R Notebook

Load Libraries and data

```r
library(readr)
library(glmnet)
library(ggplot2)
library(tidyverse)
library(bnstruct)
library(MASS)
library(caret)

data = read_csv("../data/NFWBS_PUF_2016_data.csv")
head(data)
```

```
## # A tibble: 6 x 217
##   PUF_ID sample   fpl SWB_1 SWB_2 SWB_3 FWBscore FWB1_1 FWB1_2 FWB1_3
##    <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>    <dbl>  <dbl>  <dbl>  <dbl>
## 1  10350      2     3     5     5     6       55      3      3      3
## 2   7740      1     3     6     6     6       51      2      2      3
## 3  13699      1     3     4     3     4       49      3      3      3
## 4   7267      1     3     6     6     6       49      3      3      3
## 5   7375      1     3     4     4     4       49      3      3      3
## 6  10910      1     3     5     7     5       67      5      1      1
## # ... with 207 more variables: FWB1_4 <dbl>, FWB1_5 <dbl>, FWB1_6 <dbl>,
## #   FWB2_1 <dbl>, FWB2_2 <dbl>, FWB2_3 <dbl>, FWB2_4 <dbl>, FSscore <dbl>,
## #   FS1_1 <dbl>, FS1_2 <dbl>, FS1_3 <dbl>, FS1_4 <dbl>, FS1_5 <dbl>,
## #   FS1_6 <dbl>, FS1_7 <dbl>, FS2_1 <dbl>, FS2_2 <dbl>, FS2_3 <dbl>,
## #   SUBKNOWL1 <dbl>, ACT1_1 <dbl>, ACT1_2 <dbl>, FINGOALS <dbl>,
## #   PROPPLAN_1 <dbl>, PROPPLAN_2 <dbl>, PROPPLAN_3 <dbl>,
## #   PROPPLAN_4 <dbl>, MANAGE1_1 <dbl>, MANAGE1_2 <dbl>, MANAGE1_3 <dbl>,
## #   MANAGE1_4 <dbl>, SAVEHABIT <dbl>, FRUGALITY <dbl>, AUTOMATED_1 <dbl>,
## #   AUTOMATED_2 <dbl>, ASK1_1 <dbl>, ASK1_2 <dbl>, SUBNUMERACY2 <dbl>,
## #   SUBNUMERACY1 <dbl>, CHANGEABLE <dbl>, GOALCONF <dbl>, LMscore <dbl>,
## #   FINKNOWL1 <dbl>, FINKNOWL2 <dbl>, FINKNOWL3 <dbl>, FK1correct <dbl>,
## #   FK2correct <dbl>, FK3correct <dbl>, KHscore <dbl>, KHKNOWL1 <dbl>,
## #   KHKNOWL2 <dbl>, KHKNOWL3 <dbl>, KHKNOWL4 <dbl>, KHKNOWL5 <dbl>,
## #   KHKNOWL6 <dbl>, KHKNOWL7 <dbl>, KHKNOWL8 <dbl>, KHKNOWL9 <dbl>,
## #   KH1correct <dbl>, KH2correct <dbl>, KH3correct <dbl>,
## #   KH4correct <dbl>, KH5correct <dbl>, KH6correct <dbl>,
## #   KH7correct <dbl>, KH8correct <dbl>, KH9correct <dbl>, ENDSMEET <dbl>,
## #   HOUSING <dbl>, LIVINGARRANGEMENT <dbl>, HOUSERANGES <dbl>,
## #   IMPUTATION_FLAG <dbl>, VALUERANGES <dbl>, MORTGAGE <dbl>,
## #   SAVINGSRANGES <dbl>, PRODHAVE_1 <dbl>, PRODHAVE_2 <dbl>,
## #   PRODHAVE_3 <dbl>, PRODHAVE_4 <dbl>, PRODHAVE_5 <dbl>,
## #   PRODHAVE_6 <dbl>, PRODHAVE_7 <dbl>, PRODHAVE_8 <dbl>,
## #   PRODHAVE_9 <dbl>, PRODUSE_1 <dbl>, PRODUSE_2 <dbl>, PRODUSE_3 <dbl>,
## #   PRODUSE_4 <dbl>, PRODUSE_5 <dbl>, PRODUSE_6 <dbl>, CONSPROTECT1 <dbl>,
## #   CONSPROTECT2 <dbl>, CONSPROTECT3 <dbl>, EARNERS <dbl>,
## #   VOLATILITY <dbl>, SNAP <dbl>, MATHARDSHIP_1 <dbl>,
## #   MATHARDSHIP_2 <dbl>, MATHARDSHIP_3 <dbl>, MATHARDSHIP_4 <dbl>,
## #   MATHARDSHIP_5 <dbl>, ...
```

Data Cleaning

```
data <- data %>%
  remove_rownames %>%
  column_to_rownames(var="PUF_ID")

