Machine Learning and Data Mining
Ordinary Regression & Ridge Regression

Patricia Hoffman, PhD

Most Slides taken from

Stanford Professor Robert Tibshirani's

Statistic 315a Course

Slides also taken from

Stanford Professor Trevor Hastie

Regression Resources
Main Sources

- Stanford Professor Robert Tibshirani
  - http://www-stat.stanford.edu/~tibs/stat315a/LECTURES/chap3.pdf
- Stanford Professor Trevor Hastie
  - http://www.stanford.edu/~hastie/index.html
  - http://www-stat.stanford.edu/~tibs/stat315a/LECTURES/linear.pdf
- Patricia Hoffman, PhD Section 1.2
  - RoadmapW1stChapter.pdf

More Resources

- Introduction to Data Mining
  - By Tan, Steinbach, Kumar
  - Appendix D & Section 5.8
- Elements of Statistical Learning
  - By Hastie, Tibshirani, Friedman
  - http://www-stat.stanford.edu/~tibs/ElemStatLearn/
  - Chapters 3 & 4

Project Suggestions

- Project Paper Suggestion: Elastic Net
  - Fast Algorithm for combination of
    - Ridge Regression & Lasso Regression
  - http://www.jstatsoft.org/v33/i01/paper
  - http://www.sfbayacm.org/event/advances-regularization-bridge-regression-and-coordinate-descent-algorithms
- R Package glmnet implements Elastic Net
  - http://cran.r-project.org/web/packages/glmnet/glmnet.pdf
- Basis Expansions
  - ESL Hastie, Tibshirani, Friedman
    - Chapter 5 Second Edition
    - Polynomials and Splines
  - Example Code in lmvsridgeConcrete.R

Regression – Example Code

- MyFirstRLesson .R & UserDefnFunction.R
  - Solve Ax = b
  - Simple Example of R function lm
  - Scale Function
- lmvsridgeSonarData.R
  - Illustrates Ordinary Linear Regression – lm
  - Compared with Ridge Regression – lm.ridge
- MulticlassRegressionIrisData.R
  - Use Regression on a Multi-classification Problem
- basicExpansion.R

Preliminaries

Data $(x_1, y_1), \ldots (x_N, y_N)$.

x_i is the predictor (regressor, covariate, feature, independent variable)

y_i is the response (dependent variable, outcome)

We denote the *regression function* by

$$f(\mathrm{X}) = \mathrm{E}\,(Y|x)$$

This is the conditional expectation of Y given x.

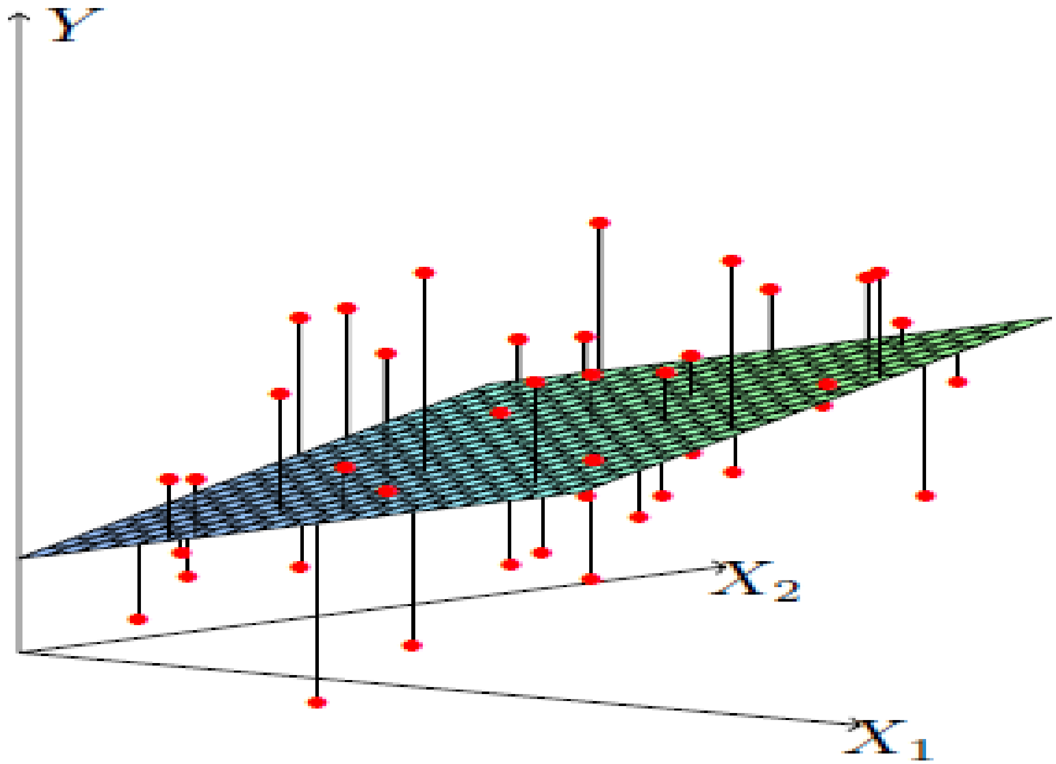
The linear regression model assumes a specific linear form for $f(\mathbf{X})$

