Overview

- Bias-Variance Tradeoff

- Managing the Bias-Variance Tradeoff for Linear Regression
  - Subset predictors: Stepwise
  - Shrinkage/Regularization: Lasso, Ridge
  - Dimension Reduction: PCA (not covered today)

Bias-Variance Tradeoff

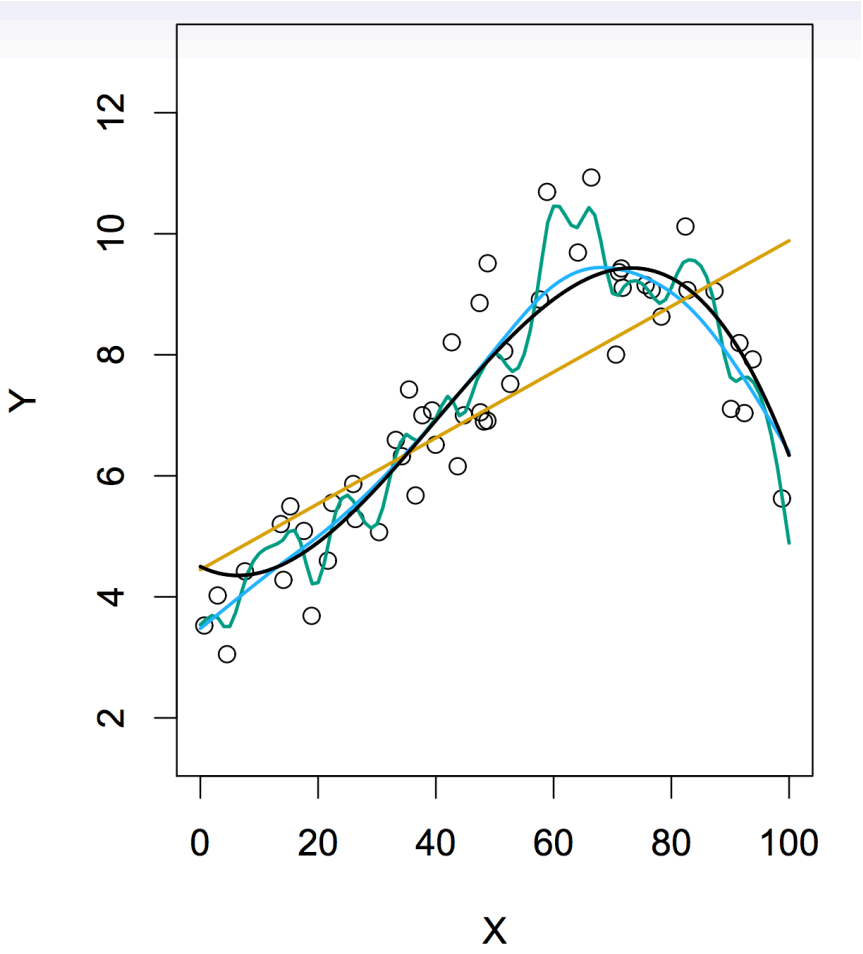
Suppose we have fit a model $\hat{f}(x)$ to some training data Tr, and let (x_0, y_0) be a test observation drawn from the population. If the true model is $Y = f(X) + \epsilon$ (with $f(x) = E(Y|X = x)$), then

$$E\left(y_0 - \hat{f}(x_0)\right)^2 = \text{Var}(\hat{f}(x_0)) + [\text{Bias}(\hat{f}(x_0))]^2 + \text{Var}(\epsilon).$$

Ok….what is going on here?

- Applies to modeling in general, beyond Linear Regression
- Want your model to minimize the expected test MSE on LHS. But how?
  - Var(ε), or "Irreducible Error". Can't do anything about that!
  - Can reduce Variance
  - Can reduce Bias

Bias-Variance Tradeoff



$$\mathrm{Var}(\hat{f}(x_0))$$

Amount by which $\hat{f}$ would change if estimated it using a different training dataset

$$\mathrm{Bias}(\hat{f}(x_0))] = E[\hat{f}(x_0)] - f(x_0)$$

Difference between expected prediction of our model and correct value we are trying to predict

Generally speaking, the *more flexible* the model, the *greater the variance*.

Managing the Bias-Variance Tradeoff with Linear Regression

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

I want to pare down my model, reducing the variance!

