1. Data preparation

set.seed(222)

dataset_lab3 <- read.delim("~/Downloads/dataset_lab3.txt")
#check for data types
sapply(dataset_lab3, class)

```
> sapply(dataset_lab3, class)
   Income      Limit     Rating      Cards        Age
"numeric" "integer" "integer" "integer" "integer"
Education     Gender    Married Ethnicity    Balance
"integer"  "factor"  "factor"  "factor" "integer"
```

Verify baselines of categorical features

```
> levels(dataset_lab3$Gender)
[1] "Female" "Male"
> # "Female" "Male"
> levels(dataset_lab3$Married)
[1] "No"  "Yes"
> #"No"  "Yes"
> levels(dataset_lab3$Ethnicity)
[1] "African American" "Asian"            "Caucasian"
> #"African American" "Asian"            "Caucasian"
```

Head of original dataset (before converting categorical features to dummy variables)

```
> head(dataset_lab3)
   Income Limit Rating Cards Age Education Gender Married
1  14.891  3606    283     2  34        11   Male     Yes
2 106.025  6645    483     3  82        15 Female     Yes
3 104.593  7075    514     4  71        11   Male      No
4 148.924  9504    681     3  36        11 Female      No
5  55.882  4897    357     2  68        16   Male     Yes
6  80.180  8047    569     4  77        10   Male      No
  Ethnicity Balance
1 Caucasian     333
2     Asian     903
3     Asian     580
4     Asian     964
5 Caucasian     331
6 Caucasian    1151
```

Head of dataset after creating design matrix out of original dataset (after converting categorical features to dummy variables)

```
> head(data)
   Income Limit Rating Cards Age Education GenderMale MarriedYes
1  14.891  3606    283     2  34        11          1          1
2 106.025  6645    483     3  82        15          0          1
3 104.593  7075    514     4  71        11          1          0
4 148.924  9504    681     3  36        11          0          0
5  55.882  4897    357     2  68        16          1          1
6  80.180  8047    569     4  77        10          1          0
  EthnicityAsian EthnicityCaucasian Balance
1              0                  1     333
2              1                  0     903
3              1                  0     580
4              1                  0     964
5              0                  1     331
6              0                  1    1151
```

Dividing dataset to X and y sets

X <- data[,1:10]
y <- data[,11]

```
> head(X)
   Income Limit Rating Cards Age Education GenderMale MarriedYes
1  14.891  3606    283     2  34        11          1          1
2 106.025  6645    483     3  82        15          0          1
3 104.593  7075    514     4  71        11          1          0
4 148.924  9504    681     3  36        11          0          0
5  55.882  4897    357     2  68        16          1          1
6  80.180  8047    569     4  77        10          1          0
  EthnicityAsian EthnicityCaucasian
1              0                  1
2              1                  0
3              1                  0
4              1                  0
5              0                  1
6              0                  1
> head(y)
   1    2    3    4    5    6
 333  903  580  964  331 1151
```

Splitting data into train and test sets

#train/test split
sample <- sample.split(dataset_lab3[,1], SplitRatio = 0.8)

X_train <- subset(X, sample == TRUE)
y_train <- subset(y, sample == TRUE)