$$f(\mathbf{X}) = \beta_0 + \Sigma_j\, \mathbf{X}\beta_j$$

which is usually thought of as an approximation to the truth.

http://www-stat.stanford.edu/~tibs/stat315a/LECTURES/chap3.pdf

Linear Regression



Elements of Statistical Learning Figure 3.1

Goal – Find the Best Betas

Find the betas which will make the difference between $f(x_i)$ and y_i the smallest where

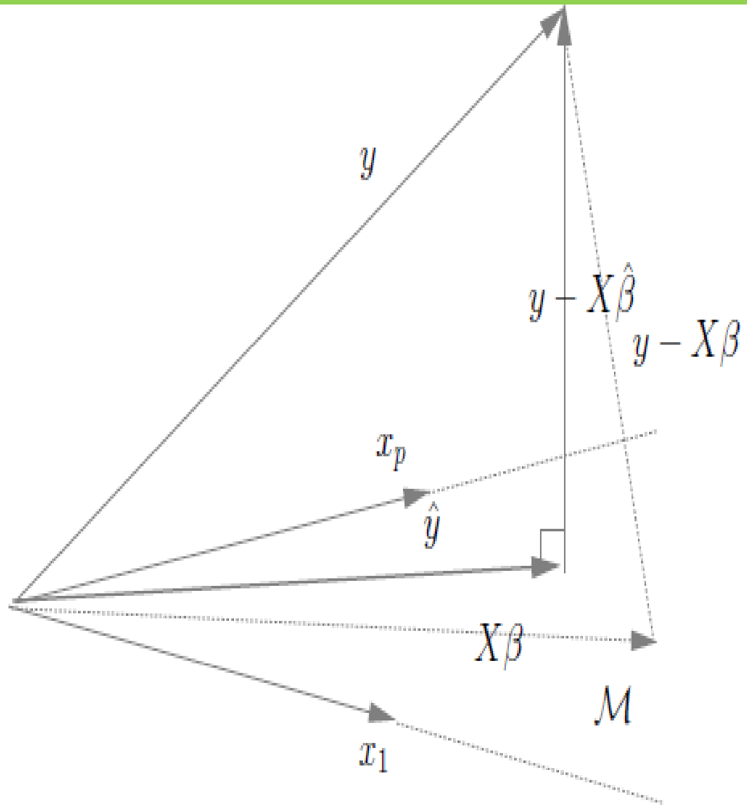
$$f(\mathbf{x}) = \beta_0 + \Sigma_j\, \mathbf{x}\beta_j$$

That is minimize the root mean square error

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

Regular Calculus: Take the derivative, set it equal to zero, and solve for the betas
Solution is in Appendix D of our text

Geometry of Least Squares



$\hat{y} = X\hat{\beta}$ is the *orthogonal projection* of y onto the subspace $\mathcal{M} \subset \mathbb{R}^n$ spanned by the columns of X. This is true even if X is not of full column rank.

