Charitable Organizations Classification and Prediction Models

Kailu Wang, Kaiyue Wang, Georgia Tompkins

**Introduction:**

The purpose of this project is to help the charitable organization to maximize the profits in order to acquire the potential successful outcomes of the donations. There are two response variables that are donar and damt in the analysis. In order to help the charitable organization succeed, we will develop a classification model and prediction model in order to capture the donors that maximize the expected net profits.

By running the describe function in R, we could see the vars, mean, sd, median, trimmed mean, min, max, range, skew, kurtosis and se of the various variables. Some variables seem to have errors such as ID, avhv, tgif and lgif since they have a large max value. Additionally, as we look at the trimmed, skew and kurtosis outputs we find out that the ID, avhv, incm, tgif and lgif are not normally distributed suggesting there may be errors that need to be fixed. However, by checking missing values, we find there are no missing values.

> describe(charity)

vars n mean sd median trimmed mad min max range skew kurtosis se

ID 1 8009 4005.00 2312.14 4005.00 4005.00 2968.17 1.00 8009.00 8008.00 0.00 -1.20 25.84

reg1 2 8009 0.20 0.40 0.00 0.13 0.00 0.00 1.00 1.00 1.50 0.24 0.00

reg2 3 8009 0.32 0.47 0.00 0.27 0.00 0.00 1.00 1.00 0.78 -1.40 0.01

reg3 4 8009 0.13 0.34 0.00 0.04 0.00 0.00 1.00 1.00 2.15 2.63 0.00

reg4 5 8009 0.14 0.35 0.00 0.05 0.00 0.00 1.00 1.00 2.08 2.33 0.00

home 6 8009 0.87 0.34 1.00 0.96 0.00 0.00 1.00 1.00 -2.16 2.64 0.00

chld 7 8009 1.72 1.40 2.00 1.61 1.48 0.00 5.00 5.00 0.27 -0.80 0.02

hinc 8 8009 3.91 1.47 4.00 3.89 1.48 1.00 7.00 6.00 0.01 -0.09 0.02

genf 9 8009 0.61 0.49 1.00 0.63 0.00 0.00 1.00 1.00 -0.43 -1.81 0.01

wrat 10 8009 6.91 2.43 8.00 7.35 1.48 0.00 9.00 9.00 -1.35 0.79 0.03

avhv 11 8009 182.65 72.72 169.00 174.55 59.30 48.00 710.00 662.00 1.54 4.49 0.81

incm 12 8009 43.47 24.71 38.00 40.19 19.27 3.00 287.00 284.00 2.05 8.31 0.28

inca 13 8009 56.43 24.82 51.00 53.46 19.27 12.00 305.00 293.00 1.94 7.87 0.28

plow 14 8009 14.23 13.41 10.00 12.24 11.86 0.00 87.00 87.00 1.36 1.89 0.15

npro 15 8009 60.03 30.35 58.00 58.88 34.10 2.00 164.00 162.00 0.31 -0.62 0.34

tgif 16 8009 113.07 85.48 89.00 99.77 47.44 23.00 2057.00 2034.00 6.55 107.52 0.96

