Final Project

MAP4191
Christopher Prats

Setting up the project

- The first data set (regression) was selected
- The data was processed into the R project by utilizing the read.table function.
- A variable was created to join the X and Y lists into a single data frame. This simplified the process of performing regression on the data.

```
# Load required libraries
library(glmnetUtils)

# Import data into list
X ← read.table("data/X.txt")
Y ← read.table("data/y.txt")$V1
XY ← as.data.frame(cbind(Y, X))
```

Performing Linear/Multiple regression

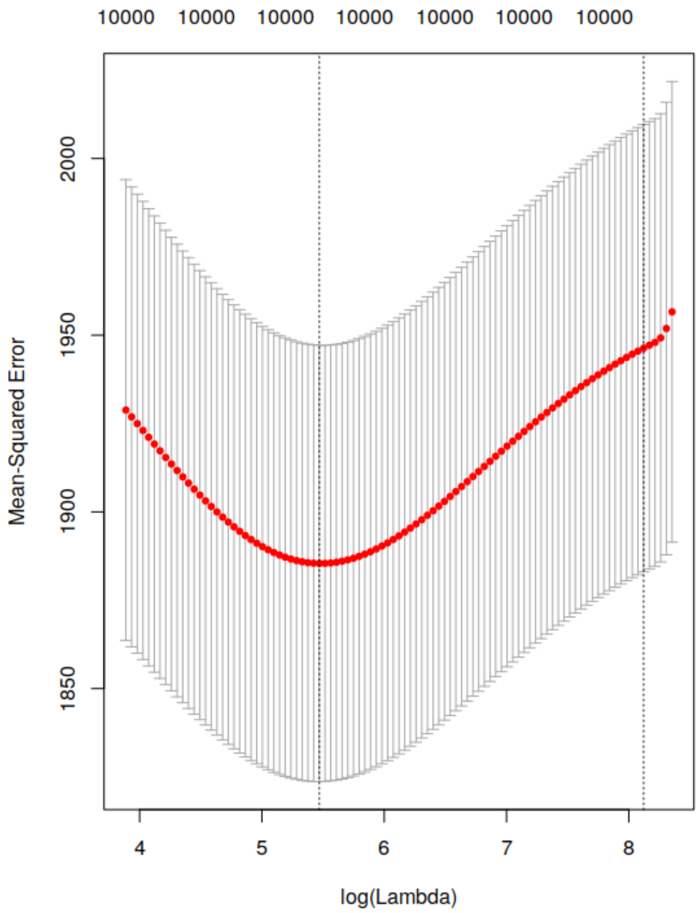
- Linear regression was performed using the lm function.
- I faced some difficulty utilizing this function with the data imported using read.table, so I utilized the XY data frame as the data for the model.
  - linear_regression <- lm(Y ~ ., data = XY)
- Some of the resulting coefficients are as follows:

(Intercept)	57.52873459
V1	-5.14389866
V2	35.57618227
V3	23.92985387
V4	18.06831795
V5	4.12318841
V6	-57.01675300
V7	-29.93890019
V8	81.15356165
V9	-2.28073956
V10	27.05541577

Performing Ridge Regression

- Ridge regression was done utilizing the glmnet and cv.glmnet functions found in the glmnetUtils library.
- I first performed ridge regression without finding the optimal lamda utilizing the following R code.
  - ridge_regression <- glmnet(Y ~ ., data = XY, alpha = 0)
- I then found the optimal lamba (the value for λ that results in the smallest MSE) utilizing the following R code.
  - cv_ridge_model <- cv.glmnet(as.matrix(X), as.matrix(Y), alpha = 0)
  - lambda_ridge <- cv_ridge_model$lambda.min

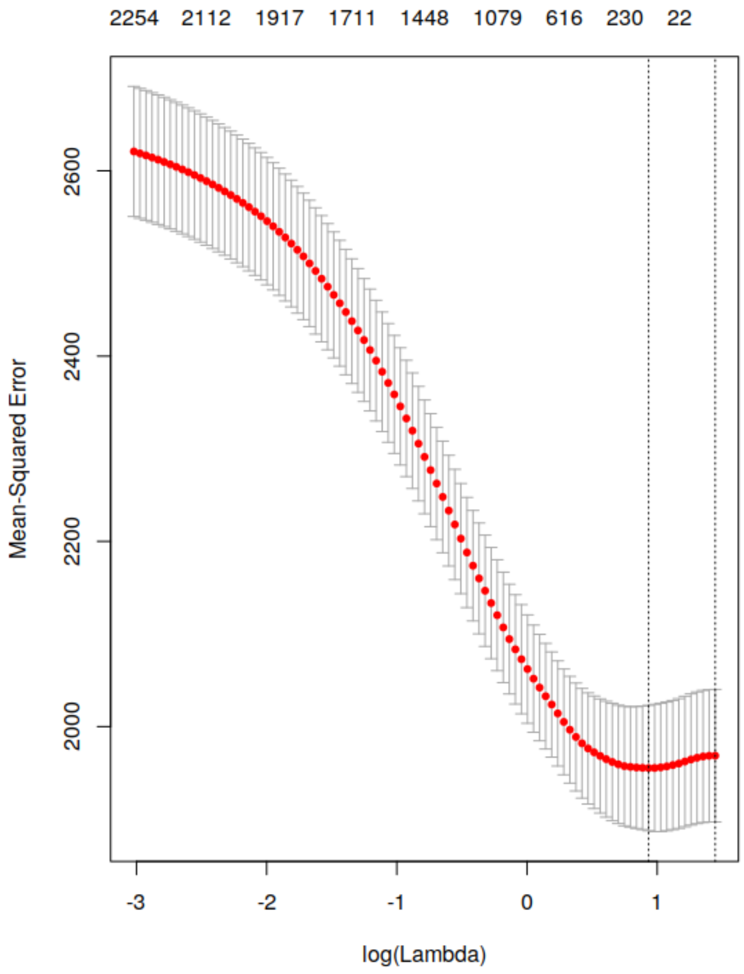
λ Plot


10000 10000 10000 10000 10000 10000 10000
Mean-Squared Error
1850
1900
1950
2000
4
5
6
7
8
log(Lambda)

Performing Lasso Regression

- The process was largely similar to that used in ridge regression. The only difference was alpha is equal to 1 in lasso, and 0 in ridge. Lasso regression was done utilizing the glmnet and cv.glmnet functions found in the glmnetUtils library.
- I first performed lasso regression without finding the optimal lamda utilizing the following R code.
  - lasso_regression <- glmnet(Y ~ ., data = XY, alpha = 1)
- I then found the optimal lamba (the value for λ that results in the smallest MSE) utilizing the following R code.
  - cv_lasso_model <- cv.glmnet(as.matrix(X), as.matrix(Y), alpha = 1)
  - lambda_lasso <- cv_lasso_model$lambda.min

λ Plot


2254 2112 1917 1711 1448 1079 616 230 22
2600
2400
2200
2000
Mean-Squared Error
-3 -2 -1 0 1
log(Lambda)

Evaluating the Models

- The models were evaluated using k-fold cross-validation.
- 5 folds were used.
- I used the train function in the caret library to evaluate this.
- Lasso regression gave the largest R^2 and was the best performing model.