STAT5003

Week 8 : Feature Selection

Dr. Justin Wishart

THE UNIVERSITY OF SYDNEY

Readings 🏛 and R functions covered

- 🏛 readings
  - Feature selection covered in Chapter 6.1-6.2 in James, Witten, Hastie, and Tibshirani (2013)
- R functions
  - `glmnet::glmnet` (Fix lasso and ridge regression models)
  - `leaps::regsubets` (Feature selection through exhaustive searching)
  - `stats::model.matrix` (Needed to interface with glmnet)

Feature Selection - Searching

Goals of feature selection

- **Prediction accuracy**: especially when $p > n$
    - where p is the number of features and n denotes number of observations
- **Model interpretability**:
    - Removing irrelevant or poor features (that is, by setting the corresponding coefficient estimates to zero) $\rightsquigarrow$ we can obtain a model that is more easily interpreted
- Some approaches for feature selection are presented.

Approaches for feature selection

1. **Subset selection**:
    - Identify a subset of the p predictors that we believe to be related to the response or class (y).
    - Fit a classification or regression model on the reduced set of variables.
2. **Shrinkage**:
    - Primarily used for regression models
    - Fit a model involving all p predictors.
    - Some estimation coefficients are shrunk towards zero relative
    - This shrinkage (also known as regularisation) has the effect of reducing variance and can also be used for feature selection.
3. **Dimension reduction**:
    - We project the p predictors into M-dimensional subspace, $M < p$

Best subset selection

1. Denote $\mathcal{M}_0$ to be the null model
    - Contains no predictors.
2. For $k = 1, 2, \dots, p$
    - Fit all $\binom{p}{k}$ models that contain exactly k predictors
    - Denote $\mathcal{M}_k$ the best among the $\binom{p}{k}$ models.
        - Measured as best against some metric (smallest residual sum of squares or highest accuracy etc.)
3. Select the single best model among the $\mathcal{M}_0, \mathcal{M}_1, \dots, \mathcal{M}_p$
    - Using cross-validated prediction error or Residual sum of squares etc.

- Consider as an example Linear regression
    - $\mathcal{M}_0 : Y = \beta_0 + \varepsilon$
    - $\mathcal{M}_p : Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \varepsilon$

Best subset selection methods

- It can be too computationally expensive to apply best subset selection when p is large.
  - Too many possible feature subsets.
- Statistical problems with large p
  - Larger search space $\rightsquigarrow$ increased chance of findings models that overfit.
  - Perform well on training data

Forward Stepwise selection

- Forward stepwise selection begins with a model containing no 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

1. Denote $\mathcal{M}_0$ to be the null model (e.g. $Y = \beta_0 + \varepsilon$ in linear reg)
   - Contains no predictors.
2. For $k = 0, 1, 2, \ldots, p-1$
   - 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 assign it as $\mathcal{M}_{k+1}$.
     - Best measured against some metric (RSS or classification error)
3. Select the single best model among the $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_p$
   - Using cross-validated prediction error or Residual sum of squares etc.

- Computational advantage over best subset selection is clear.
- However,
  - not guaranteed to find the best possible model out of all 2^p models containing subsets of the p predictors.
  - Why not?

Backward stepwise selection

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

Backward stepwise selection

1. Denote $\mathcal{M}_p$ to be the **full** model (e.g. $Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \varepsilon$ in linear reg)
    - Contains all predictors.
2. For $k = p, p-1, \ldots, 1$
    - 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 assign it as $\mathcal{M}_{k-1}$.
        - Best measured against some metric (RSS or classification error)
3. Select the single best model among the $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_p$
    - Using cross-validated prediction error or Residual sum of squares etc.

More on backward stepwise selection

- Similarities to forward selection
  - it searches through only $1 + p(p+1)/2$ models
  - can be applied in settings where p is too large to apply best subset selection
  - backward selection is not guaranteed to yield the best model containing a subset of the p predictors.
- **Note**: for some models such as linear regression, backward selection requires that the number of cases n is larger than the number of features p (so that the full model can be fit).
  - In contrast, forward stepwise can be used even when $n < p$.

Direct vs Indirect methods

Linear model (feature) selection

- Recall the linear model

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \varepsilon$$

How to choose the optimal model?