lgif 17 8009 22.94 29.95 16.00 17.43 10.38 3.00 681.00 678.00 7.81 110.38 0.33

rgif 18 8009 15.66 12.43 12.00 13.66 8.90 1.00 173.00 172.00 2.63 13.92 0.14

tdon 19 8009 18.86 5.78 18.00 18.31 4.45 5.00 40.00 35.00 1.10 2.12 0.06

tlag 20 8009 6.36 3.70 5.00 5.70 2.97 1.00 34.00 33.00 2.42 8.41 0.04

agif 21 8009 11.68 6.57 10.23 10.83 5.50 1.29 72.27 70.98 1.78 6.02 0.07

donr 22 6002 0.50 0.50 0.00 0.50 0.00 0.00 1.00 1.00 0.00 -2.00 0.01

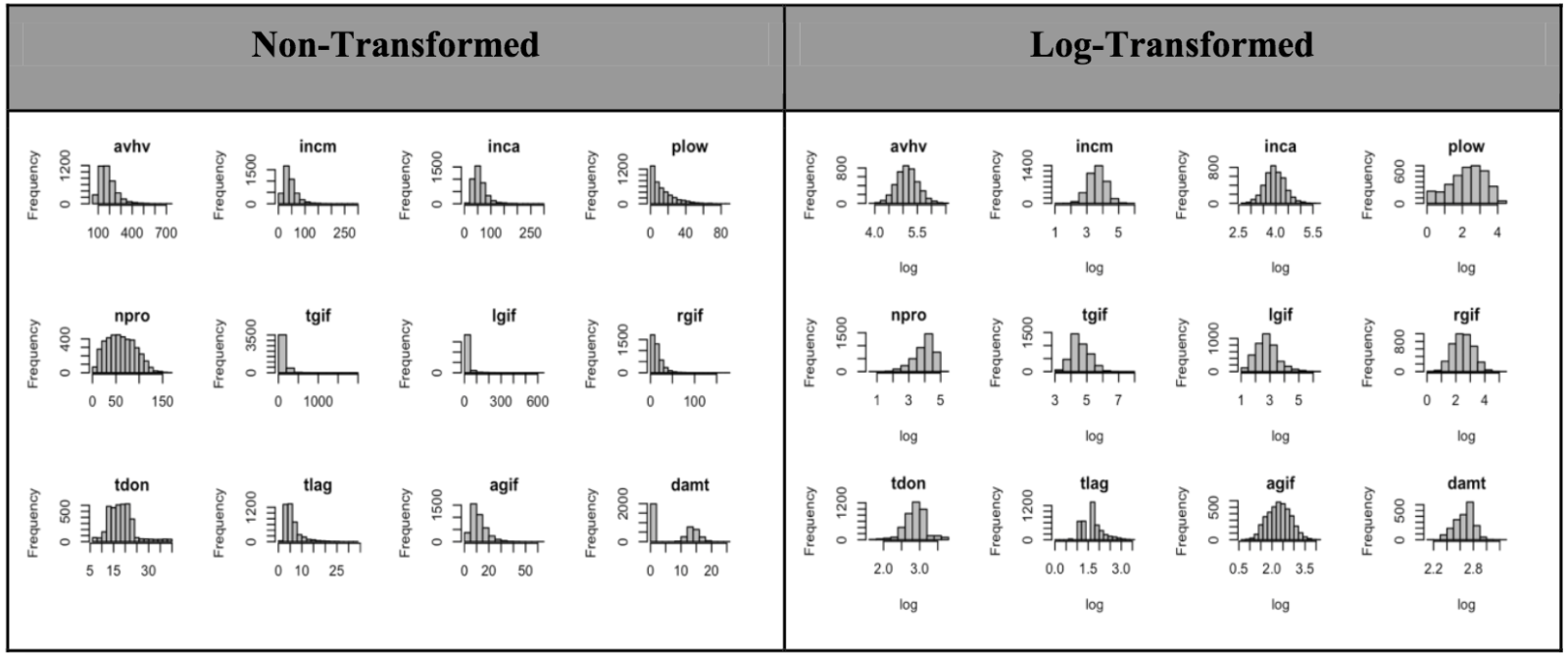
damt 23 6002 7.21 7.36 0.00 6.85 0.00 0.00 27.00 27.00 0.12 -1.83 0.10