Proof: Pythagoras.

$$y - \hat{y} \perp \mathcal{M}$$
$$\Updownarrow$$
$$(y - X\hat{\beta}) \perp x_j \; \forall j \quad (x_j \text{ is a column of } X \text{ here})$$
$$\Updownarrow$$
$$X^T(y - X\hat{\beta}) = 0$$

$$\hat{\beta} = (X^T X)^{-1} X^T \mathbf{y}$$

See Figure 3.1 of ESL Text

http://www-stat.stanford.edu/~tibs/stat315a/LECTURES/linear.pdf

Ordinary Least Squares Normal Equation & Solution

Find best function f (where $\mathbf{X}$ is the input matrix of N observations of p factors)

$$f(\mathbf{X}) = \beta_0 + \sum_j \mathbf{X}\beta_j$$

Oservations: $\mathbf{X}^T = (\mathbf{X}_1, \mathbf{X}_2, ...\mathbf{X}_p)$

Regression Coef: $\widehat{\beta} = (\beta_0 ... \beta_p)$

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

Solution:

$$\widehat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Note: β_0 is included in the beta vector and the constant value 1 is added to $\mathbf{X}$. That is $x_{i0} = 1$ for all i and the matrix $\mathbf{X}$ is a $N \times p + 1$ matrix

Ridge Regression

λ penalizes the sum-of-squares of the parameters. $\lambda \geq 0$ is the complexity parameter which controls shrinkage.
$\lambda = 0 \Rightarrow$ solution is the same as regular regression.
If $\lambda \to \infty$, then $\beta_{j=1...p} \to 0$ and the solution is the average $\mathbf{y}$

Minimize the following:

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

Solution:

$$\widehat{\beta}_\lambda(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y} \text{ where } \mathbf{I} \text{ is the identity matrix}$$

Ridge Regression - Solution Normal Equation

$\lambda \geq 0$ complexity parameter controls shrinkage.

$\lambda = 0 \Rightarrow$ solution is the same as regular regression.

λ penalizes the sum-of-squares of the parameters.

if $\lambda \longrightarrow \infty$, then $\hat{\beta} \longrightarrow 0$ the solution becomes the average y

Minimize the following:

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

The Solution:

$$\hat{\beta}_\lambda = (X^T X + \lambda I)^{-1} X^T \mathbf{y}$$

Compare OLS with Ridge Regression

Ordinary Linear Regression

$$\widehat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Ridge Regression

$$\hat{\beta}_\lambda = (X^TX + \lambda I)^{-1}X^T\mathbf{y}$$

The lambda parameter is used to control the fit of the model

Ridge Regression Benefits

- Over fitting results in models that are more complex than necessary
- Ridge Regression provides a tuning parameter lambda which is used to adjust the fit of the model to the data
- Use cross validation to find the best value for lambda