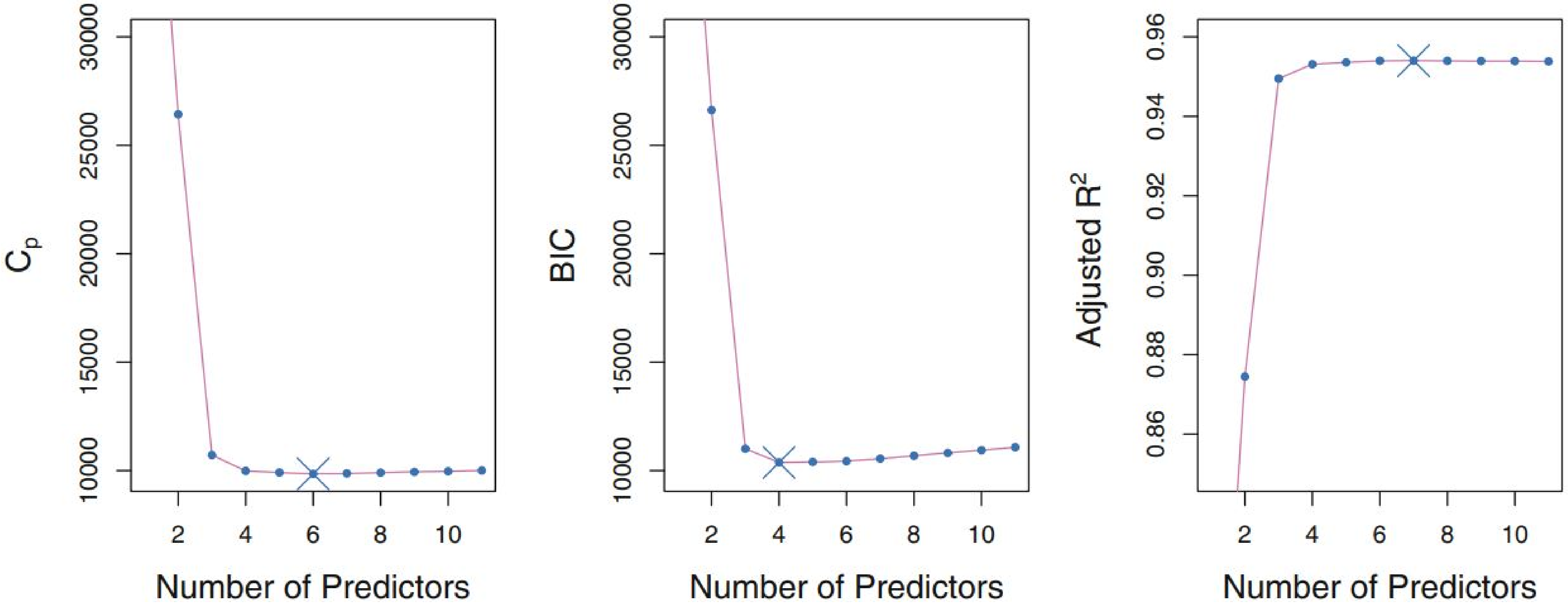
- The model containing all of the predictors will always have the smallest RSS, since these quantities are related to the training error.
- We wish to choose a model with low test error, not a model with low training error.
  - Training error is usually a poor estimate of test error.
- Therefore, RSS are not suitable for selecting the best model among a collection of models with different number of predictors.

Estimating test error - two approaches

- **Indirectly** estimate test error by making an adjustment to the training error.
    - Account for the bias due to overfitting.
- **Directly** estimate the test error, using either a test set or cross-validation set approach (covered in previous lectures)
- Will illustrate the **indirect** approach and also review cross-validation.

Indirect approaches (e.g. C_p and BIC)

- Adjust the training error for the model size (model complexity)
  - Can be used to select among a set of models with a different number of features
- Figure below displays Mallow's C_p and the Bayesian Information Criterion (BIC) and adjusted R^2 for the best model produced by best subset selection on the credit data set.