part\* 24 8009 2.00 0.71 2.00 2.00 1.48 1.00 3.00 2.00 0.00 -1.01 0.01

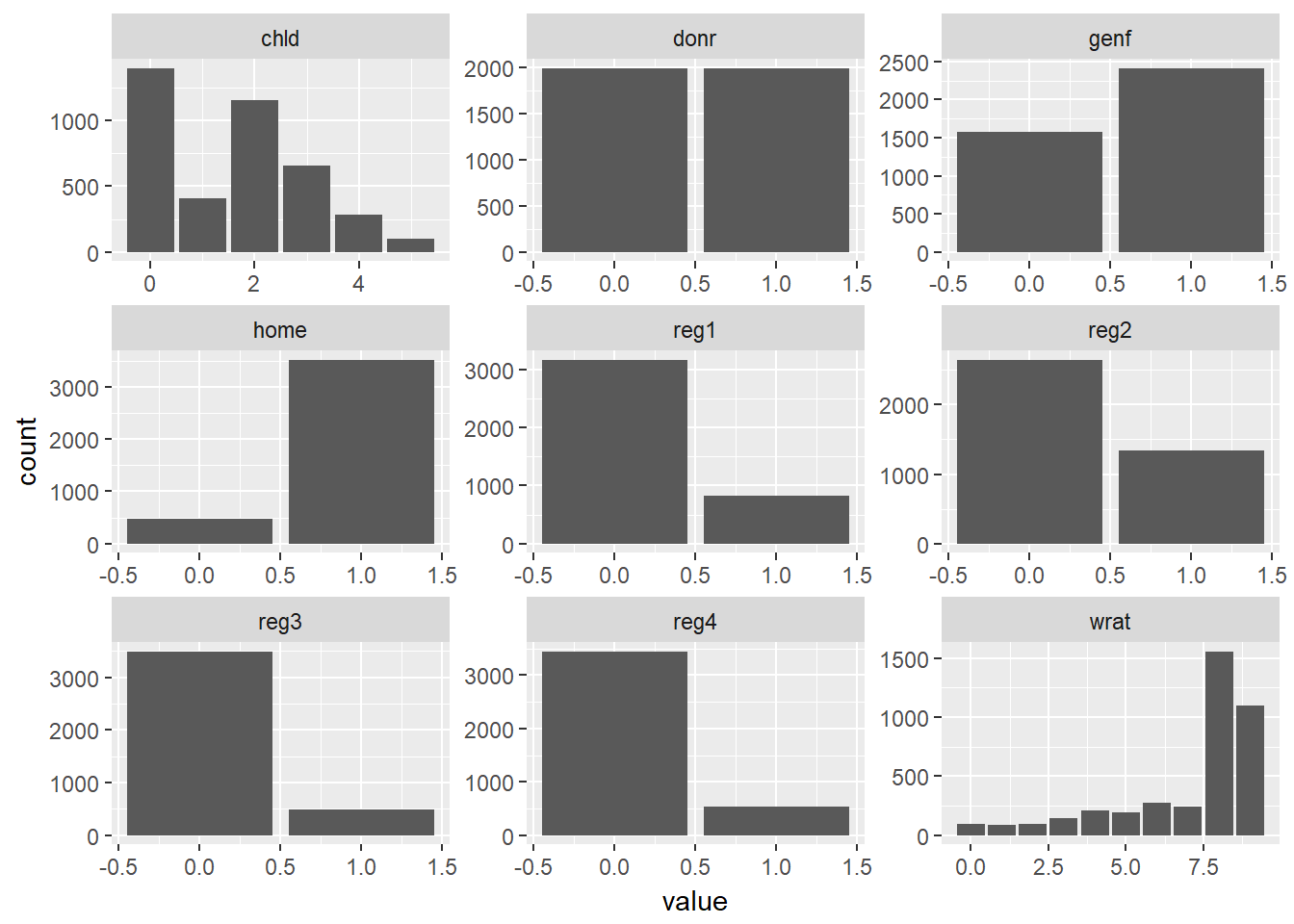
part damt donr agif tlag tdon rgif lgif tgif npro plow inca incm avhv wrat genf hinc chld home reg4 reg3 reg2 reg1 ID

