NVO Mailing Campaign Predictions with logistic LASSO and Decision Trees

```r
rm(list = ls())
knitr::opts_chunk$set(echo = FALSE, warning = FALSE, message = FALSE)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
```

```r
library(glmnet)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Loaded glmnet 4.1-8
```

```r
library(ggcorrplot)
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```r
library(dummy)
```

```
## dummy 0.1.3
## dummyNews()
```

```r
library(lubridate)
```

1. The NVO wants to predict if people are likely to respond to their mailing campaign. This makes sense as a classification problem because the response or dependent variable is going to be either yes or no, if they do respond to the mailing or if they do not. This is two classes of people.

```
##       age        numberChildren   incomeRating    wealthRating
##  Min.   : 1.00   Min.   :1.00    Min.   :1.000   Min.   :0.00
##  1st Qu.:48.00   1st Qu.:1.00    1st Qu.:2.000   1st Qu.:3.00
##  Median :62.00   Median :1.00    Median :4.000   Median :6.00
##  Mean   :61.61   Mean   :1.53    Mean   :3.886   Mean   :5.35
##  3rd Qu.:75.00   3rd Qu.:2.00    3rd Qu.:5.000   3rd Qu.:8.00
##  Max.   :98.00   Max.   :7.00    Max.   :7.000   Max.   :9.00
##  NA's   :23665   NA's   :83026   NA's   :21286   NA's   :44732
##  mailOrderPurchases totalGivingAmount  numberGifts      smallestGiftAmount
##  Min.   :  0.000    Min.   :  13.0    Min.   :  1.000   Min.   :   0.000
##  1st Qu.:  0.000    1st Qu.:  40.0    1st Qu.:  3.000   1st Qu.:   3.000
##  Median :  0.000    Median :  78.0    Median :  7.000   Median :   5.000
##  Mean   :  3.321    Mean   : 104.5    Mean   :  9.602   Mean   :   7.934
##  3rd Qu.:  3.000    3rd Qu.: 131.0    3rd Qu.: 13.000   3rd Qu.:  10.000
##  Max.   :241.000    Max.   :9485.0    Max.   :237.000   Max.   :1000.000
##
##  largestGiftAmount averageGiftAmount  yearsSinceFirstDonation
##  Min.   :   5      Min.   :   1.286   Min.   : 0.000
##  1st Qu.:  14      1st Qu.:   8.385   1st Qu.: 2.000
##  Median :  17      Median :  11.636   Median : 5.000
##  Mean   :  20      Mean   :  13.348   Mean   : 5.596
##  3rd Qu.:  23      3rd Qu.:  15.478   3rd Qu.: 9.000
##  Max.   :5000      Max.   :1000.000   Max.   :13.000
##
##  monthsSinceLastDonation inHouseDonor    plannedGivingDonor sweepstakesDonor
##  Min.   : 0.00           Mode :logical   Mode :logical      Mode :logical
##  1st Qu.:12.00           FALSE:88709     FALSE:95298        FALSE:93795
##  Median :14.00           TRUE :6703      TRUE :114          TRUE :1617
##  Mean   :14.36
##  3rd Qu.:17.00
##  Max.   :23.00
##
##   P3Donor           state            urbanicity       socioEconomicStatus
##  Mode :logical   Length:95412       Length:95412       Length:95412
##  FALSE:93395     Class :character   Class :character   Class :character
##  TRUE :2017      Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  isHomeowner       gender          respondedMailing
##  Mode:logical   Length:95412       Mode :logical
##  TRUE:52354     Class :character   FALSE:90569
##  NA's:43058     Mode  :character   TRUE :4843
```

```
##
##
##
##
```

2. If we make a model to predict if people will or won't be responsive to a mailing campaign, we can learn about which features are more predictive and what makes someone more likely to be responsive. NVO can use this information to send mail to only the people that are more likely to respond, saving time, money, and resources by not wasting as much mail being sent to people that are not very likely to respond. This way, hopefully, they get more responses in proportion to the amount of mail they send. (A higher return on their investment.)

```
## [1] "Mean of Total Giving Amonut per Person:"
```

```
## [1] 104.4894
```

3. I would think that it would be better to air on the side of sending a little bit more mail (so more false positives, send to people who are predicted to respond and donate but actually don't) than it would be to send less mail (false negatives, people who are predicted to not respond but they actually would donate). The reason I think this is because I think that the amount of money brought in by just one donation would most likely be worth it to have a few extra letters sent. (Since the average total given amount is about $104 and the cost of stamps and papers is less than a few dollars.) So, the confusion matrix measure I will use to check accuracy will be false negative rate. I will try to make sure this is as low as possible while balancing it with overall accuracy (the number of accurate predictions divided by the total number of predictions in the test data.)