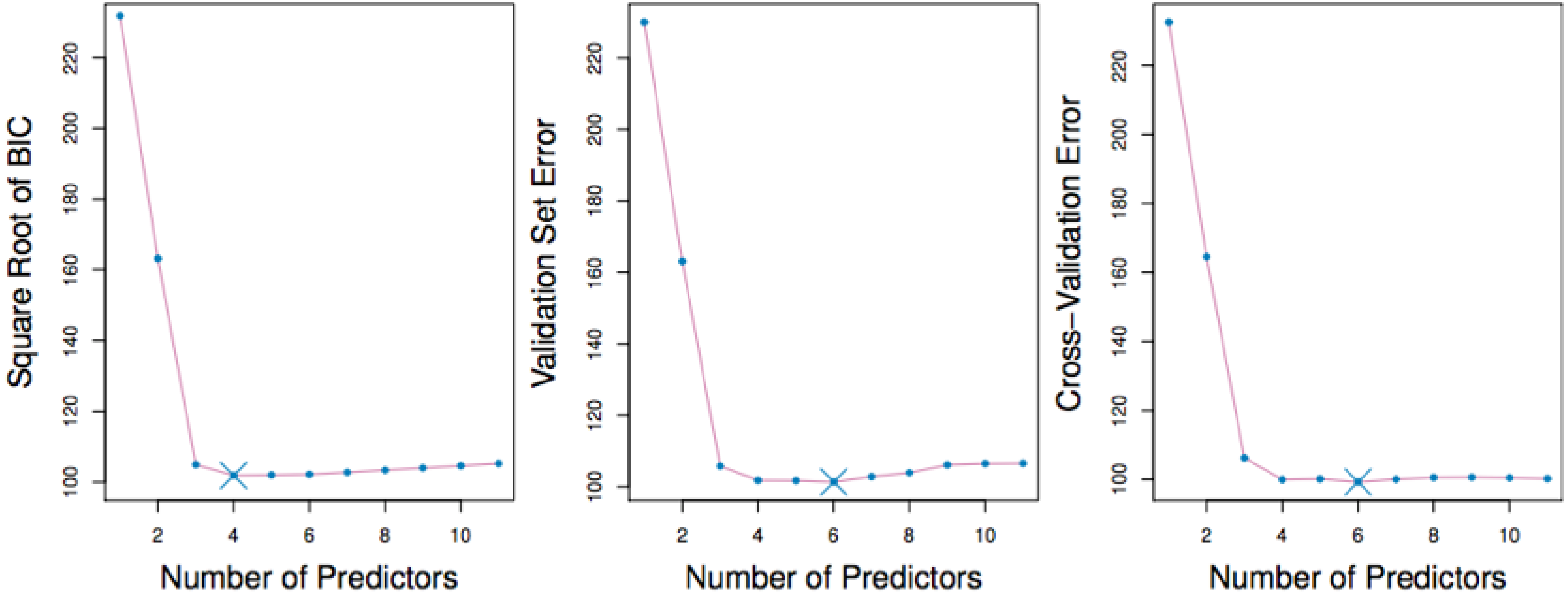
Details of C_p and BIC

- Mallow's $C_p = \frac{1}{n}(RSS + 2d\widehat{\sigma}^2)$
  - d is the total number of parameters
  - $\widehat{\sigma}^2$ is an estimate of the variance of ε
- Bayesian Information Criterion $\mathrm{BIC} = \frac{1}{n\widehat{\sigma}^2}(RSS + \log(n)d\widehat{\sigma}^2)$
- Like C_p, the BIC will tend to take on small value for model with a low test error, and so generally we select model that has the lowest BIC value.
- Notice that BIC replaces the $2d\widehat{\sigma}^2$ used by C_p with a $\log(n)d\widehat{\sigma}^2$ term, where n is the number of samples.
- Since $\log n > 2$ when $n > 7$, the BIC statistic generally places a heavier penalty on models with many variables, and hence results in the selection of smaller models than C_p.

Test set and cross-validation

- Each of the procedures returns a sequence of models $\mathcal{M}_k$ indexed by model size $k = 0, 1, 2, \ldots$. Our job here is to select k. Once selected, we will return model $\mathcal{M}_k$
- We compute the validation set error or the cross-validation error for each model $\mathcal{M}_k$
    - select the k for which the resulting estimated test error is smallest.
- This procedure has an advantage relative to C_p and BIC, in that it provides direct estimate of the test error, and doesn't require an estimate of the error variance σ^2.
- It can also be used in a wider range of model selection tasks, even in cases where it is hard to pinpoint the model degrees of freedom (e.g. the number of predictors in the model) or hard to estimate the error variance σ^2.

