FTAP HW 7

Zachary Fogelson

July 21, 2015

Lasso Homework

Problem 1

1.

```
train <- read.csv("train.csv")
trainMeaningFull <-subset(train, select=-c(id, member_id))
trainMeaningFull <- trainMeaningFull[complete.cases(trainMeaningFull),]
pandoc.table(head(train))
```

```
##
## ------------------------------------------------
##  id    member_id   funded_amnt   term_m   int_rate
## ----- ----------- ------------- -------- ----------
## 54734    80364        25000        36      0.1189
##
## 55742   114426        7000         36      0.1071
##
## 57245   138150        1200         36      0.1311
##
## 57416   139635        10800        36      0.1357
##
## 58915   153417        5025         36      0.1008
##
## 59006   154254        3000         36      0.1426
## ------------------------------------------------
##
## Table: Table continues below
##
##
## ----------------------------------------------------------
##  installment   emp_length   annual_inc   dti   revol_util
## ------------- ------------ ------------ ----- ------------
##     829.1          0          85000     19.48    0.521
##
##     228.2          0          65000     14.29    0.767
##
##     40.5           15         54000     5.47     0.404
##
##     366.9          6          32000     11.63    0.256
##
##     162.3          3          85000      8.1     0.732
##
##     102.9          3          80800     14.97    0.395
## ----------------------------------------------------------
```

```
pandoc.table(rbind(summary(train), sapply(train, sd)))
```

```
##
## ------------------------------------------------------------------
##        id           member_id       funded_amnt         term_m
## --------------- ---------------- ---------------- ----------------
##  Min.  : 54734    Min.  : 70699     Min.  : 500      Min.  :36.00
##
##  1st Qu.:496560  1st Qu.: 635543   1st Qu.: 5000    1st Qu.:36.00
##
##  Median :623258  Median : 798198   Median : 9250    Median :36.00
##
##   Mean :626300    Mean : 783617     Mean :10654      Mean :42.22
##
##  3rd Qu.:767345  3rd Qu.: 967828   3rd Qu.:15000    3rd Qu.:60.00
##
##  Max.  :975902    Max.  :1198245    Max.  :35000     Max.  :60.00
##
##        NA               NA               NA               NA
##
## 170023.183351294 225109.217961916 6974.75731097762 10.5166250234748
## ------------------------------------------------------------------
##
## Table: Table continues below
##
##
## ------------------------------------------------------------------
##      int_rate        installment      emp_length      annual_inc
## ----------------- ---------------- -------------- ----------------
##   Min.  :0.0542     Min.  : 15.69   Min.  : 0.000    Min.  : 4000
##
##   1st Qu.:0.0925   1st Qu.: 163.67  1st Qu.: 2.000  1st Qu.: 40002
##
##   Median :0.1171   Median : 272.80  Median : 4.000  Median : 59000
##
##    Mean :0.1191     Mean : 317.15    Mean : 6.044    Mean : 69152
##
##   3rd Qu.:0.1435   3rd Qu.: 417.36  3rd Qu.: 9.000  3rd Qu.: 82500
##
##   Max.  :0.2459     Max.  :1305.19  Max.  :15.000   Max.  :6000000
##
##         NA                NA          NA's :868           NA
##
## 0.0362426792330949 205.332202918801       NA       66645.1319511594
## ------------------------------------------------------------------
##
## Table: Table continues below
##
##
## -------------------------------
##       dti          revol_util
## --------------- --------------
##   Min.  : 0.00   Min.  :0.0000
```

```
##
##  1st Qu.: 8.00   1st Qu.:0.2420
##
##  Median :13.28   Median :0.4760
##
##   Mean :13.18     Mean :0.4774
##
##  3rd Qu.:18.49   3rd Qu.:0.7120
##
##   Max.  :29.99   Max.  :0.9990
##
##        NA           NA's :44
##
## 6.69912095939332       NA
## -------------------------------
```

2.

a)

```
ols <- lm(int_rate ~ ., trainMeaningFull)
olsSum <- summary(ols)
olsSum
```

Based on the OLS regression, all of the columns which are not IDs appear to be significant

b)

```
olsSum$r.squared
```

The baseline model has an R^2 of about .51. It is hard to judge if the baseline model is successful because, we do not have anything to compare it to. But using all of the available information we can explain 39% of the variance which is not terrible.

c)

```
plot(ols$fitted.values, ols$residuals)
```

d)

```
rmse <- mean((ols$residuals)^2)
rmse
```

e)

```
test <- read.csv("test.csv")
test <- subset(test, select=-c(id, member_id))
test <- test[complete.cases(test),]
```

```
predOLS <- predict(ols,test, interval = "none")
rmseOLS <- mean((test$int_rate - predOLS)^2)
cat("RMSE OLS: ", rmseOLS)
```