Dealing with Missing Data

Age Median Imputation

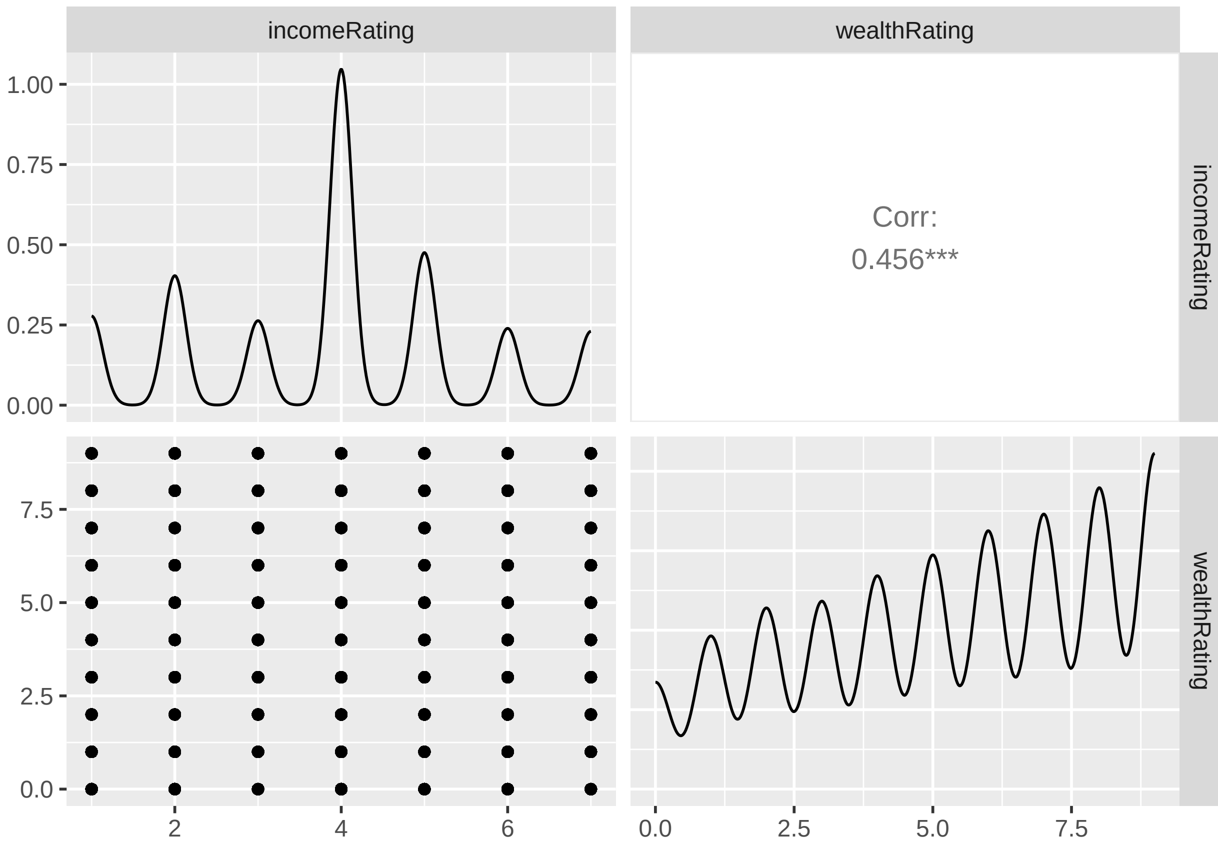
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   52.00   62.00   61.71   71.00   98.00
```

Number Children Remove Column (Since the number of missing of observations with missing data is so high in this, I decided to just drop the variable.)

Income Rating Median Imputation

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   3.000   4.000   3.912   5.000   7.000
```

Wealth Rating Are wealth rating and income rating very highly correlated? If so, I might drop wealth rating and just use income rating.