Credit card example



Shrinkage methods

Shrinkage methods

- We will introduce two methods specifically designed for linear regression

Ridge-regression and Lasso

- 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 coefficient estimates.
- 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 squares fitting procedure estimates $\beta_0, \beta_1, \ldots, \beta_p$ using the values that minimize

$$RSS = \sum_{i=1}^{n} \left(Y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 = ||Y - \boldsymbol{X\beta}||_2^2$$

- The ridge regression coefficient estimates $\widehat{\beta}_R$ are the values that minimize

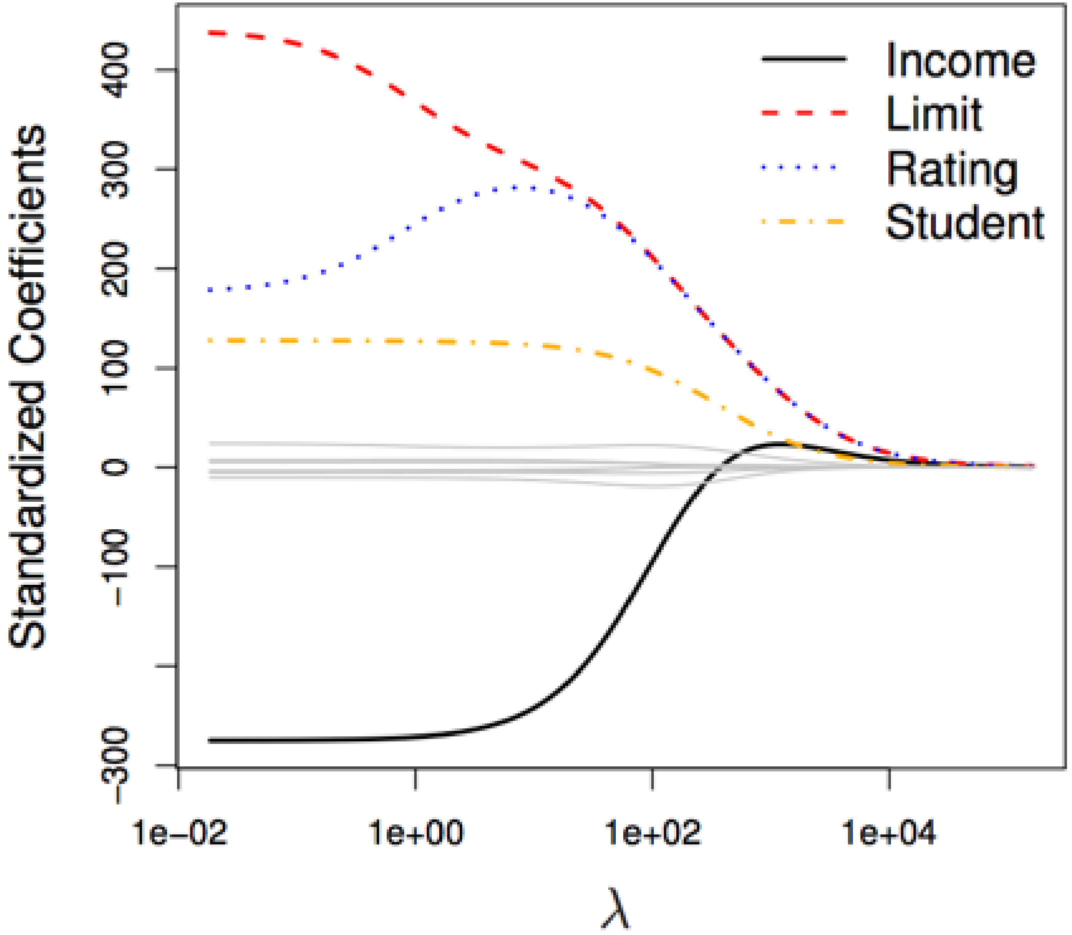
$$RSS + \lambda \sum_{j=1}^{p} \beta_j^2$$

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

Ridge regression

- As with least squares, ridge regression seeks coefficient estimates that fit the data well, by making the RSS small.
- However, the second term, $\lambda \sum_{j=1}^{p} \beta_j^2$, is 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 card example



Ridge regression: scaling of predictors

- The standard least squares coefficient estimates are scale **invariant**: 比例不变
  - 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 j^{th} predictor is scaled, X_j, $\widehat{\beta}_j$ will remain the same.
- In contrast, the ridge regression coefficients 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 **standardising the predictors**, using a formula such as below:

$$\widetilde{X_{ij}} = \frac{X_{ij}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_{ij} - \overline{X}_j)^2}}$$