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

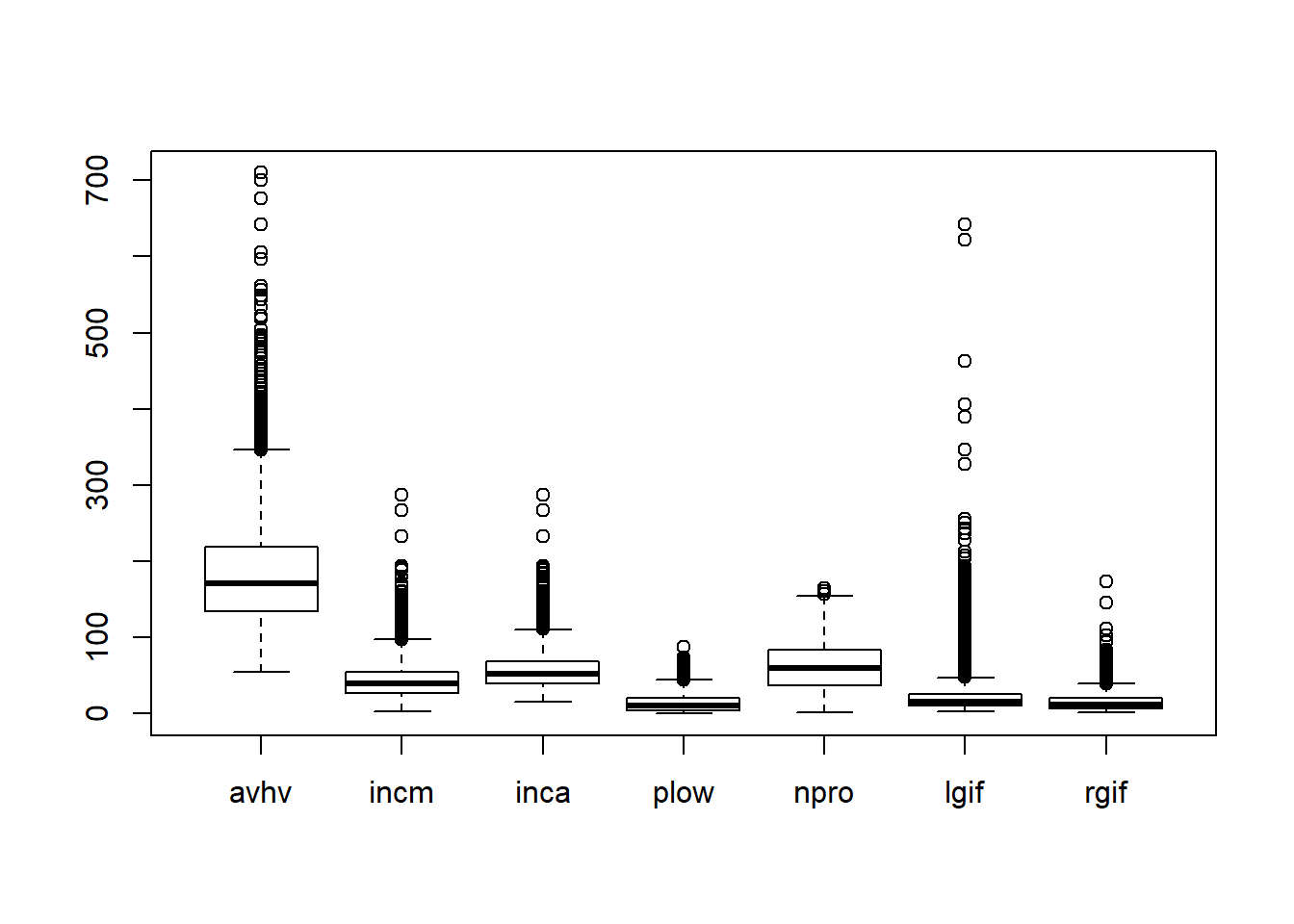
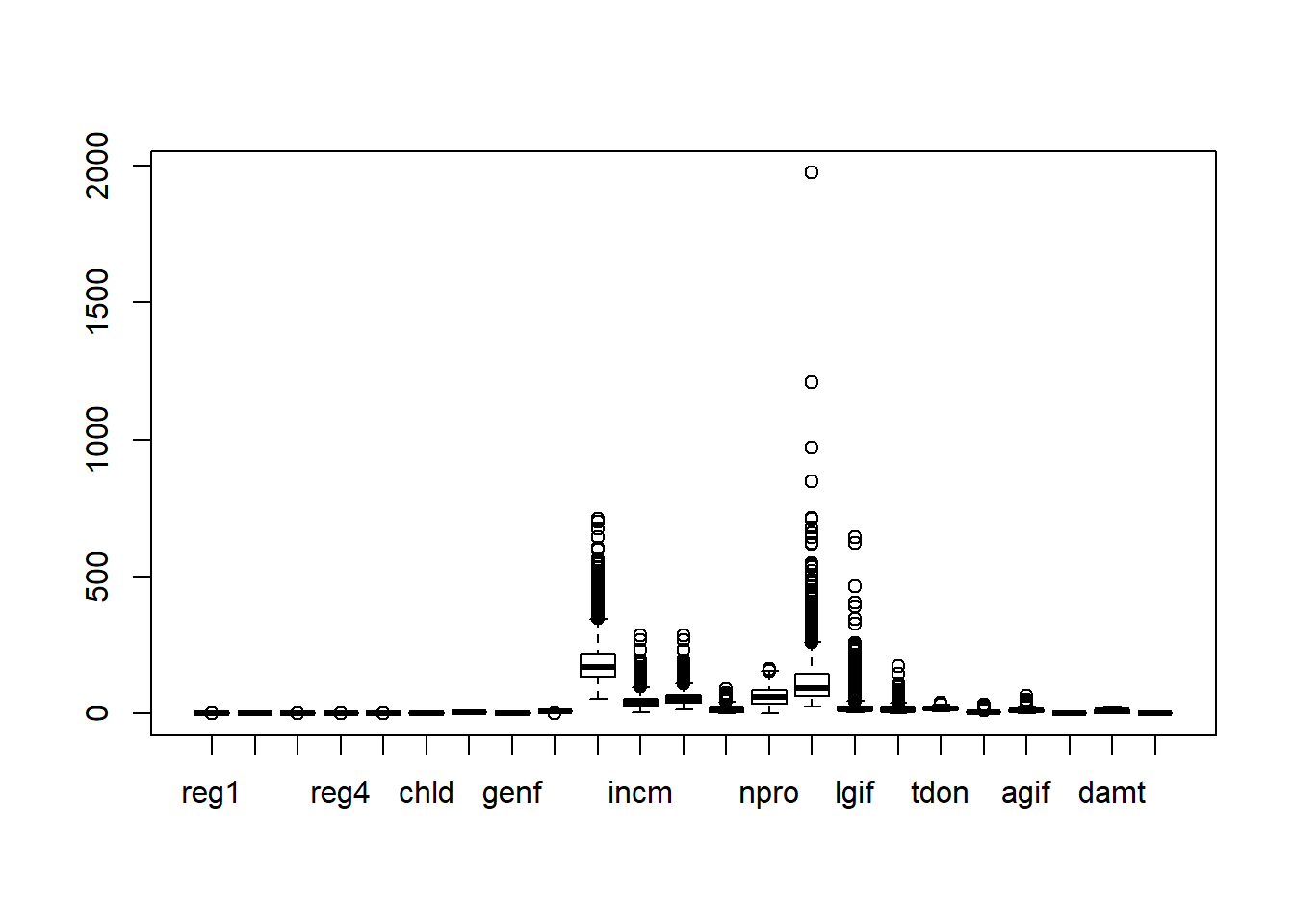
Utilizing ggplot, we can identify graphs with large outliers or extreme values that need correction. By observing the ggplot graph below, some of the histograms have extreme values since their center is in the extreme left and right such as agif, home, inca, incm, reg1, reg2, reg3, reg4 etc.Furthermore, chld, damt, npro are not normally distributed which means they may have errors. Comparatively, log-transform has an improved performance that the histograms of some variables become more normally distributed such as avha, incm, inca, plow, tgif, lgif, and rgif. Hence, we may use log-transformed variables into the models for better prediction.