Model Complexity

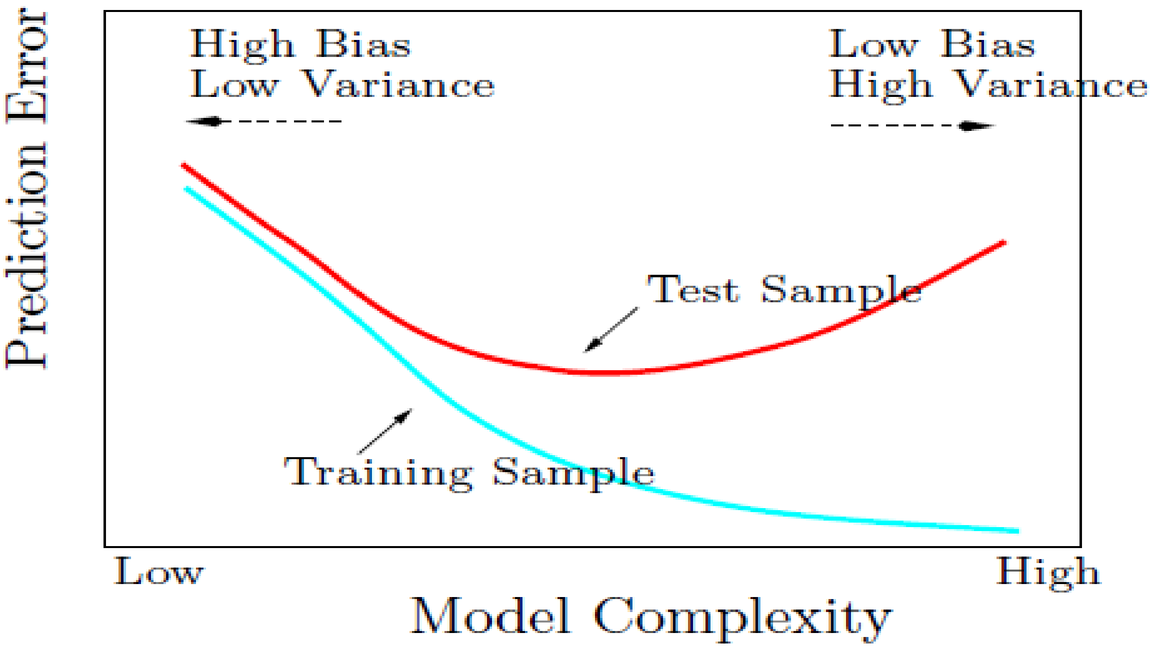


FIGURE 2.11. *Test and training error as a function of model complexity.*

From Elements of Statistical Learning by Hastie, Tibshirani, Friedman

lmvsridgeSonarData.R Example Code

Linear Regression

```
lmSonar <- lm(V61~., data = sonarTrain)