The Lasso

- Ridge regression does have one obvious disadvantage:
  - Ridge regression will include all p predictors in the final model
  - Subset selection will generally select models that involve a subset of the predictors
- The Lasso is a relatively recent alternative to ridge regression that overcomes this disadvantage.
  - The lasso coefficients, $\widehat{\beta}_L$, minimise the quantity

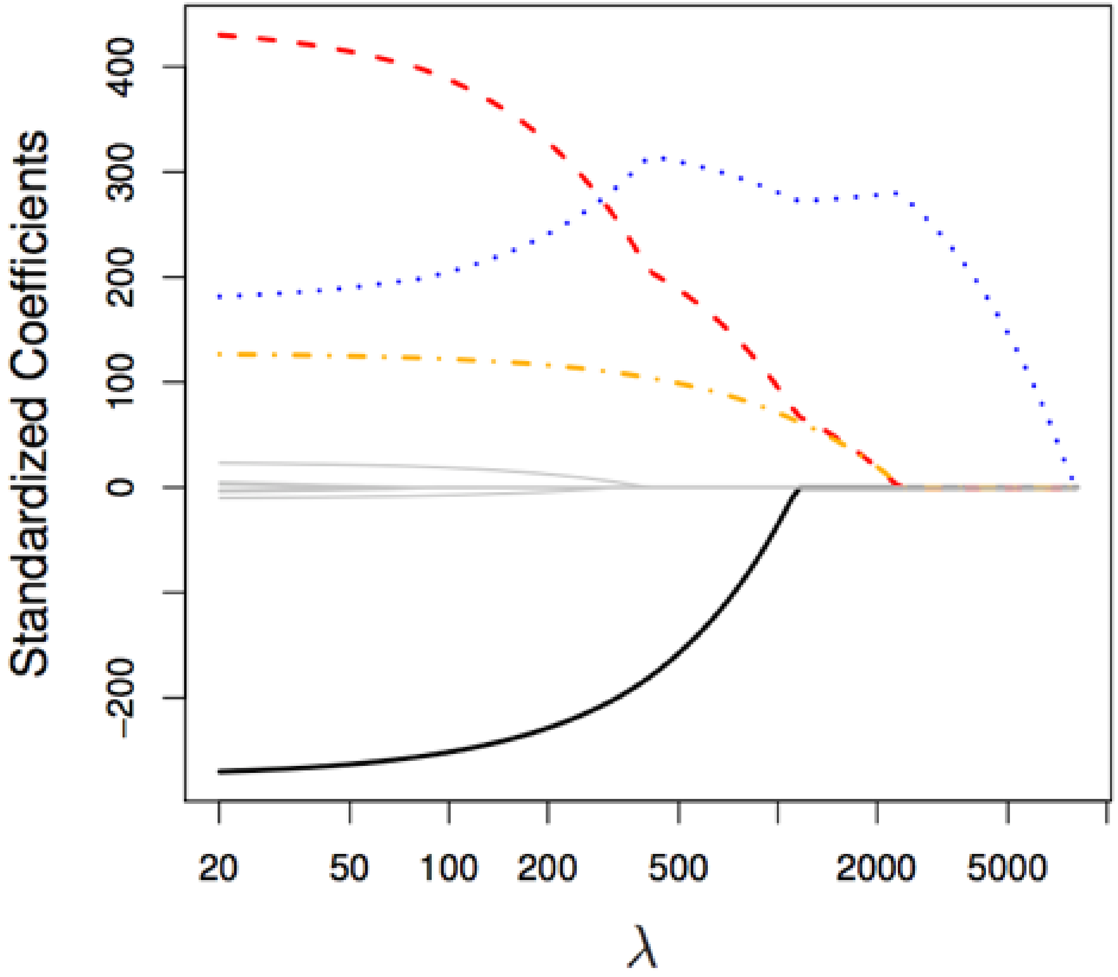
$$RSS + \lambda \sum_{j=1}^{p} |\beta_j|$$

- The lasso uses an ℓ_1 penalty instead of the ℓ_2 penalty.
  - The ℓ_1 norm of a coefficient vector $\boldsymbol{\beta}$ is $||\boldsymbol{\beta}||_1 = \sum_{j=1}^{p} |\beta_j|$

The Lasso

- 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 **feature selection** (in an embedded manner).
- We say that the lasso yields **sparse** models – that is, models that involve only a subset of variables.
- As in ridge regression, selecting a good value of λ for the lasso is critical; cross-validation is again the method of choice.

