

START Tuesday 2/22/22 (Week 6, lecture 10)

Thursday 2/24/22 - went over related r-code to week 7
↳ regularization & (week 8, lecture 11).
↳ wrong (1/1)

Shrinkage / Regularization Methods¹

Thursday 2/17/22 (Week 5, lecture 9)

- went over R code (classification &)
(resampling &)

¹Based on materials in ISLR 6.2

- Recall the linear model

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

- Despite its simplicity, the linear model has distinct advantages in terms of its **interpretability** and often shows good **predictive performance**.
- Hence we discuss in this 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 squares?

- **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**.

coefficients will be biased towards 0.

- **Shrinkage:** we fit a model involving all p predictors, but the estimated coefficients are shrunk towards zero relative to the least squares estimates. This shrinkage (also known as **regularization**) has the effect of reducing variance and selecting variables.
- 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.

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

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

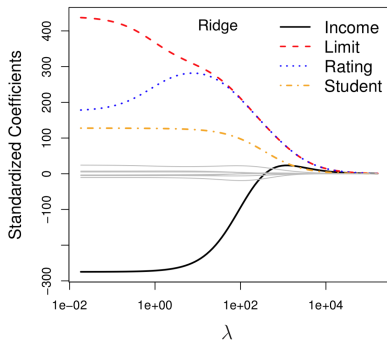
- In contrast, the ridge regression coefficient estimates $\hat{\beta}^R$ are the values that minimize

$$\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p \beta_j^2 = \text{RSS} + \lambda \|\beta\|_2^2,$$

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

- The notation $\|\beta\|_2$ denotes the ℓ^2 norm of a vector, and is defined as $\|\beta\|_2 = \sqrt{\sum_{j=1}^p \beta_j^2}$.

- As with least squares, ridge regression seeks coefficient estimates that fit the data well, by making the RSS small.
- However, the second term, $\lambda \sum_j \beta_j^2$, called a **shrinkage penalty**, is small when β_1, \dots, β_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.



- Each curve corresponds to the ridge regression coefficient estimate for one of the ten variables, plotted as a function of λ .

- The standard least squares coefficient estimates are **scale equivariant**: 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, $X_j \hat{\beta}_j$ 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

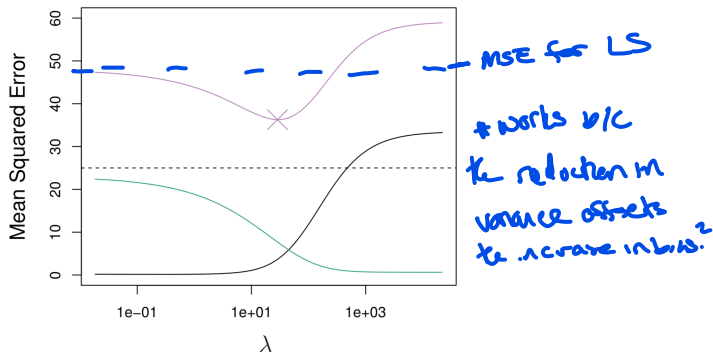
we first standardized our x vars.

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

↳ we don't have to standardize, really just need to make sure all the x_i are on the same scale. Standardization is just 1 way to do that.

Why Does Ridge Regression Improve Over Least Squares?

The Bias-Variance trade-off



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 . The horizontal dashed lines indicate the minimum possible MSE (irreducible error). The purple crosses indicate the ridge regression models for which the MSE is smallest.

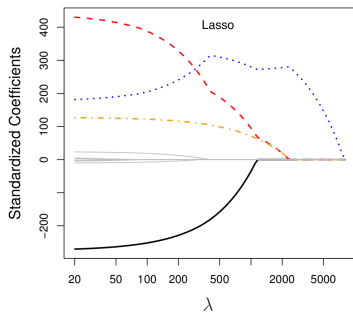
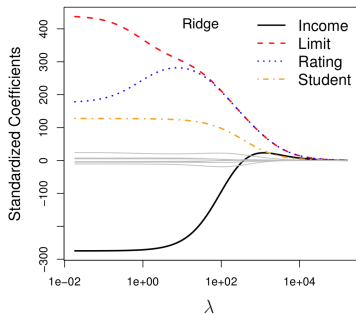
- Ridge regression does have one obvious disadvantage: it does not perform variable selection and include all p predictors in the final model, which is sometimes hard to interpret especially when p is large
- The **Lasso** is a relatively recent alternative to ridge regression that overcomes this disadvantage. The lasso coefficients, $\hat{\beta}_\lambda^L$, minimize the quantity

$$\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p |\beta_j| = \text{RSS} + \lambda \|\beta\|_1.$$

- 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 $\|\beta\|_1 = \sum_{j=1}^p |\beta_j|$.

- 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, 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.

Example: Credit dataset.



Why lasso results in coefficient estimates that are exactly equal to zero?