It looks like I should not just drop the column. I'll use median imputation.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   5.000   6.000   5.652   6.000   9.000
```

Is Homeowner Value Imputation. It looks like all of the values in this variable are true, so I'm going to assume that all of the NA's are false. In a real world situation I would probably confirm before assuming that.

```
##    Mode   FALSE    TRUE
## logical   43058   52354
```

Gender, Urbanicity, and SocioEconomic Status

```
#Summary of Gender Statistics
#0 indicates missing data
summary(data$gender)
```

```
##      0 female  joint   male
##   4676  51277    365  39094
```

Recoding all Factor Type-Variables with Dummy Variables

```
dum_dum = dummy(data, int = TRUE)
num_num = data %>%
  keep(is.numeric)
data = bind_cols(num_num, dum_dum)
rm(dum_dum, num_num)
```

```
# Partition the data.(and remove respondedMailing_FALSE)
data = subset(data, select = -c(respondedMailing_FALSE) )
```

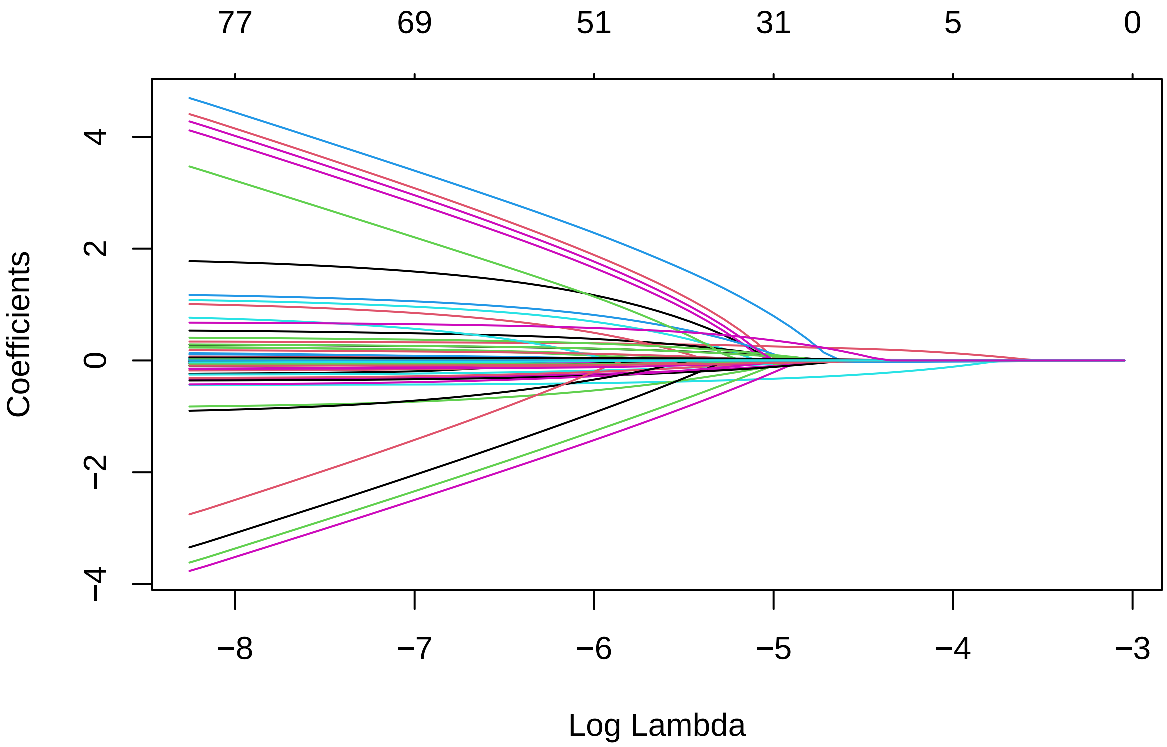
```r
set.seed(555)
samp = createDataPartition(data$respondedMailing_TRUE, p = 0.70, list = FALSE)
training = data[samp, ]
testing = data[-samp, ]
rm(samp)
```

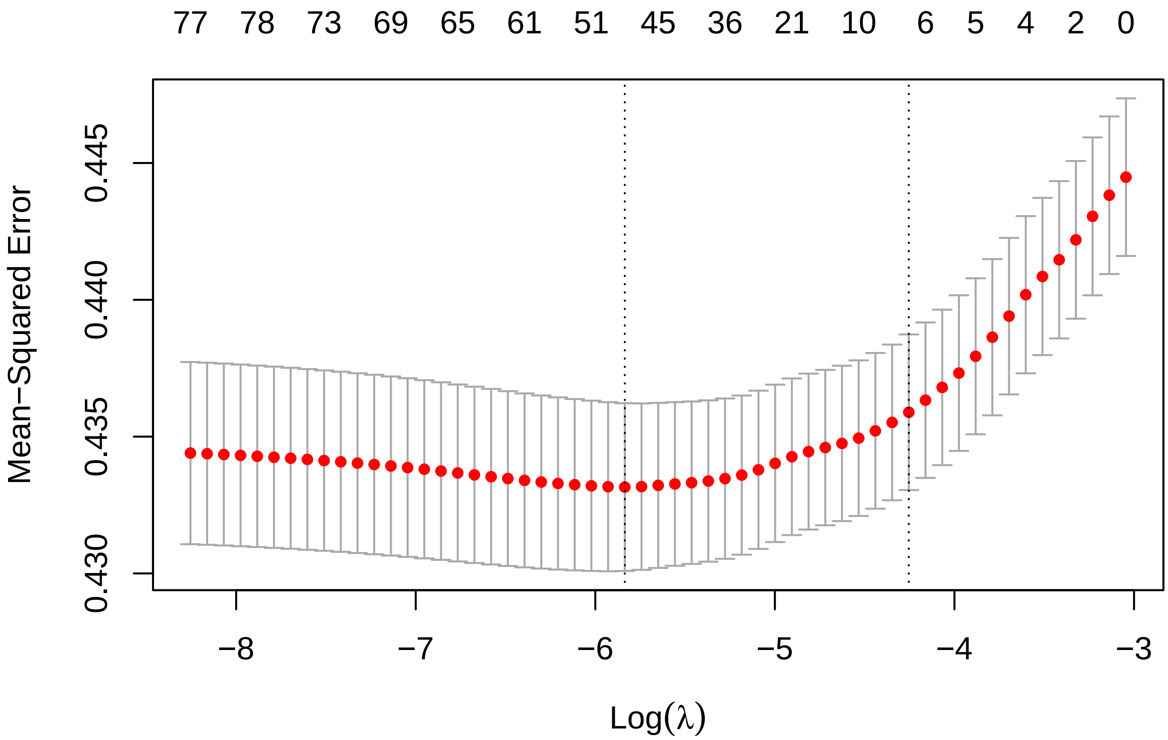
Smote

```
## respondedMailing_TRUE
##    0    1
## 6812 3406
```