X_test <- subset(X, sample == FALSE)
y_test <- subset(y, sample == FALSE)

```
> summary(X_train)
     Income            Limit            Rating          Cards      
 Min.   : 10.35   Min.   : 1134   Min.   :112.0   Min.   :1.000  
 1st Qu.: 20.90   1st Qu.: 3086   1st Qu.:248.8   1st Qu.:2.000  
 Median : 33.12   Median : 4654   Median :344.0   Median :3.000  
 Mean   : 45.93   Mean   : 4797   Mean   :359.2   Mean   :2.987  
 3rd Qu.: 58.11   3rd Qu.: 5991   3rd Qu.:439.2   3rd Qu.:4.000  
 Max.   :186.63   Max.   :13913   Max.   :982.0   Max.   :9.000  
      Age          Education       GenderMale    
 Min.   :23.00   Min.   : 6.00   Min.   :0.0000  
 1st Qu.:42.00   1st Qu.:11.00   1st Qu.:0.0000  
 Median :57.00   Median :14.00   Median :0.0000  
 Mean   :56.37   Mean   :13.46   Mean   :0.4688  
 3rd Qu.:70.00   3rd Qu.:16.00   3rd Qu.:1.0000  
 Max.   :98.00   Max.   :19.00   Max.   :1.0000  
   MarriedYes     EthnicityAsian   EthnicityCaucasian
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000    
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000    
 Median :1.0000   Median :0.0000   Median :0.0000    
 Mean   :0.5969   Mean   :0.2625   Mean   :0.4906    
 3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000    
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000    
```

```
> summary(X_test)
     Income           Limit            Rating          Cards      
 Min.   : 10.59   Min.   :  855   Min.   : 93.0   Min.   :1.000  
 1st Qu.: 23.41   1st Qu.: 3173   1st Qu.:236.5   1st Qu.:2.000  
 Median : 33.12   Median : 4462   Median :325.0   Median :3.000  
 Mean   : 42.36   Mean   : 4490   Mean   :338.0   Mean   :2.837  
 3rd Qu.: 53.78   3rd Qu.: 5625   3rd Qu.:414.8   3rd Qu.:4.000  
 Max.   :163.33   Max.   :10673   Max.   :750.0   Max.   :7.000  
      Age          Education      GenderMale       MarriedYes   
 Min.   :24.00   Min.   : 5.0   Min.   :0.0000   Min.   :0.000  
 1st Qu.:40.75   1st Qu.:12.0   1st Qu.:0.0000   1st Qu.:0.000  
 Median :50.00   Median :13.5   Median :1.0000   Median :1.000  
 Mean   :52.86   Mean   :13.4   Mean   :0.5375   Mean   :0.675  
 3rd Qu.:66.00   3rd Qu.:16.0   3rd Qu.:1.0000   3rd Qu.:1.000  
 Max.   :83.00   Max.   :20.0   Max.   :1.0000   Max.   :1.000  
 EthnicityAsian  EthnicityCaucasian
 Min.   :0.000   Min.   :0.000     
 1st Qu.:0.000   1st Qu.:0.000     
 Median :0.000   Median :1.000     
 Mean   :0.225   Mean   :0.525     
 3rd Qu.:0.000   3rd Qu.:1.000     
 Max.   :1.000   Max.   :1.000     
```