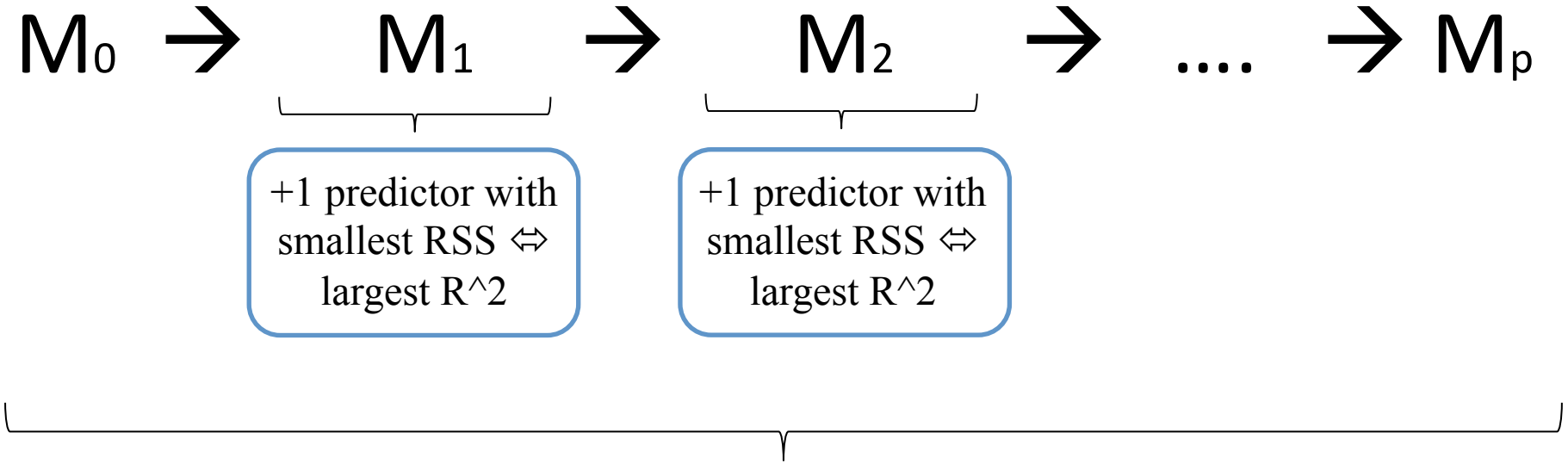
- **Subset selection** - choose subset of p predictors
- **Regularization** – keep p predictors, shrink coefficient estimates towards 0 (some variable selection for Lasso)
- **Dimension Reduction** – Project p predictors into M-dim space where M < p

Subset Selection

- Best subset: Try every model. Every possible combination of p predictors
  - Computationally intensive, especially for p large
  - Also, huge search space. Higher chance of finding models that look good on training data but have little predictive power on future data

- Stepwise
  - In practice, what people do!
  - Forward, Backward, Forward + Backward

Subset Selection - Forward Stepwise



Now we have *p* candidate models

Are RSS and R^2 good ways to decide amongst the *p* candidates?

Subset selection

Choosing among *p* candidate models…

- Cross-validation - always a great standby
- Mallow's C_p
- AIC
- BIC
- Adjusted R^2

Subset selection

Mallow's C_p:

$$C_p = \frac{1}{n}\left(\mathrm{RSS} + 2d\hat{\sigma}^2\right),$$

where d is the total # of parameters used and $\hat{\sigma}^2$ is an estimate of the variance of the error ϵ associated with each response measurement.

The *AIC* criterion is defined for a large class of models fit by maximum likelihood:

$$\mathrm{AIC} = -2\log L + 2 \cdot d$$

where L is the maximized value of the likelihood function for the estimated model.

Can show AIC and Mallow's Cp are equivalent for linear case

Subset selection

$$\text{BIC} = \frac{1}{n}\left(\text{RSS} + \log(n) d \hat{\sigma}^2\right)$$

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 observations.

Since $\log n > 2$ for any $n > 7$, the BIC statistic generally places a heavier penalty on models with many variables

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

Unlike the R^2 statistic, the adjusted R^2 statistic *pays a price* for the inclusion of unnecessary variables in the model.

Managing the Bias-Variance Tradeoff with Linear Regression

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

I want to pare down my model, reducing the variance!