```

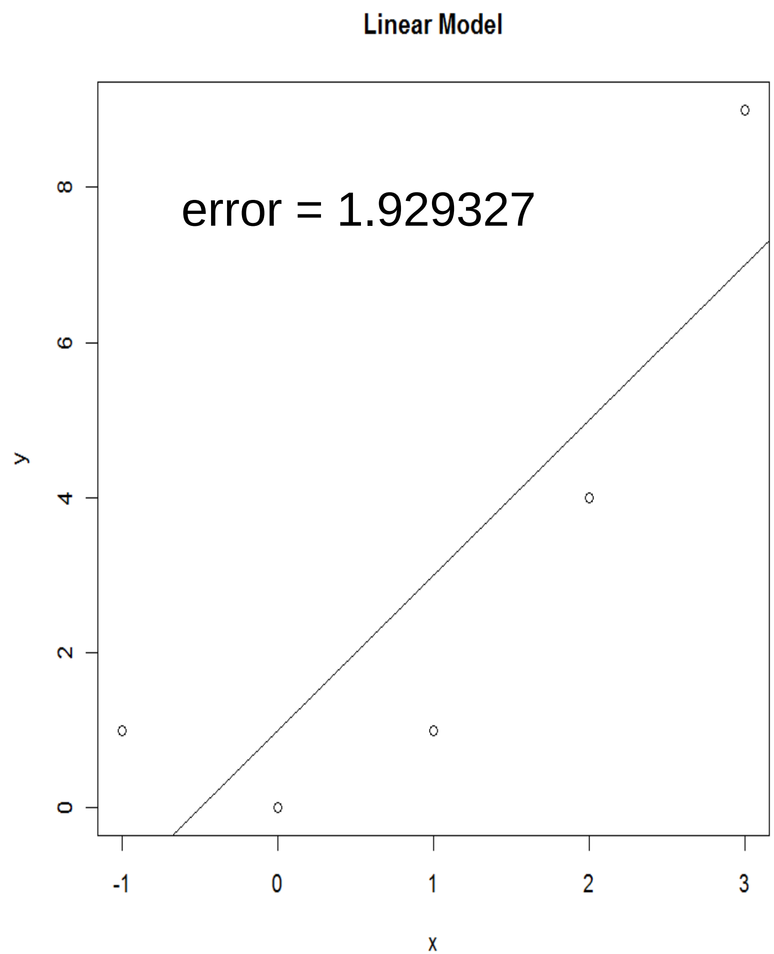
Ridge Regression

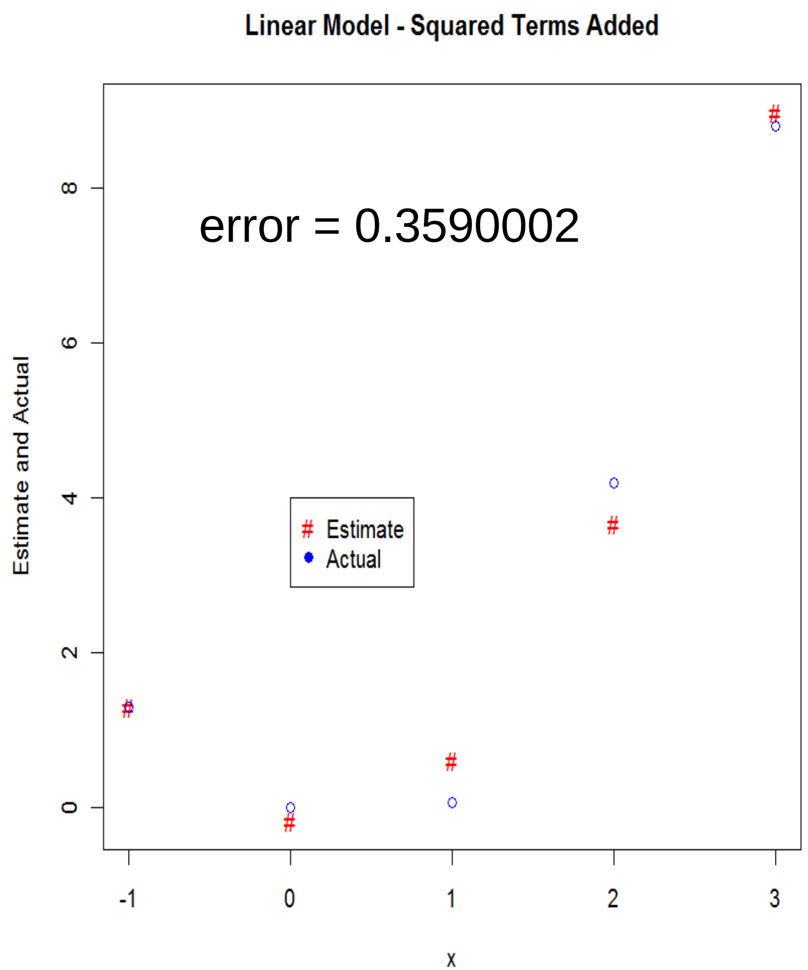
```
lmSonar <- lm.ridge(V61~., data = SonarIn)
```

Note: ridge regression does not have a predict function in R

Basis Expansion

(basisExpansion.r)





Basis Expansion

(basisExpansion.r)