Setting initial lambdas

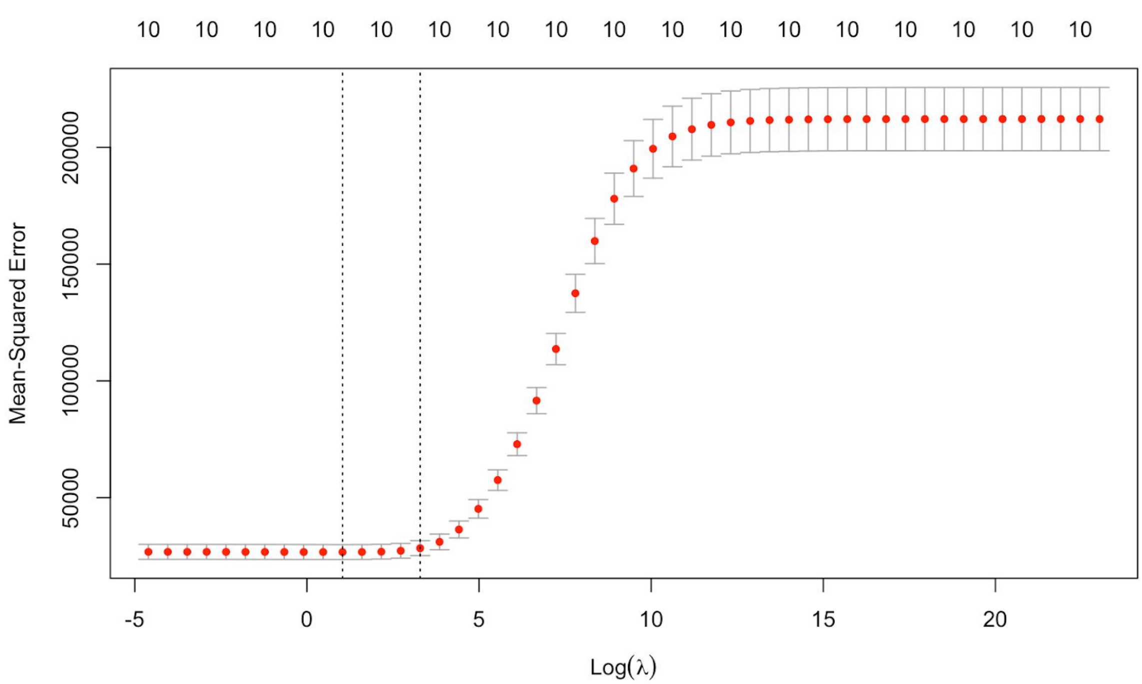
lambdas <- 10^seq(10,-2, length.out = 50)

```
> lambdas
 [1] 1.000000e+10 5.689866e+09 3.237458e+09 1.842070e+09
 [5] 1.048113e+09 5.963623e+08 3.393222e+08 1.930698e+08
 [9] 1.098541e+08 6.250552e+07 3.556480e+07 2.023590e+07
[13] 1.151395e+07 6.551286e+06 3.727594e+06 2.120951e+06
[17] 1.206793e+06 6.866488e+05 3.906940e+05 2.222996e+05
[21] 1.264855e+05 7.196857e+04 4.094915e+04 2.329952e+04
[25] 1.325711e+04 7.543120e+03 4.291934e+03 2.442053e+03
[29] 1.389495e+03 7.906043e+02 4.498433e+02 2.559548e+02
[33] 1.456348e+02 8.286428e+01 4.714866e+01 2.682696e+01
[37] 1.526418e+01 8.685114e+00 4.941713e+00 2.811769e+00
[41] 1.599859e+00 9.102982e-01 5.179475e-01 2.947052e-01
[45] 1.676833e-01 9.540955e-02 5.428675e-02 3.088844e-02
[49] 1.757511e-02 1.000000e-02
```