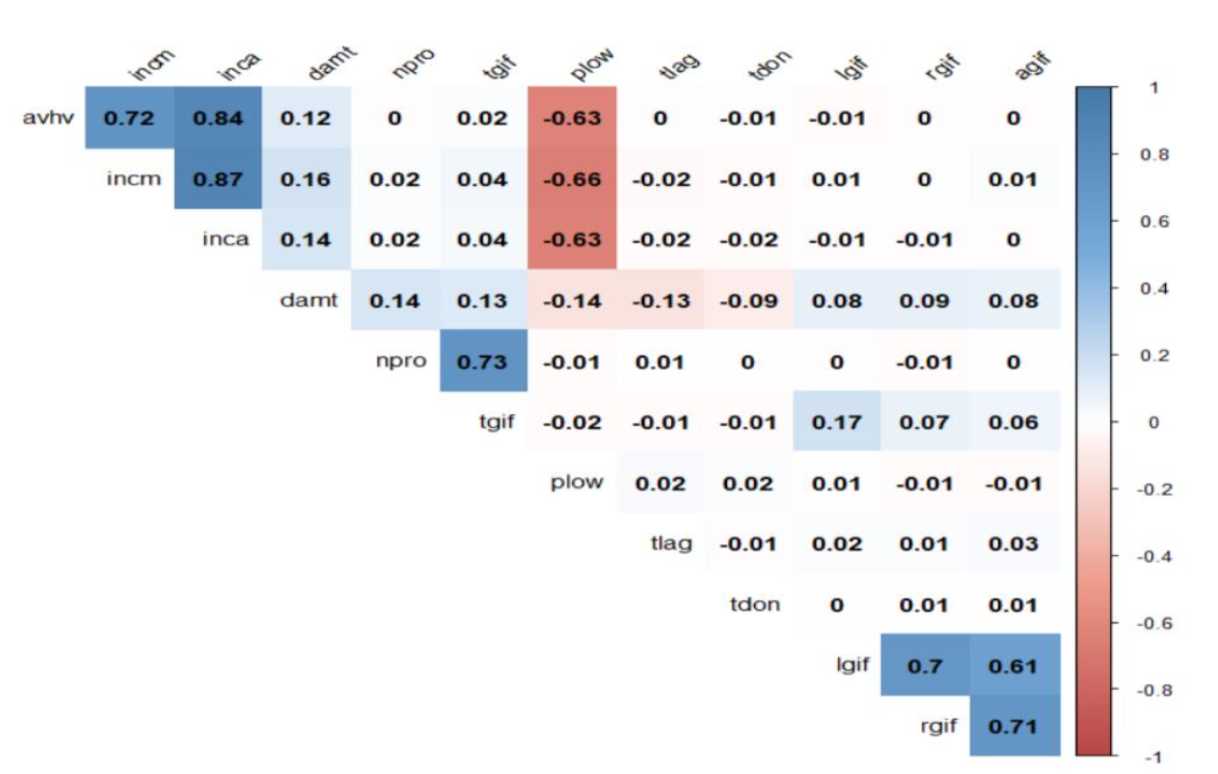
The bar graph could help us to see more clearly about the data distribution. Only donr is unchanged but all other graphs’ count is changed through value. Almost no graphs are normally distributed so we will use these when we build models.