# notice that negative values are invalid entries,
# so replacing them with NA
for (i in 1:nrow(data)){
  for (j in 1:ncol(data)){
    if (data[i,j] < 0){
      data[i,j] = NA
    }
  }
}
```

```
# use knn impute to resolve NA problem
cleandata = knn.impute(as.matrix(data)) %>%
  as.data.frame()
rownames(cleandata) = rownames(data)
colnames(cleandata) = colnames(data)
colSums(is.na(cleandata)) %>% mean
```

```
## [1] 0
```

Regression using LASSO with 80/20 Train/Test data split

```
inpt = cleandata[, -which(colnames(cleandata) == "HEALTH")]
resp = cleandata$HEALTH
# separation of train and test data
testind = sample(1:nrow(cleandata), round(nrow(cleandata) * 0.2), replace = F)
train.x = inpt[-testind, ]
test.x = inpt[testind, ]
train.y = resp[-testind]
test.y = resp[testind]

# train LASSO regression
cv <- cv.glmnet(x = as.matrix(train.x), y = as.double(train.y),
               family="gaussian", alpha = 1, intercept=TRUE)
bestlambda <- cv$lambda.min
lasso <- glmnet(x = as.matrix(train.x), y = as.double(train.y),
               family="gaussian", alpha = 1, intercept=TRUE, lambda = bestlambda)

summary(lasso)
```

```
##           Length Class     Mode
## a0          1    -none-    numeric
## beta      215    dgCMatrix S4
## df          1    -none-    numeric
## dim         2    -none-    numeric
## lambda      1    -none-    numeric
## dev.ratio   1    -none-    numeric
## nulldev     1    -none-    numeric
## npasses     1    -none-    numeric
```

```
## jerr        1    -none-    numeric
## offset      1    -none-    logical
## call        7    -none-    call
## nobs        1    -none-    numeric
```

```
coef = rbind("(intercept)" = lasso$a0, as.data.frame(as.matrix(lasso$beta))) %>%
  dplyr::arrange(desc(s0)) %>% filter(s0 != 0)

coef
```

```
##                                s0
## (intercept)          2.004325e+00
## SELFCONTROL_3        9.335492e-02
## PRODHAVE_7           5.976391e-02
## SWB_2                5.323914e-02
## SHOCKS_3             5.144353e-02
## FINSOC2_7            4.900105e-02
## SWB_1                4.621479e-02
## HHEDUC               4.392744e-02
## FINSOC2_1            4.133588e-02
## SHOCKS_6             4.118799e-02
## ACT1_1               3.955163e-02
## FINSOC2_4            3.644983e-02
## SELFCONTROL_2        3.345952e-02
## OUTLOOK_2            2.593648e-02
## SHOCKS_2             2.262838e-02
## EMPLOY1_1            1.975357e-02
## PPGENDER             1.973873e-02
## ENDSMEET             1.945584e-02
## SUBNUMERACY1         1.924602e-02
## PAIDHELP             1.805709e-02
## FWB1_4               1.765160e-02
## COVERCOSTS           1.672018e-02
## INTERCONNECTIONS_1   1.653769e-02
## SUBKNOWL1            1.586547e-02
## INTERCONNECTIONS_8   1.577754e-02
## EMPLOY1_2            1.531753e-02
## HOUSESAT             1.507492e-02
## KH8correct           1.455607e-02
## SHOCKS_7             