2. Ridge regression

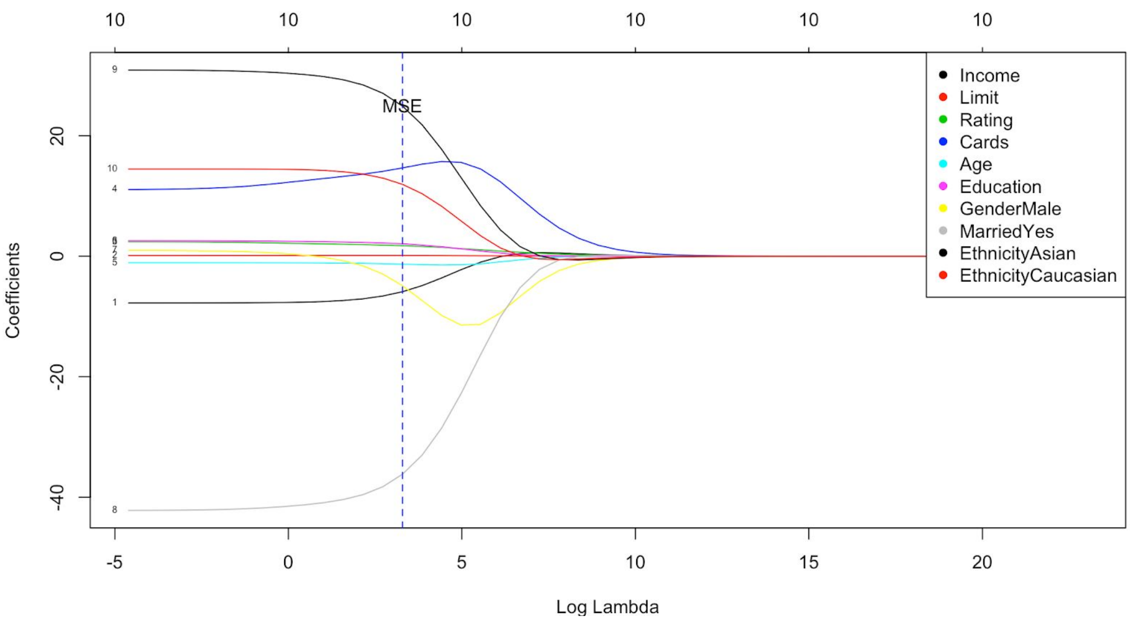
Relation between lambda and MSE

Best MSE: **28358**
Best Lambda (according to 1 SE rule): **3.289407**



Investigating the plot we see that at the beginning when the lambdas are very small we have the smallest MSE as well. But as the lambda (penalty for B coefficients) is starting to grow and the slope of the regression line is getting smaller and our model is getting closer to be just a random guess the accuracy is decreasing causing the model to be less reliable.

Change of coefficients according to provided lambda



Looking at the given outcome of k-fold cross validation we conclude that the best score of MSE occurde for lambda equals 3.289407. We can state that given lambda provides lowest variance. It also means that slope of the regression line is getting smaller compared to “regular regression” while still providing better results than slope equals to 0 which would be just an intercept (random guess).

Best lambda according to 1 SE rule: **3.289407**

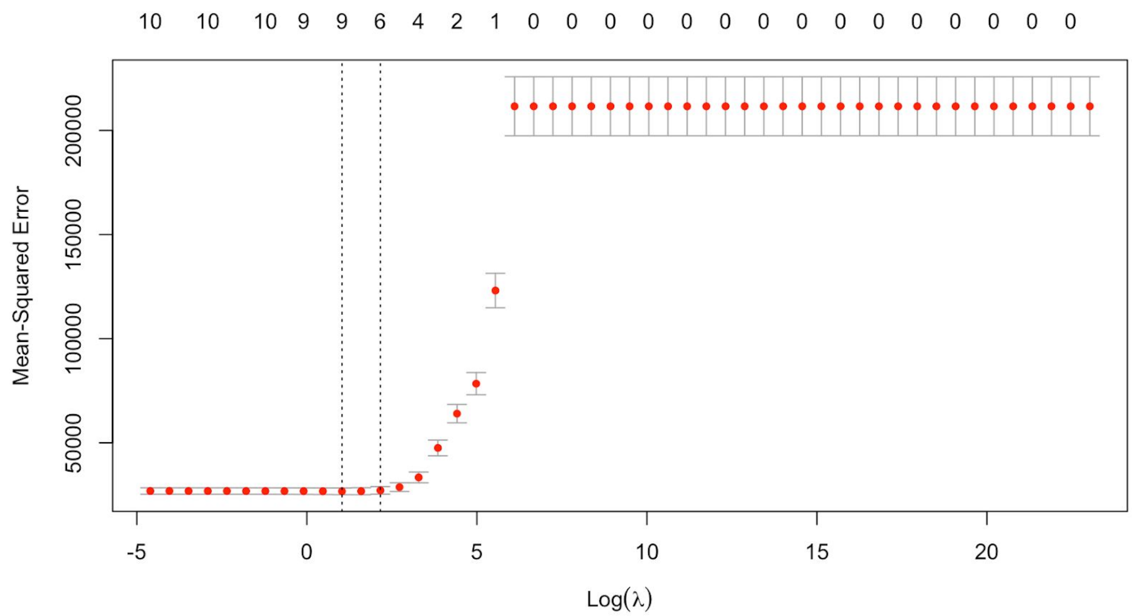
Model on train dataset evaluation:
Coefficient of determination:
R^2: **0.8730152**

Model on test dataset evaluation:
Coefficient of determination:
R^2: **0.865495**