Then we look at the boxplot to visualize the spread, median, and quartiles. We can observe that the incm, npro, gif and avhv have large outliers and extreme values which may have the same problems from the first place that they have large maximum values and ranges.



Next, we find the correlation between numerical variables. From the graph below, we can observe clearly that damt has a positive relation npro and tgif but little relation with lgif, rgif, and agif. Also, damt has a negative correlation with plow, tlag, and tdon. These relationships could help us to build models. Let’s talk more detail in data preparation and may use these correlations to build models.



**Preparation:**

The data is prepared via standardization and thus variables have zero mean deviation and zero unit standard deviation.

reg1 reg2 reg3 reg4 home chld hinc

2.151811e-17 -2.526099e-17 3.693258e-17 -6.017778e-17 -9.663428e-18- 2.051129e-17 -1.463197e-17

genf wrat avhv incm inca plow npro

4.465563e-17 -1.062688e-16 1.816327e-17-1.335981e-16 8.260581e-17 4.848389e-17 -6.655491e-17

tgif lgif rgif tdon tlag agif

2.532023e-17 2.669731e-17 5.429393e-18 -1.835154e-16 1.010477e-16 -1.258480e-16

reg1 reg2 reg3 reg4 home chld hinc genf wrat avhv incm inca plow npro tgif lgif rgif tdon tlag agif

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

In the section the train, validation and test datasets have been separated. We also standardized all of the column fields to avoid the scaling issues affecting some statistical methods. Then we are going to look at the correlation among all the variables and select the most useful variables. We find that incm, inca and plow are highly correlated; npro and tgif are highly correlated; and lgif, rgif and agif are highly correlated. These correlations may have some influences on the models.

hinc avhv incm inca plow npro tgif lgif

hinc 1.000000000 0.017689974 0.012054151 0.0131884781-0.006398353 0.0230979942 0.01149685 -0.009139114

avhv 0.017689974 1.000000000 0.722519341 0.8408769868-0.625136632 -0.0026582392 0.01888690 -0.011299427

incm 0.012054151 0.722519341 1.000000000 0.8729018527-0.655455152 0.0170736403 0.04167707 0.006731373

inca 0.013188478 0.840876987 0.872901853 1.0000000000-0.634632891 0.0182656423 0.03544441 -0.007539508

plow -0.006398353 -0.625136632 -0.655455152 -0.63463289061.000000000 -0.0116676863 -0.01761563 0.005928332

