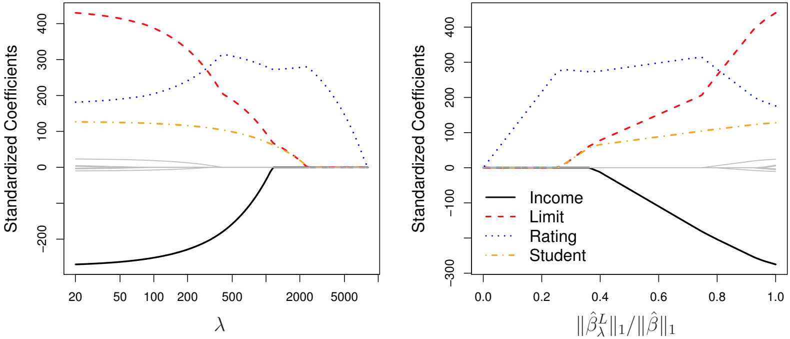
$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| = RSS + \lambda\sum_{j=1}^{p}|\beta_j|.$$

- In statistical parlance, the lasso uses an l_1 penalty instead of an l_2 penalty. The l_1 norm of a coefficient vector β is given by $||\beta||_1 = \sum|\beta_j|$.

The Lasso (continued)

- As with ridge regression, the lasso shrinks the coefficient estimates towards zero.
- However, in the case of the lasso, the l_1 penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the tuning parameter λ is sufficiently large.
- Hence, much like best subset selection, the lasso performs *variable selection*.
- We say that the lasso yields *sparse* models – that is, models that involve only a subset of the variables.
- As in ridge regression, selecting a good value of λ for the lasso is critical; cross-validation is again the method of choice.

Credit data example



The variable selection property of the Lasso

- Why is it that the lasso, unlike the ridge regression, results in coefficient estimates that are exactly equal to zero? One can show that the lasso and ridge regression coefficient estimates solve the problems

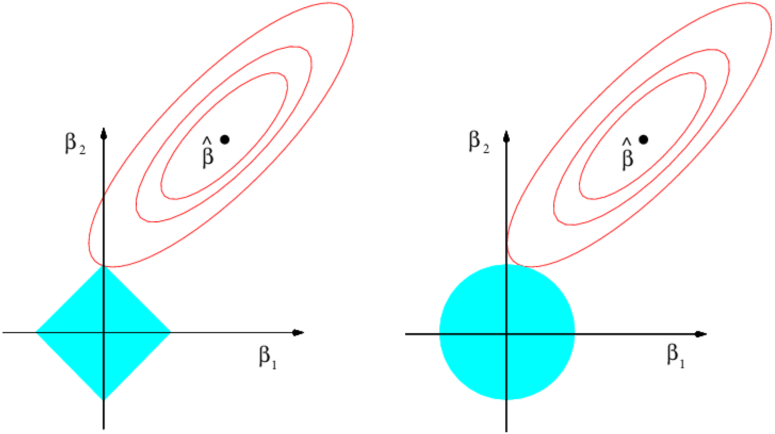
$$\text{minimize}_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \text{ subject to } \sum_{j=1}^{p} |\beta_j| \leqslant s$$

and

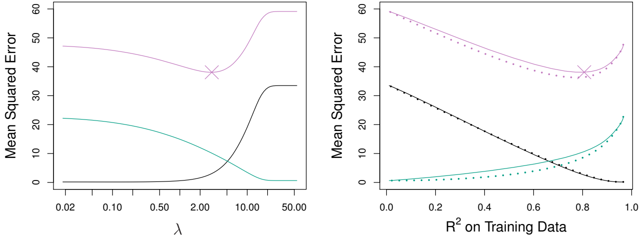
$$\text{minimize}_{\beta} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \text{ subject to } \sum_{j=1}^{p} \beta_j^2 \leqslant s,$$

respectively.

Lasso (left) and ridge (right) regression constraints

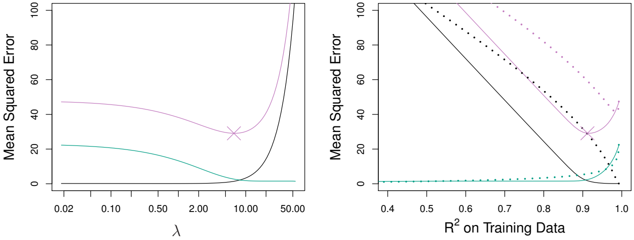


Comparing the lasso and ridge regression



- Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso on simulated data set in slide 33.
- Right: Comparison of squared bias, variance and test MSE between lasso (solid) and ridge (dashed). Both are plotted against their R^2 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.

Comparing the lasso and ridge regression (continued)



- Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso on simulated data set in slide 33, 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 (dashed). Both are plotted against their R^2 on the training data, as a common form of

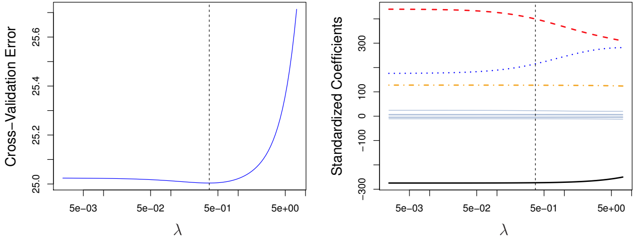
Conclusions

- These two examples illustrate that neither ridge regression nor the lasso will universally dominate the other.
- In general, one might expect the lasso to perform better when the response is a function of only a relatively small number of predictors.
- However, the number of predictors that is related to the response is never known a priori for real data sets.
- A technique such as cross-validation can be used in order to determine which approach is better on a particular data set.

Selecting the tuning parameter for ridge regression and lasso

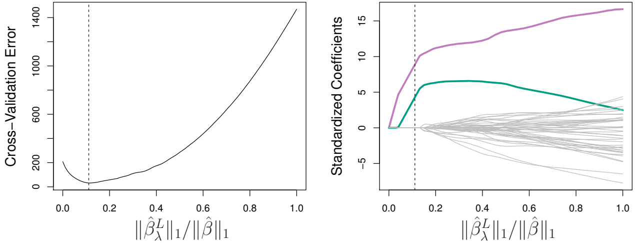
- As for subset selection, for ridge regression and lasso we require a method to determine which of the models under consideration is best.
- That is, we require a method selecting a value for the tuning parameter λ or equivalently, the value of the constraint s.
- *Cross-validation* provides a simple way to tackle this problem. We choose a grid of λ values, and compute the cross-validation error rate for each value of λ.
- We then select the tuning parameter value for which the cross-validation error is smallest.
- Finally, the model is re-fit using all of the available observations and the selected value of the tuning parameter.

Credit data example



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

Simulated data example



- Left: Ten-fold cross-validation MSE for the lasso, applied to the sparse simulated data set from Slide 39.
- Right: The corresponding lasso coefficient estimates are displayed. The vertical dashed lines indicate the lasso fit for which the cross-validation error is smallest.