3. Lasso regression

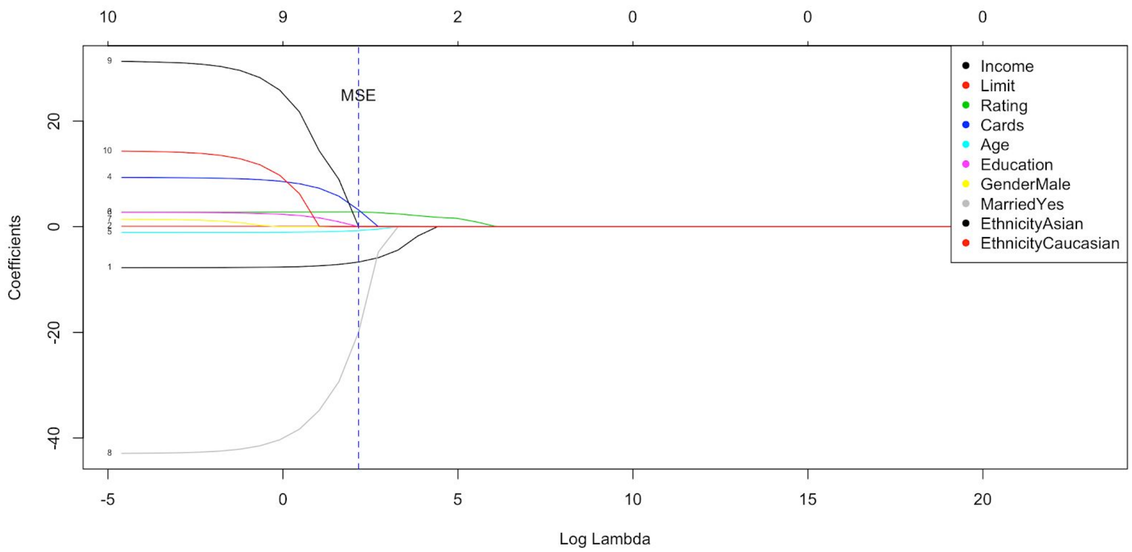
Relation between lambda and MSE

Best MSE: **27115**
Best Lambda (according to 1 SE rule): **2.16161**



Investigating the plot we see that at the beginning when the lambdas are very small we have the smallest MSE as well. But as the lambda (penalty for B coefficients) is starting to grow and the slope of the regression line is getting smaller and then at some value of lambda we see jump which we can interpret as that our slope did reached 0, so our model is actually being just a random guess and no B coefficients are taking part in predicting out outcome value.

Change of coefficients according to provided lambda



Looking at the given outcome of k-fold cross validation we conclude that the best score of MSE occurde for lambda equals 2.16161. We can state that given lambda provides lowest variance. We can see that actually some of B coefficients were canceled by our lambda penalty and are not included in predicting outcome variables anymore. Just as we could expect it form lasso regression as it allows the regression line to have slope of 0.

Best lambda according to 1 SE rule: **2.16161**

Model on train dataset evaluation:
Coefficient of determination:
R^2: **0.8770468**

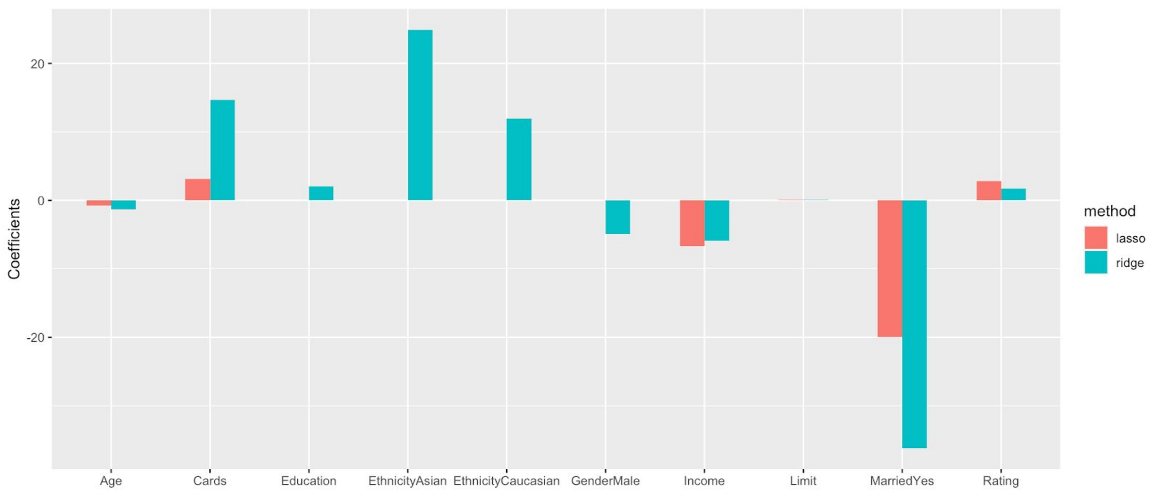
Model on test dataset evaluation:
Coefficient of determination:
R^2: **0.8688283**