npro 0.023097994 -0.002658239 0.017073640 0.0182656423-0.011667686 1.0000000000 0.72664798 -0.001328901

tgif 0.011496846 0.018886905 0.041677073 0.0354444121-0.017615631 0.7266479833 1.00000000 0.173434853

lgif -0.009139114 -0.011299427 0.006731373 -0.0075395084-0.017615631 0.7266479833 1.00000000 0.173434853

rgif -0.002380876 -0.003325955 0.002748777 -0.00537412190.005928332 -0.0013289011 0.17343485 1.000000000

tdon 0.014723148 -0.008495970 -0.013859730 -0.0162847340 0.016682061 0.0009863842 -0.01130312 0.003634269

tlag -0.030735162 -0.004848666 -0.021305788 -0.01694511190.021362168 0.0057474624 -0.01172427 0.016816132

agif -0.004664742 -0.002257410 0.010328347 -0.0001531168-0.013683353 0.0022454370 0.05577226 0.609630157

rgif tdon tlag agif

hinc -0.002380876 0.0147231478 -0.030735162 -0.0046647419

avhv -0.003325955 -0.0084959698 -0.004848666 -0.0022574096

incm 0.002748777 -0.0138597299 -0.021305788 0.0103283470

inca -0.005374122 -0.0162847340 -0.016945112 -0.0001531168

plow -0.013643922 0.0166820612 0.021362168 -0.0136833534

npro -0.012489364 0.0009863842 0.005747462 0.0022454370

tgif 0.073608064 -0.0113031199 -0.011724267 0.0557722567

lgif 0.696059825 0.0036342686 0.016816132 0.6096301573

rgif 1.000000000 0.0063324838 0.012158261 0.7053413905

tdon 0.006332484 1.0000000000 -0.006151912 0.0079658768

tlag 0.012158261 -0.0061519123 1.000000000 0.0299336658

agif 0.705341391 0.0079658768 0.029933666 1.0000000000

Then a summary is run of the model with all the variables included in order to select significant variables. Variables reg3, reg4, home, chld, hinc, genf, incm, plow, npro, rgif, tdon, agif have a higher absolute t-value and hence are significant. These variables are better utilized for building models.

Call:

lm(formula = damt ~ ., data = data.train.std.damt)

Residuals:

Min 1Q Median 3Q Max

-4.4624 -0.7966 -0.1533 0.5999 9.1086

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 14.18934 0.04735 299.660 < 2e-16 \*\*\*

reg1 -0.03923 0.03962 -0.990 0.32217

reg2 -0.07434 0.04294 -1.731 0.08352 .

reg3 0.32690 0.04041 8.089 1.04e-15 \*\*\*

reg4 0.63517 0.04158 15.275 < 2e-16 \*\*\*

home 0.23834 0.06073 3.925 8.99e-05 \*\*\*

chld -0.60477 0.03794 -15.939 < 2e-16 \*\*\*

hinc 0.50143 0.03984 12.587 < 2e-16 \*\*\*

genf -0.06318 0.02850 -2.217 0.02675 \*

wrat -0.00109 0.04150 -0.026 0.97905

avhv -0.04815 0.05136 -0.937 0.34864

incm 0.29408 0.05845 5.031 5.32e-07 \*\*\*

inca 0.04726 0.07161 0.660 0.50936

plow 0.24829 0.04341 5.719 1.23e-08 \*\*\*

npro 0.13613 0.04442 3.065 0.00221 \*\*

tgif 0.05965 0.04603 1.296 0.19517

lgif -0.05501 0.03843 -1.431 0.15251

rgif 0.51685 0.04387 11.783 < 2e-16 \*\*\*

tdon 0.07254 0.03493 2.077 0.03796 \*

tlag 0.02206 0.03366 0.655 0.51229

agif 0.67139 0.04048 16.585 < 2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.273 on 1974 degrees of freedom