3.

```
interactDF <- as.data.frame(model.matrix(~(.)^2,subset(trainMeaningFull, select=-c(int_rate))))
interactTest <- as.data.frame(model.matrix(~(.)^2,subset(test, select=-c(int_rate))))
```

a)

```
smallest = lm(trainMeaningFull$int_rate ~ 1, interactDF)
biggest = as.formula(lm(trainMeaningFull$int_rate ~ ., interactDF))
stepForwardAIC <- step(smallest, scope = biggest, k = 2, direction ="forward", trace = F)
stepForwardBIC <- step(smallest, scope = biggest, k = log(length(trainMeaningFull$int_rate)), direction
cat("R^2 AIC: ", summary(stepForwardAIC)$r.squared, "R^2 BIC: ", summary(stepForwardBIC)$r.squared)
```

b)

```
cat("AIC: ", AIC(stepForwardAIC), "BIC: ", BIC(stepForwardBIC))
```

c)

```
rmseAIC <- mean((stepForwardAIC$residuals)^2)
rmseBIC <- mean((stepForwardBIC$residuals)^2)
cat("RMSE AIC: ", rmseAIC, "RMSE BIC: ", rmseBIC)
```

d)

```
predAIC <- predict(stepForwardAIC,interactTest, interval = "none")
predBIC <- predict(stepForwardBIC,interactTest, interval = "none")
rmseAIC <- mean((test$int_rate - predAIC)^2)
rmseBIC <- mean((test$int_rate - predBIC)^2)
cat("RMSE AIC: ", rmseAIC, "RMSE BIC: ", rmseBIC)
```

4. a)

```
biggest = lm(trainMeaningFull$int_rate ~ ., interactDF)
smallest = as.formula(lm(trainMeaningFull$int_rate ~ 1, interactDF))
stepBackwardAIC <- step(biggest, scope = smallest, k = 2, direction ="backward", trace = F)
stepBackwardBIC <- step(biggest, scope = smallest, k = log(length(trainMeaningFull$int_rate)), directio
cat("R^2 AIC: ", summary(stepBackwardAIC)$r.squared, "R^2 BIC: ", summary(stepBackwardBIC)$r.squared)
```

b)

```
cat("AIC: ", AIC(stepBackwardAIC), "BIC: ", BIC(stepBackwardBIC))
```

c)

```
rmseBackAIC <- mean((stepBackwardAIC$residuals)^2)
rmseBackBIC <- mean((stepBackwardBIC$residuals)^2)
cat("RMSE AIC: ", rmseBackAIC, "RMSE BIC: ", rmseBackBIC)
```

d)

```
predBackAIC <- predict(stepBackwardAIC,interactTest, interval = "none")
predBackBIC <- predict(stepBackwardBIC,interactTest, interval = "none")
rmseBackAIC <- mean((test$int_rate - predBackAIC)^2)
rmseBackBIC <- mean((test$int_rate - predBackBIC)^2)
cat("RMSE AIC: ", rmseBackAIC, "RMSE BIC: ", rmseBackBIC)
```

5.

a)

```
model=as.formula(paste("~", paste(names(interactDF)[-1], collapse= "+")))
x=model.matrix(model,interactDF);
lassoFit <- glmnet(x,trainMeaningFull$int_rate)
outLasso <- predict(lassoFit, newx = as.matrix(interactDF), s=.0001)
cat("R^2: ", var(outLasso)/var(ols$residuals+ols$fitted.values))
coef(lassoFit, s=.0001)
```

The R^2 is calculated based on the R^2 calculation in lasso_class; however, prof. Russell did mention that there is no R^2 for the lasso, so I printed the coefficients of the lambda model instead.

b)

```
insamplePred <- predict(lassoFit, newx = as.matrix(interactDF), s=.0001)
lassoRMSE <- mean((insamplePred - trainMeaningFull$int_rate)^2)
cat("RMSE Lasso: ", lassoRMSE)
```

c)

```
predLasso <- predict(lassoFit,newx = as.matrix(interactTest), s=.0001)
lassoRMSE <- mean((test$int_rate - predLasso)^2)
cat("RMSE Lasso: ", lassoRMSE)
```

6.

a)

```
lassoCVFit <- cv.glmnet(x,trainMeaningFull$int_rate, nfolds = 10)
lassoCVFit$lambda.min
```

b)

```
insampleCVPred <- predict(lassoCVFit, newx = as.matrix(interactDF), s="lambda.min")
lassoCVRMSE <- mean((insampleCVPred - trainMeaningFull$int_rate)^2)
cat("RMSE CVLasso: ", lassoCVRMSE)
```

c)

```
predCVLasso <- predict(lassoCVFit,newx = as.matrix(interactTest), s="lambda.min")
lassoCVRMSE <- mean((test$int_rate - predCVLasso)^2)
cat("RMSE CVLasso: ", lassoCVRMSE)
```

Logit Homework

```
trades <- read.csv("detailed_trades_est.csv")
```

1)

```
logit <- glm(PCHANGE~LASK+LBID+RETURN+SIGN_VOL, family=binomial, data=trades)
cat("Lasso AIC: ", AIC(logit), "Lasso BIC: ", BIC(logit))
```

2)

```
interactions <- subset(trades,select = -c(PCHANGE,PCHANGE0))
interactions <- as.data.frame(model.matrix(~(.)^2, interactions))
```

3)

```
trades.pca <- prcomp(interactions)
myModel <- lm(trades$PCHANGE~trades.pca$x[,1]+trades.pca$x[,2]+trades.pca$x[,3])
```

4)

```
error <- trades$PCHANGE-(as.numeric(myModel$fitted.values > .5))
mean(abs(error))
```