4. Ridge vs. Lasso

Comparing plots of B coefficients for ridge and lasso regression we can conclude that as we would expect the ridge tends to select bigger B coefficients for model and it will never exclude them from model. Where in lasso regression at some value of lambda some of B coefficients were excluded and not taken into predicting outcome value which would result in achieving a simpler final model.

Comparing coefficients of determination (R^2) we did achieve overall slightly better results for lasso regression. We could state that some of the provided features are unnecessary for predicting outcome variables and only result in providing a more complex model. Also by comparing results of the predictions on train and test sets we can state that our models variance/bias trade-off is acceptable. To truthly state which model is performing better we could use some of more reliable criteria for evaluating efficiency of our model. But keeping in mind that we should choose a simpler model, the lasso model would be the right choice.

Coefficients for best lambdas



Just like we stated before, ridge regression will keep all of the features and less useful in prediction of outcome variables will tend to 0, but will never be excluded from the model. Where for lasso some of B coefficients will be excluded from the model and that’s what the bar plot shows. According to lasso regression the highest impact on predicted outcome value has featured **MarriedYes**, where for ridge regression **EthincityAsian**. Those results can also provide useful insights for our problem of predicting balance at the end of month for students.

5. Recursive feature elimination - wrapper

Best features according to RFE method (top 5):
MarriedYes, EthnicityAsian, EthnicityCaucasian, Cards, Income

All features selected by RFE method:
(Intercept) MarriedYes EthnicityAsian EthnicityCaucasian Cards Income GenderMale Education Rating Age Limit
(Which actually are all of available features)

```
Recursive feature selection

Outer resampling method: Cross-Validated (10 fold, repeated 5 times)

Resampling performance over subset size:

 Variables  RMSE Rsquared   MAE RMSESD RsquaredSD MAESD Selected
         4 447.4  0.09037 382.6  41.85    0.10638 30.30
         8 161.7  0.87654 123.8  27.34    0.04201 18.08
        10 161.0  0.87762 123.2  27.50    0.04135 18.56        *

The top 5 variables (out of 10):
   MarriedYes, EthnicityAsian, EthnicityCaucasian, Cards, Income
```

Summary of best model

```
Call:
lm(formula = y ~ ., data = tmp)

Residuals:
    Min      1Q  Median      3Q     Max
-230.42 -110.71  -40.62   55.14  515.50

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)        -468.78646   65.47063  -7.160 5.91e-12 ***
MarriedYes          -42.68060   18.63365  -2.291  0.02266 *
EthnicityAsian       31.19629   25.75175   1.211  0.22666
EthnicityCaucasian   14.40304   22.15802   0.650  0.51617
Cards                 9.94006    7.76887   1.279  0.20169
Income               -7.76217    0.41713 -18.608  < 2e-16 ***
GenderMale            1.27535   18.07889   0.071  0.94381
Education             2.65536    2.87731   0.923  0.35680
Rating                2.62704    0.90958   2.888  0.00415 **
Age                  -1.08054    0.53076  -2.036  0.04262 *
Limit                 0.08784    0.06079   1.445  0.14944
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 160.3 on 309 degrees of freedom
Multiple R-squared:  0.8821,    Adjusted R-squared:  0.8783
F-statistic: 231.1 on 10 and 309 DF,  p-value: < 2.2e-16
```

Model on train dataset evaluation:
Coefficient of determination:
R^2: **0.8820696**

Model on test dataset evaluation:
Coefficient of determination:
R^2: **0.8677047**