Final Project

Samuel Harris

11/30/2016

Run 1

Preliminaries

```
grocery <- data.frame(read.csv("GrocerySales.csv"))
head(grocery)
```

```
##   store     brand week  logmove feat price     AGE60      EDUC    ETHNIC
## 1     2 tropicana   40 9.018695    0  3.87 0.2328647 0.2489349 0.1142799
## 2     2 tropicana   46 8.723231    0  3.87 0.2328647 0.2489349 0.1142799
## 3     2 tropicana   47 8.253228    0  3.87 0.2328647 0.2489349 0.1142799
## 4     2 tropicana   48 8.987197    0  3.87 0.2328647 0.2489349 0.1142799
## 5     2 tropicana   50 9.093357    0  3.87 0.2328647 0.2489349 0.1142799
## 6     2 tropicana   51 8.877382    0  3.87 0.2328647 0.2489349 0.1142799
##     INCOME   HHLARGE   WORKWOM   HVAL150 SSTRDIST  SSTRVOL CPDIST5
## 1 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 2 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 3 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 4 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 5 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 6 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
##     CPWVOL5
## 1 0.3769266
## 2 0.3769266
## 3 0.3769266
## 4 0.3769266
## 5 0.3769266
## 6 0.3769266
```

Data Preperation

```
x <- model.matrix ( logmove ~
 log(price) *
(feat + brand + AGE60 + EDUC + ETHNIC +
INCOME + HHLARGE + WORKWOM + HVAL150 + SSTRDIST +
SSTRVOL + CPDIST5 + CPWVOL5)^2, data=grocery)

dim(x)
```

```
## [1] 28947   210
```

```
x=x[,-1]
dim(x)
```

```
## [1] 28947   209
```

Standardizing the Data

```
scaled.x <- scale(x)
```

Splitting the Data

```
set.seed(1)
nData = nrow(scaled.x)
samples = sample(1:nData, 20000, replace=FALSE)
training = data.frame(scaled.x[samples,])
testing = data.frame(scaled.x[-samples,])

length(training[,1])

## [1] 20000

length(testing[,1])

## [1] 8947

logmovetraining = grocery$logmove[samples]
logmovetesting = grocery$logmove[-samples]
```

Linear Model

```
set.seed(1)

lmfit = lm(logmovetraining ~ ., data = training)
prediction = predict(lmfit, testing)
mean((prediction - logmovetesting)^2)

## [1] 0.3583494
```

Ridge Regression

```
set.seed(1)

trainingmatrix = model.matrix(logmovetraining ~ ., data = training)
testingmatrix = model.matrix(logmovetesting ~ ., data = testing)
grid = 10^seq(4,-2,length=100)

cv.outridge = cv.glmnet(trainingmatrix, logmovetraining, alpha = 0)

ridge.mod=glmnet(trainingmatrix, logmovetraining, alpha=0, lambda=grid, thres
h=1e-12)

bestlambdaridge = cv.outridge$lambda.min

predictionridge = predict(cv.outridge, s=bestlambdaridge, newx = testingmatri
x)
```

```r
msebest = mean((predictionridge-logmovetesting)^2)

ridge.pred=predict(ridge.mod,s=1e10,newx = testingmatrix)
mseinf = mean((ridge.pred-logmovetesting)^2)

ridge.pred=predict(ridge.mod,s=0,newx = testingmatrix)
mse0 = mean((ridge.pred-logmovetesting)^2)

bestlambdaridge

## [1] 0.05995981

mseinf

## [1] 1.049705

mse0

## [1] 0.3781809

msebest

## [1] 0.3906986
```

Lasso Regression

```r
set.seed(1)

lasso.mod=glmnet(trainingmatrix, logmovetraining, alpha=1, lambda=grid, thres
h=1e-12)

cv.outlasso = cv.glmnet(trainingmatrix, logmovetraining, alpha = 1)

bestlambdalasso = cv.outlasso$lambda.min

predictionlasso = predict(cv.outlasso, s=bestlambdalasso, newx = testingmatri
x)

msebest = mean((predictionlasso-logmovetesting)^2)

lasso.pred=predict(lasso.mod,s=1e10,newx = testingmatrix)
mseinf = mean((lasso.pred-logmovetesting)^2)

lasso.pred=predict(lasso.mod,s=0,newx = testingmatrix)
mse0 = mean((lasso.pred-logmovetesting)^2)

bestlambdalasso

## [1] 0.0001831086

mseinf
```

```
## [1] 1.051729

mse0

## [1] 0.4043329

msebest

## [1] 0.3697927

best.lasso.mod = glmnet(trainingmatrix, logmovetraining, alpha = 1, lambda =
bestlambdalasso, thresh=1e-12)
coefBestLasso = coef(best.lasso.mod)
sum(coefBestLasso != 0)

## [1] 138

sum(coefBestLasso == 0)

## [1] 73
```

Run 2

Preliminaries

```
rm(list = ls())
grocery <- data.frame(read.csv("GrocerySales.csv"))
head(grocery)

##   store     brand week  logmove feat price     AGE60      EDUC    ETHNIC
## 1     2 tropicana   40 9.018695    0  3.87 0.2328647 0.2489349 0.1142799
## 2     2 tropicana   46 8.723231    0  3.87 0.2328647 0.2489349 0.1142799
## 3     2 tropicana   47 8.253228    0  3.87 0.2328647 0.2489349 0.1142799
## 4     2 tropicana   48 8.987197    0  3.87 0.2328647 0.2489349 0.1142799
## 5     2 tropicana   50 9.093357    0  3.87 0.2328647 0.2489349 0.1142799
## 6     2 tropicana   51 8.877382    0  3.87 0.2328647 0.2489349 0.1142799
##     INCOME   HHLARGE   WORKWOM   HVAL150 SSTRDIST  SSTRVOL CPDIST5
## 1 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 2 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 3 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 4 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 5 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
## 6 10.55321 0.1039534 0.3035853 0.4638871 2.110122 1.142857 1.92728
##     CPWVOL5
## 1 0.3769266
## 2 0.3769266
## 3 0.3769266
## 4 0.3769266
## 5 0.3769266
## 6 0.3769266
```