Example: Credit dataset



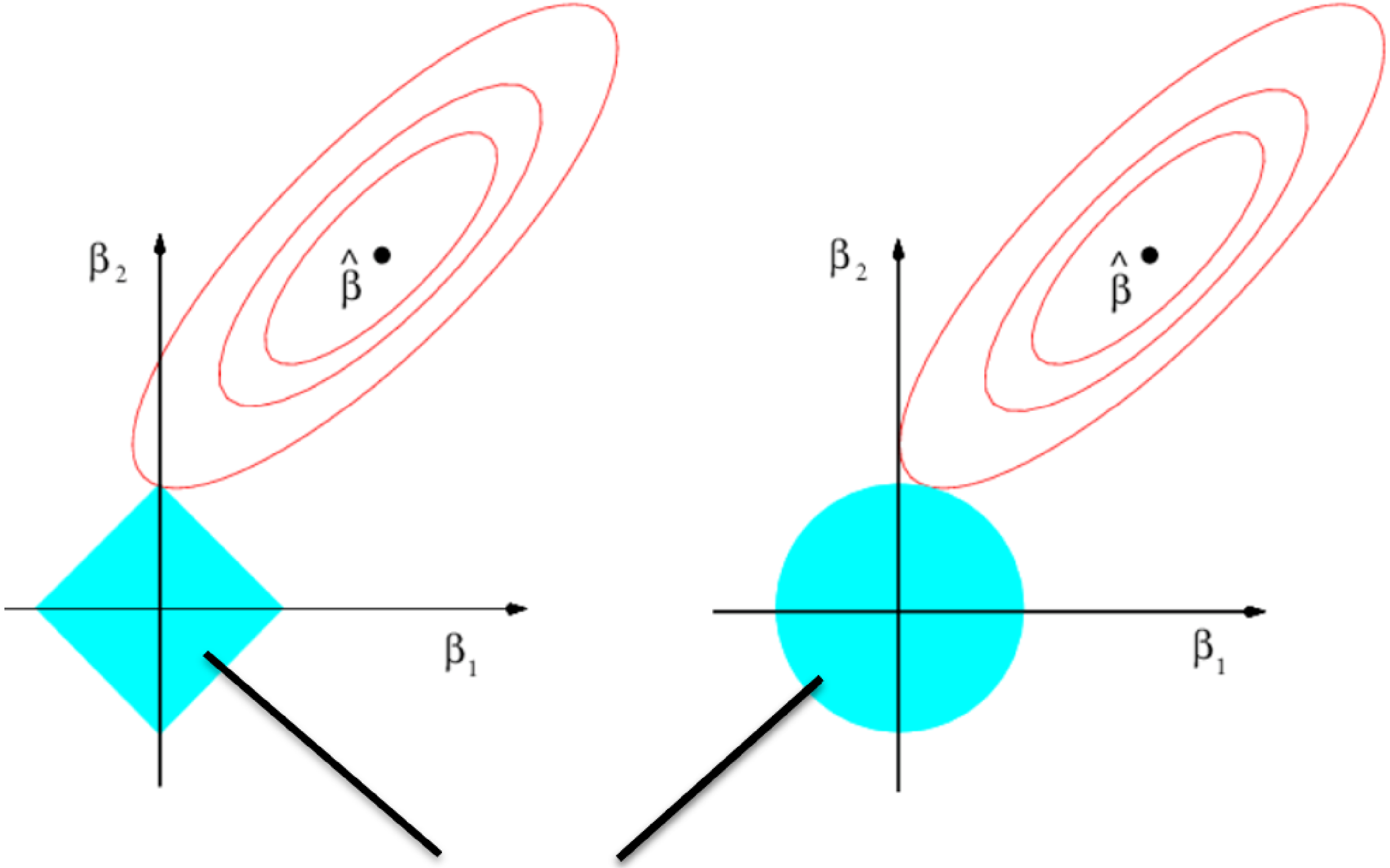
Variable selection property of the Lasso

- Why is it that the lasso, unlike 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

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n}(Y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij})^2 \qquad \text{subject to} \qquad \sum_{j=1}^{p} |\beta_j| \leq s.$$

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n}(Y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij})^2 \qquad \text{subject to} \qquad \sum_{j=1}^{p} \beta_j^2 \leq s.$$