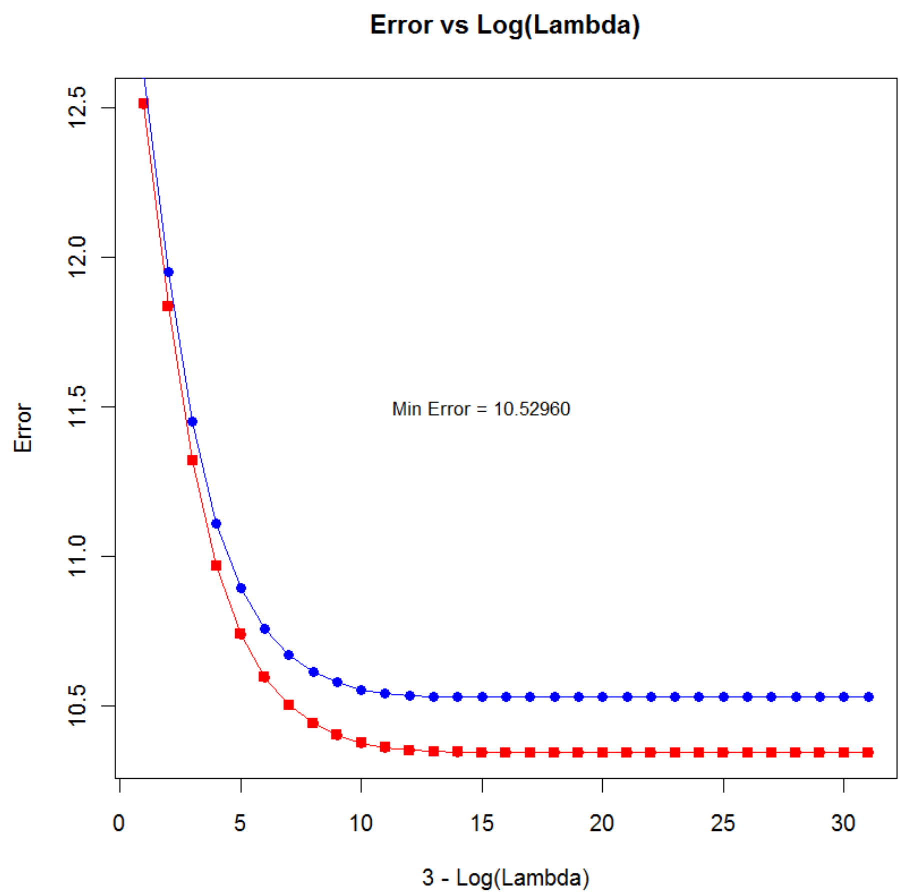
error = 1.929327

x	y
-1	1.30
0	0.00
1	0.07
2	4.20
3	8.80

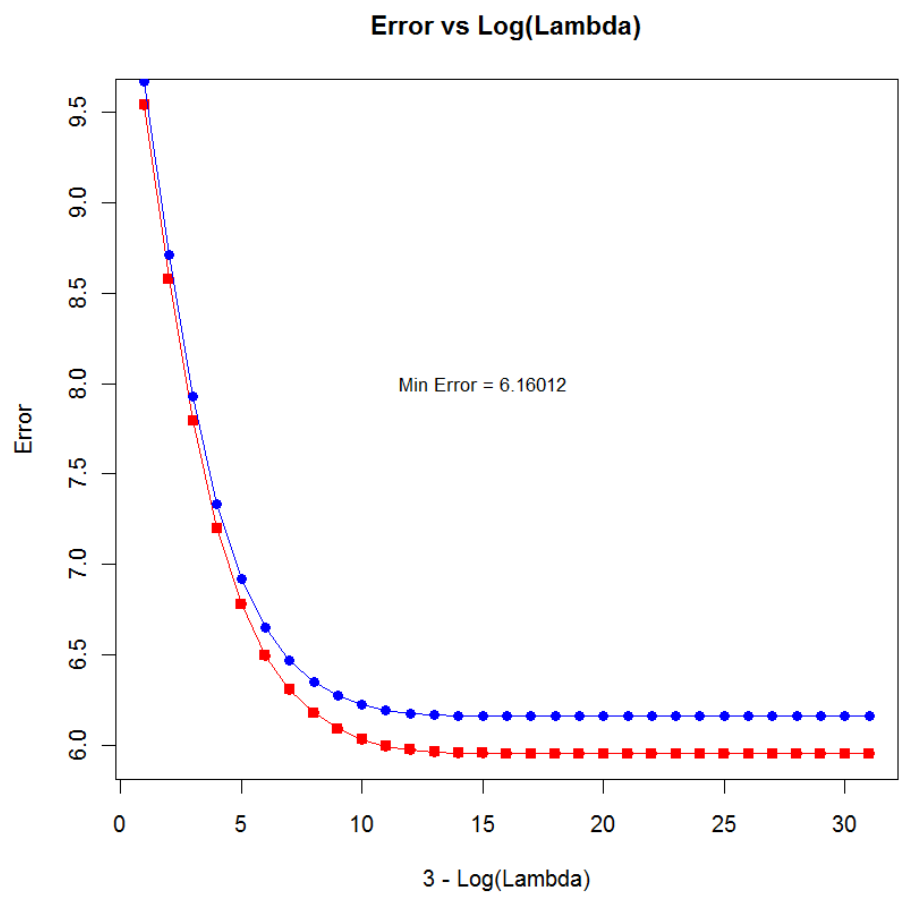
error = 0.3590002

x	x2	y
-1	1	1.30
0	0	0.00
1	1	0.07
2	4	4.20
3	9	8.80

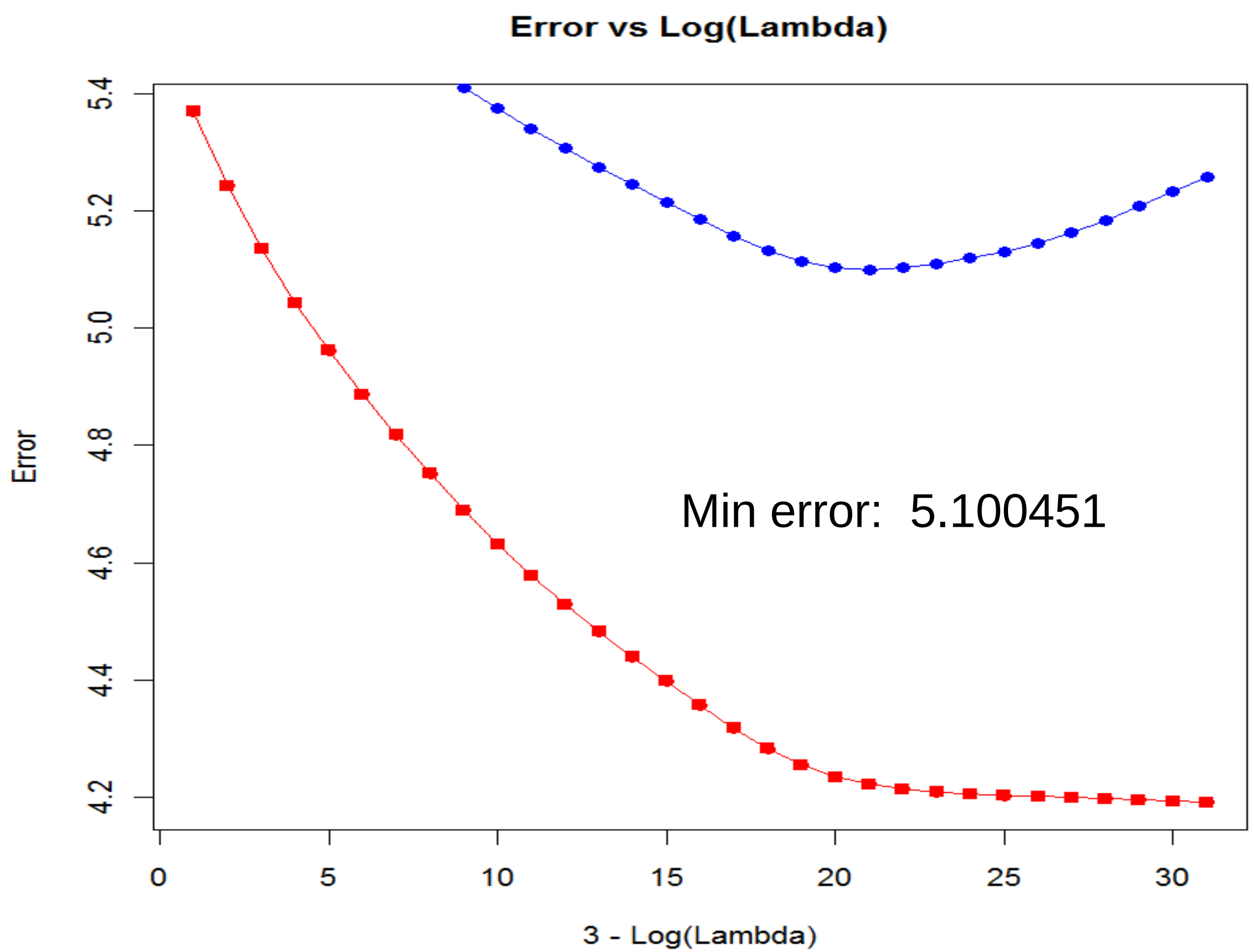
Ridge Regression Linear Terms Concrete Data



Ridge Regression Basis Expansion Spline Terms Concrete Data



Ridge Regression Basis Expansion 4th Degree Polynomial Concrete Data



Multiclass Problem

The Target Y = {y1, y2, … , yk} has multiple values – Iris Data is Example

- One-Against-Rest (1-r)
  - K binary classifiers, one for each yi in Y
  - yi is the positive example – rest are negative examples

- One-Against-One (1-1)
  - K(K-1)/2 binary classifiers
  - each classifier distinguishes between a pair of classes (yi, yj)
  - instances that do not belong to either yi or yj are ignored

Test Instances Classified
- combine predictions
- from binary classifiers
- voting scheme or probability estimate

Iris Data: 50 samples from each of three species Setosa, Versicolor, Virginica

5 columns of data:
sepal length, sepal width, petal length, petal width, species

[Iris Species - Versicolor]

MulticlassRegressionIrisData.R Example Code

Y1 has ones in the first 50 entries and -1 in the rest: indicates Setosa

Y2 has ones in entries 51 to 100 and -1 in the rest: indicates Versicolor

Y3 has ones in entries 101 to 150 and -1 in the rest: indicates Virginica

Use Ridge Regression to create 3 models

Homework 03 Problem 3

Discussion

Project

- Spend Time talking about Project Ideas