Data Preperation

```
x <- model.matrix ( logmove ~
 log(price) *
(feat + brand + AGE60 + EDUC + ETHNIC +
INCOME + HHLARGE + WORKWOM + HVAL150 + SSTRDIST +
SSTRVOL + CPDIST5 + CPWVOL5)^2, data=grocery)

dim(x)

## [1] 28947   210

x=x[,-1]
dim(x)

## [1] 28947   209
```

Standardizing the Data

```
scaled.x = scale(x)
```

Splitting the Data

```
set.seed(1)
nData = nrow(scaled.x)
samples = sample(1:nData, 1000, replace=FALSE)
training = data.frame(scaled.x[samples,])
testing = data.frame(scaled.x[-samples,])

length(training[,1])

## [1] 1000

length(testing[,1])

## [1] 27947

logmovetraining = grocery$logmove[samples]
logmovetesting = grocery$logmove[-samples]
```

Linear Model

```
set.seed(1)

lmfit = lm(logmovetraining ~ ., data = training)
prediction = predict(lmfit, testing)
mean((prediction - logmovetesting)^2)

## [1] 0.4344954
```

Ridge Regression

```
set.seed(1)

trainingmatrix = model.matrix(logmovetraining ~ ., data = training)
```

```
testingmatrix = model.matrix(logmovetesting ~ ., data = testing)
grid = 10^seq(4,-2,length=100)

cv.outridge = cv.glmnet(trainingmatrix, logmovetraining, alpha = 0)

ridge.mod=glmnet(trainingmatrix, logmovetraining, alpha=0, lambda=grid, thres
h=1e-12)

bestlambdaridge = cv.outridge$lambda.min

predictionridge = predict(cv.outridge, s=bestlambdaridge, newx = testingmatri
x)

msebest = mean((predictionridge-logmovetesting)^2)

ridge.pred=predict(ridge.mod,s=1e10,newx = testingmatrix)
mseinf = mean((ridge.pred-logmovetesting)^2)

ridge.pred=predict(ridge.mod,s=0,newx = testingmatrix)
mse0 = mean((ridge.pred-logmovetesting)^2)

bestlambdaridge

## [1] 0.05618274

mseinf

## [1] 1.038713

mse0

## [1] 0.3959942

msebest

## [1] 0.3937596
```

Lasso Regression

```
set.seed(1)

lasso.mod=glmnet(trainingmatrix, logmovetraining, alpha=1, lambda=grid, thres
h=1e-12)

cv.outlasso = cv.glmnet(trainingmatrix, logmovetraining, alpha = 1)

bestlambdalasso = cv.outlasso$lambda.min

predictionlasso = predict(cv.outlasso, s=bestlambdalasso, newx = testingmatri
x)
```

```
msebest = mean((predictionlasso-logmovetesting)^2)

lasso.pred=predict(lasso.mod,s=1e10,newx = testingmatrix)
mseinf = mean((lasso.pred-logmovetesting)^2)

lasso.pred=predict(lasso.mod,s=0,newx = testingmatrix)
mse0 = mean((lasso.pred-logmovetesting)^2)

bestlambdalasso

## [1] 0.0009156386

mseinf

## [1] 1.040535

mse0

## [1] 0.4045584

msebest

## [1] 0.3995595

best.lasso.mod = glmnet(trainingmatrix, logmovetraining, alpha = 1, lambda =
bestlambdalasso, thresh=1e-12)
coefBestLasso = coef(best.lasso.mod)
sum(coefBestLasso != 0)

## [1] 104

sum(coefBestLasso == 0)

## [1] 107
```

Note: Seed set to (1) for sampling for all data.

Training Set = 20,000 observations Test data set = 8,947 observations

	MSE for $\lambda = 0$	MSE for $\lambda = \infty$	Best λ	MSE for λ = Best
Ordinary Least Squares (OLS)				0.3583494
Ridge Regression	0.3781809	1.049705	0.05995981	0.3906986
Lasso Regression	0.4043329	1.051729	0.0001831086	0.3697927

Training Set = 1,000 observations Test data set = 27,947 observations

	MSE for $\lambda = 0$	MSE for $\lambda = \infty$	Best λ	MSE for λ = Best
Ordinary Least Squares (OLS)				0.4344954
Ridge Regression	0.3959942	1.038713	0.05618274	0.3937596
Lasso Regression	0.4045584	1.040535	0.0009156386	0.3995595

Training Set = 20,000 observations Test data set = 8,947 observations

	Decrease in MSE with respect to OLS
Ridge Regression	-9.027%
Lasso Regression	-3.193%

Training Set = 1,000 observations Test data set = 27,947

	Decrease in MSE with respect to OLS
Ridge Regression	9.375%
Lasso Regression	8.041%

Lasso Coefficients

	Predictors = 0	Predictors != 0	Total Predictors
20,000 ; 8,947	72	137	209
1,000 ; 27,947	106	103	209

(Note there are two intercept values that get counted in the R calculations which are substracted off in this table).