Comparison of ℓ_1 and ℓ_2 constraints

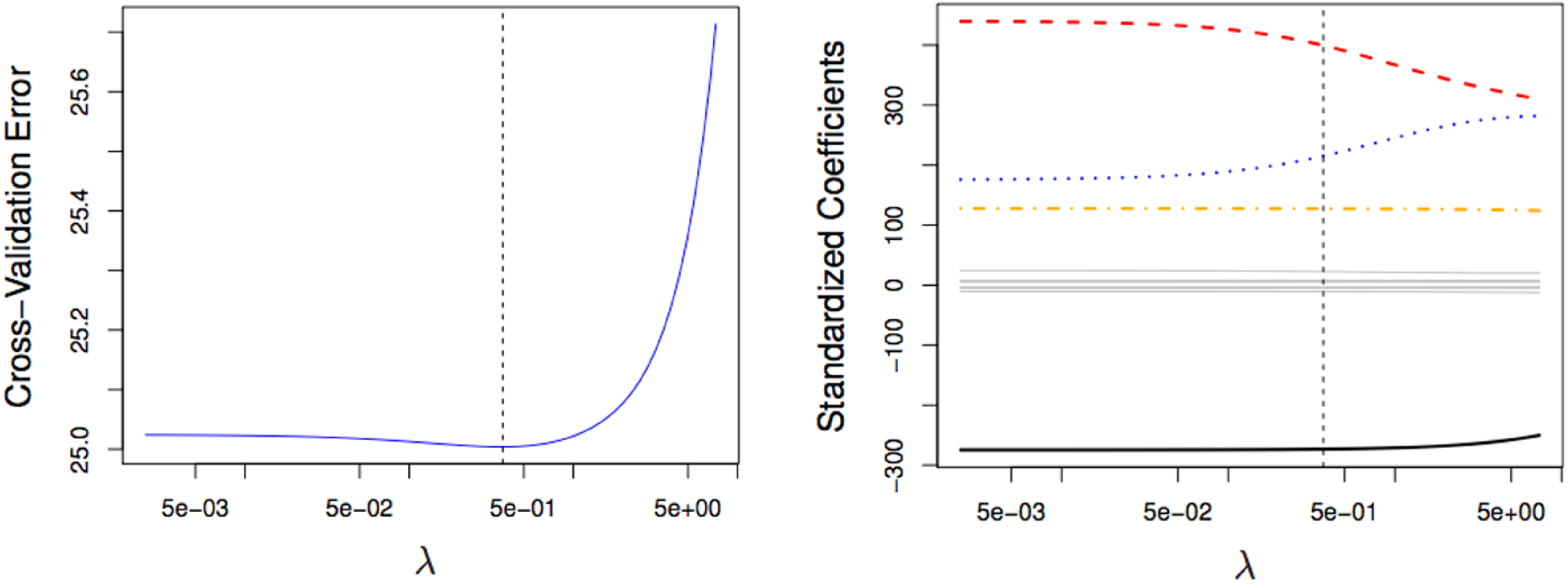


- Solution is feasible if it is within these blue regions for the Lasso (left) and Ridge (right) respectively.

Selecting the tuning parameters 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 the 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 illustrates cross-validation errors that result from applying ridge regression to the Credit data set with a range of λ values.
- Right illustrate the coefficient estimates as a function of λ. The vertical dashed lines indicate the best value of λ selected by cross-validation.

Elastic net in `glmnet`

- In `glmnet` (see Friedman, Hastie, and Tibshirani (2010)) the implementation is actually for a more general model called the Elastic net.
- It solves the following penalised minimization problem

$$\arg\min_{\boldsymbol{\beta}} ||Y - \boldsymbol{X}\boldsymbol{\beta}||_2^2 + \lambda((1-\alpha)/2||\boldsymbol{\beta}||_2^2 + \alpha||\boldsymbol{\beta}||_1)$$

- Can consider it a weighted combination (mixture) of ℓ_1 and ℓ_2 penalties.
- Objective function actually more general than this as the RSS term can be further weighted

References

Friedman, J., T. Hastie, and R. Tibshirani (2010). "Regularization Paths for Generalized Linear Models via Coordinate Descent". In: *Journal of Statistical Software* 33.1, pp. 1-22. URL: http://www.jstatsoft.org/v33/i01/.

James, G., D. Witten, T. Hastie, et al. (2013). *An introduction to statistical learning*. Vol. 112. Springer.