4. Logistic LASSO for finding which predictors to use

```
##    0    1
## 6812 3406
```






```
## 92 x 1 sparse Matrix of class "dgCMatrix"
##                                    s1
## (Intercept)                -8.212198e-02
## age                         .
## incomeRating                3.239187e-02
## wealthRating                .
## mailOrderPurchases          3.882800e-04
## totalGivingAmount          -3.092208e-04
## numberGifts                 1.887338e-02
## smallestGiftAmount          .
## largestGiftAmount           .
## averageGiftAmount          -1.432125e-02
## yearsSinceFirstDonation     .
## monthsSinceLastDonation    -3.303869e-02
## inHouseDonor_FALSE          .
## inHouseDonor_TRUE           .
## plannedGivingDonor_FALSE    .
## plannedGivingDonor_TRUE     .
## sweepstakesDonor_FALSE      9.896791e-02
## sweepstakesDonor_TRUE      -8.321652e-15
## P3Donor_FALSE              -3.968499e-01
## P3Donor_TRUE                .
## state_AA                    1.048386e+00
## state_AE                    1.662985e+00
## state_AK                   -4.835475e-01
## state_AL                    1.987168e-02
## state_AP                    6.139780e-01
## state_AR                    .
## state_AS                    .
## state_AZ                    1.980576e-01
```