- **Subset selection** - choose subset of p predictors
- **Regularization** – keep p predictors, shrink coefficient estimates towards 0 (some variable selection for Lasso)
- **Dimension Reduction** – Project p predictors into M-dim space where M < p

Regularization – Ridge regression

- Recall that the least squares fitting procedure estimates $\beta_0, \beta_1, \ldots, \beta_p$ 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

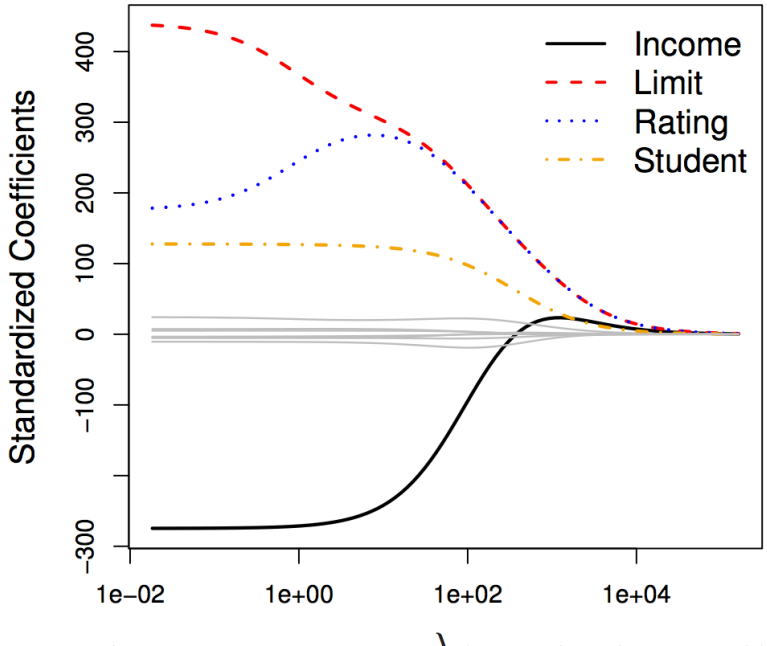
$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = \text{RSS} + \lambda \sum_{j=1}^{p} \beta_j^2,$$

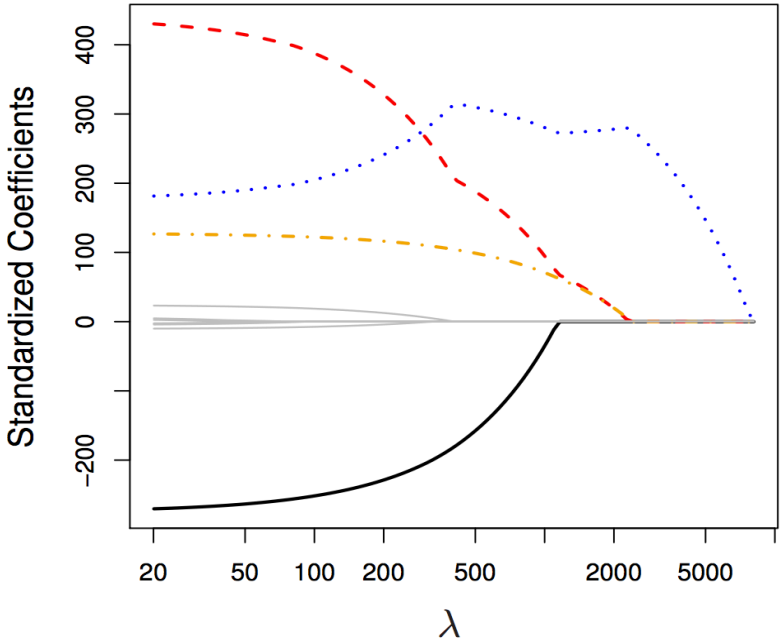
where $\lambda \geq 0$ is a *tuning parameter*, to be determined separately.