1.359083e-02
## INTERCONNECTIONS_7   1.354643e-02
## KH5correct           1.230897e-02
## FWB1_1               1.227937e-02
## fpl                  1.009620e-02
## PPEDUC               1.003033e-02
## FS1_6                8.721580e-03
## PRODHAVE_8           8.003383e-03
## PPINCIMP             7.639610e-03
## SWB_3                7.515123e-03
## SELFCONTROL_1        6.923182e-03
## BENEFITS_1           6.486401e-03
## MANAGE1_3            6.155449e-03
## LIFEEXPECT           5.542364e-03
## FINSOC2_5            4.829485e-03
## SOCSEC3              4.156309e-03
```

```
## ACT1_2                 3.730245e-03
## FWB1_3                 2.070404e-03
## CHANGEABLE             1.441510e-03
## RETIRE                 1.228652e-03
## PPMARIT                1.178596e-03
## PAREDUC                6.786660e-04
## KIDS_3                 6.697875e-04
## BENEFITS_5             6.013145e-04
## FINGOALS               2.835759e-04
## SAVINGSRANGES          1.558572e-04
## SOCSEC2                5.091142e-05
## EMPLOY                -2.734113e-04
## PRODUSE_1             -6.007069e-04
## FRAUD2                -1.890460e-03
## INTERCONNECTIONS_10 -3.167891e-03
## HSLOC                 -3.416432e-03
## PPREG4                -4.352104e-03
## PCTLT200FPL           -4.434424e-03
## KH6correct            -4.442588e-03
## MATHARDSHIP_2         -4.466183e-03
## CONSPROTECT1          -4.672809e-03
## KIDS_1                -6.177178e-03
## FINKNOWL1             -6.216764e-03
## INTERCONNECTIONS_2    -8.197397e-03
## OUTLOOK_1             -8.460387e-03
## PRODHAVE_5            -8.695254e-03
## EARNERS               -9.050680e-03
## MANAGE2               -9.789739e-03
## PPMSACAT              -1.070179e-02
## FS2_1                 -1.077778e-02
## KIDS_2                -1.115579e-02
## MILITARY              -1.195896e-02
## SCFHORIZON            -1.340734e-02
## COLLECT               -1.393843e-02
## PRODUSE_5             -1.518501e-02
## PRODHAVE_2            -2.017081e-02
## MATHARDSHIP_5         -2.480202e-02
## MANAGE1_1             -2.629325e-02
## PPT01                 -2.637656e-02
## SHOCKS_9              -2.824878e-02
## MATHARDSHIP_4         -2.827741e-02
## KHKNOWL2              -2.864334e-02
## FWB2_3                -3.103711e-02
## PPETHM                -3.130298e-02
## IMPUTATION_FLAG       -3.380611e-02
## SHOCKS_11             -4.557635e-02
## REJECTED_1            -4.875011e-02
## agecat                -5.722815e-02
## OBJNUMERACY1          -6.086741e-02
## DISTRESS              -6.383270e-02
## EMPLOY1_8             -1.328122e-01
## INTERCONNECTIONS_9    -1.345380e-01
## SHOCKS_5              -2.061217e-01
## MEMLOSS               -2.075721e-01
```

```
## EMPLOY1_6           -4.802694e-01
```

Test Regression Result

```r
pred = predict(lasso, newx = as.matrix(test.x))

mean((test.y - pred)^2)
```

```
## [1] 0.5485294
```