```
## state_CA                      3.010270e-01
## state_CO                      .
## state_CT                      2.076464e+00
## state_DC                      .
## state_DE                      .
## state_FL                      .
## state_GA                      .
## state_GU                      1.436085e+00
## state_HI                      3.547737e-01
## state_IA                     -4.839023e-02
## state_ID                      .
## state_IL                      .
## state_IN                     -1.673967e-01
## state_KS                      .
## state_KY                      .
## state_LA                     -7.759272e-02
## state_MA                     -1.079691e+00
## state_MD                      7.415435e-01
## state_ME                      .
## state_MI                      .
## state_MN                     -2.258622e-01
## state_MO                     -2.332721e-02
## state_MS                     -1.835986e-01
## state_MT                      5.569301e-02
## state_NC                      .
## state_ND                      .
## state_NE                      .
## state_NH                     -7.372349e-01
## state_NJ                      3.958161e-01
## state_NM                      2.830524e-01
## state_NV                      .
## state_NY                      .
## state_OH                     -1.238921e+00
## state_OK                     -2.557505e-01
## state_OR                      1.067361e-01
## state_PA                      .
## state_RI                      .
## state_SC                      1.988107e-01
## state_SD                      .
## state_TN                      .
## state_TX                      .
## state_UT                     -2.532778e-01
## state_VA                     -2.507317e-01
## state_VI                      .
## state_VT                      9.422350e-01
## state_WA                      .
## state_WI                      .
## state_WV                      1.542628e+00
## state_WY                      .
## urbanicity_0                  .
## urbanicity_city               .
## urbanicity_rural             -6.540101e-02
## urbanicity_suburb             9.064012e-03
## urbanicity_town               .
```

```
## urbanicity_urban            -1.125207e-01
## socioEconomicStatus_average  .
## socioEconomicStatus_highest  4.035599e-02
## socioEconomicStatus_lowest  -7.493665e-02
## isHomeowner_FALSE            .
## isHomeowner_TRUE             .
## gender_0                     .
## gender_female               -1.044633e-02
## gender_joint                 5.598898e-01
## gender_male                  .
```

These are the predictors for the minimum lamda (it looks around -6). Some things I thought were interesting is that is they were not a sweepstakes donor, they were more likely to respond to mailing. I also thought it was interesting that states like Maryland and Vermont were very positively correlated with responding and states like Louisiana and New Hampshire were very negatively correlated.

Train a logistic lasso model

```r
#build the model with min lambda

lasso.model <- glmnet(X, y, alpha = 1, family = "binomial",
                lambda = cv_lasso$lambda.min)
```

```r
#Using the model to make predictions on the test data

x.test <- model.matrix(respondedMailing_TRUE ~., testing)[,-1]
probabilities <- lasso.model %>% predict(newx = x.test)
predicted.classes <- factor(ifelse(probabilities > 0.5, "1", "0"))

# Model accuracy

testing.y <- factor(testing$respondedMailing_TRUE)

confusionMatrix(predicted.classes, testing.y, positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 27122  1422
##          1    64    15
##
##                Accuracy : 0.9481
##                  95% CI : (0.9455, 0.9506)
##     No Information Rate : 0.9498
##     P-Value [Acc > NIR] : 0.9093
##
##                   Kappa : 0.0146
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.0104384
##             Specificity : 0.9976458
##          Pos Pred Value : 0.1898734
##          Neg Pred Value : 0.9501822
##              Prevalence : 0.0502044
```

```
##          Detection Rate : 0.0005241
##    Detection Prevalence : 0.0027600
##       Balanced Accuracy : 0.5040421
##
##        'Positive' Class : 1
##
```

The accuracy is 94% which is pretty good! The false negative rate though is 98.956% which is pretty high. I would want it to predict a little bit more yes's and send more mail out.