One can show that the lasso and ridge regression coefficient estimates solve the problems

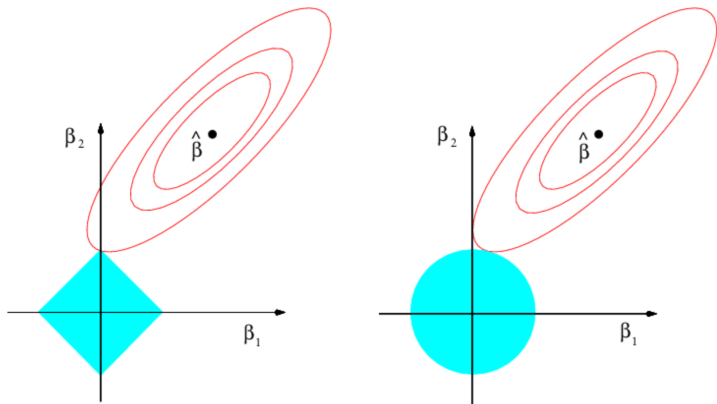
$$\text{minimize}_{\beta} \sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 \quad \text{subject to} \quad \sum_{j=1}^p |\beta_j| \leq s$$

and

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

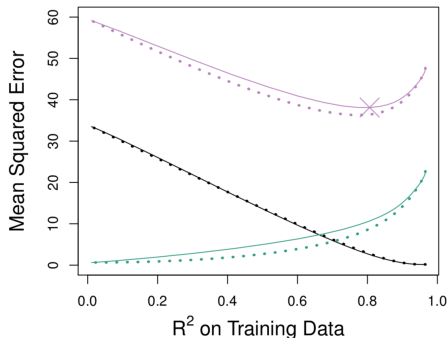
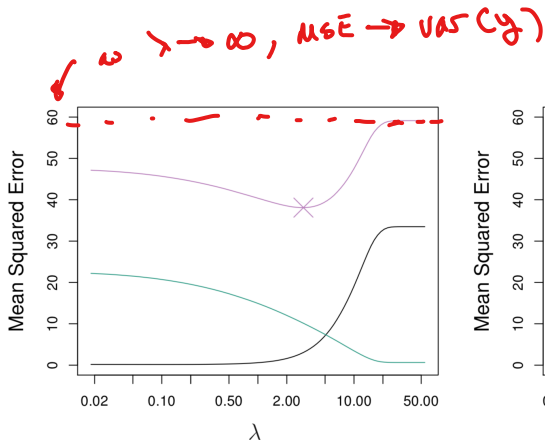
respectively.

The Lasso Picture



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.

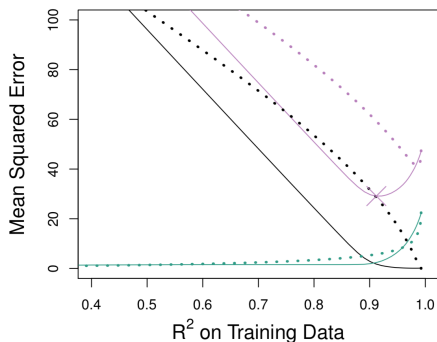
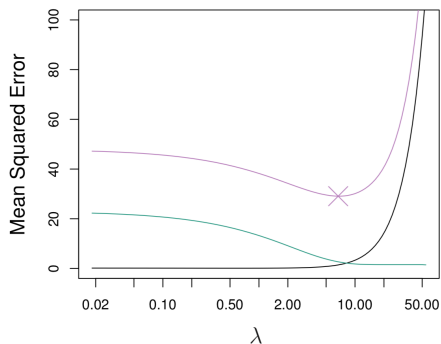
Comparing the Lasso and Ridge Regression



Left: Plots of squared bias (black), variance (green), and test MSE (purple) for the lasso on the simulated data we saw earlier.

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. The simulated data is similar to that we saw previously, 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 indexing. The crosses in both plots indicate the lasso model for which the MSE is smallest.

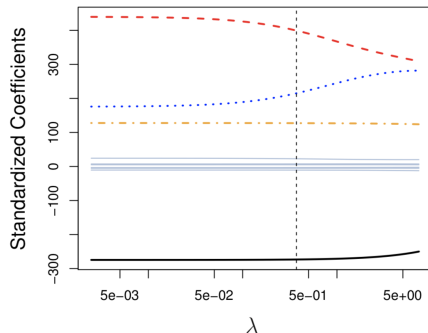
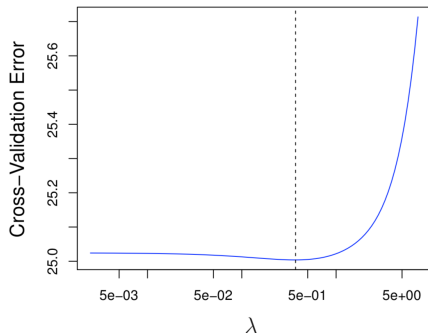


- These two examples illustrate that neither ridge regression nor the lasso will universally dominate the other.
- In general, one might expect the lasso to perform better when the response is a function of only a relatively small number of predictors.
- However, the number of predictors that is related 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.

- Ridge typically better at prediction, especially when covariates are correlated (typically the case).
- Lasso bad when covariates are correlated (multicollinearity)
 - ↳ elastic net - variation on lasso to deal w/rw.

- We need 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 dashed lines indicates the value of λ selected by cross-validation.

STOP 2/22/22 (Week 6 lecture 11)