Regularization – Lasso regression

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

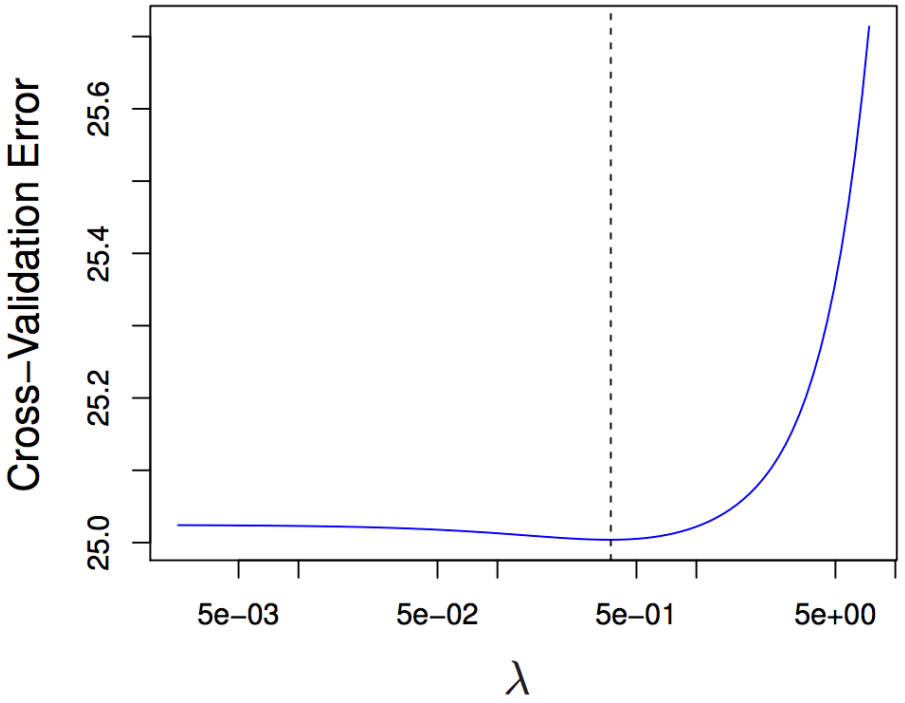
Ridge vs. Lasso

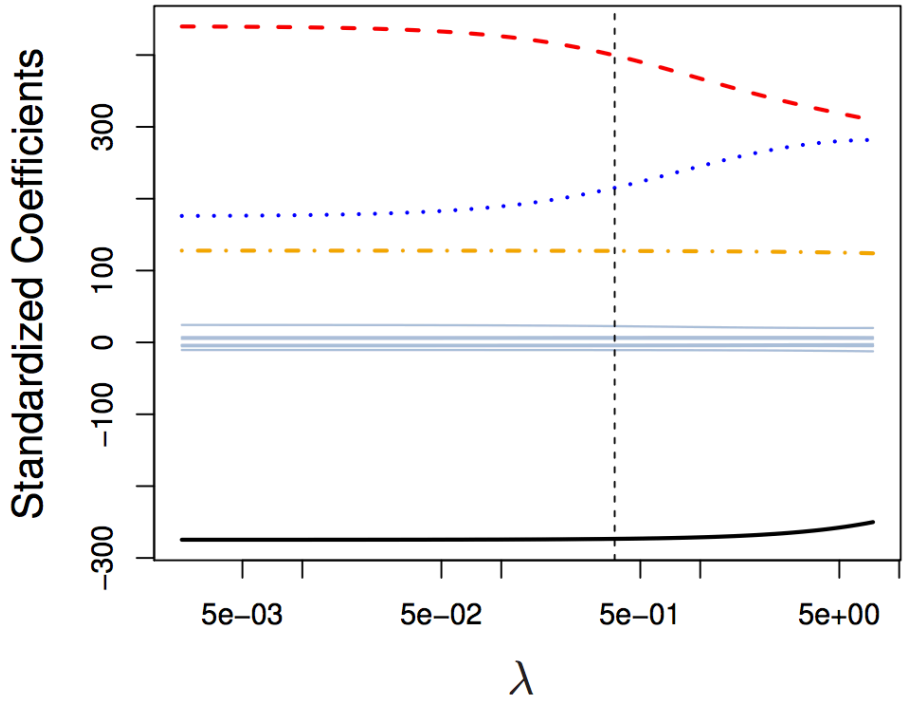




- When $\lambda = 0$, we simply have linear models.
- As λ increases, both models become less flexible, reducing variance, but increasing bias.
- Lasso has the advantage of variable selection as well (especially nice when p is large)
- Neither universally dominate, but in general one might expect Lasso to do better when response is function of relatively few predictors.
  - Of course you never actually know this, so use your friend, cross-validation!

Choosing λ





- Just increment λ along, fit a large number of models per increment, and choose λ which minimizes cross-validated error, and voila! You have your corresponding optimized model for Ridge Regression.

Don't forget….

- 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 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

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