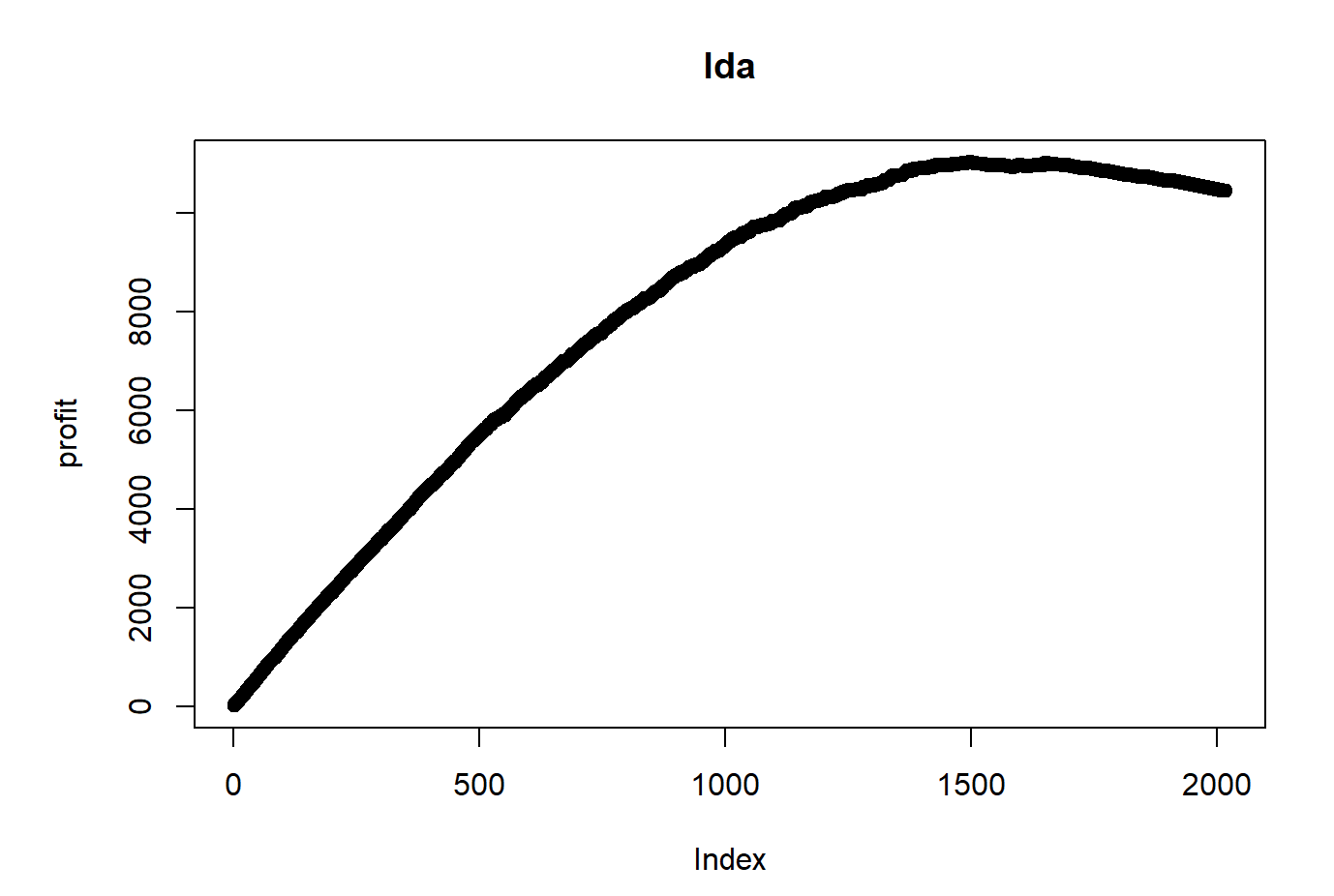
Multiple R-squared: 0.5722, Adjusted R-squared: 0.5679

F-statistic: 132 on 20 and 1974 DF, p-value: < 2.2e-16

**Classification Models:**

Model 1 (LDA model):

Model 1 is a LDA model with an accuracy of 0.722. The maximum profits of the LDA model is 11040. Also, I built a small table that contains numbers of the actual valid and non-valid donr, predicted valid and non-valid donr. The actual numbers of valid or non-valid donr is on the column and the predicted valid or non-valid donr is on the rows and they are both set as valid (1), or not valid (0). By looking at the table, it is found that there are 489 variables that are both actual and predicted non-valid, 968 variables are both actual and predicted valid. 530 variables are non-valid but predicted to be valid. 31 variables are valid but predicted as non-valid.



y.valid.donr

pred.valid.donr 0 1

0 489 31

1 530 968

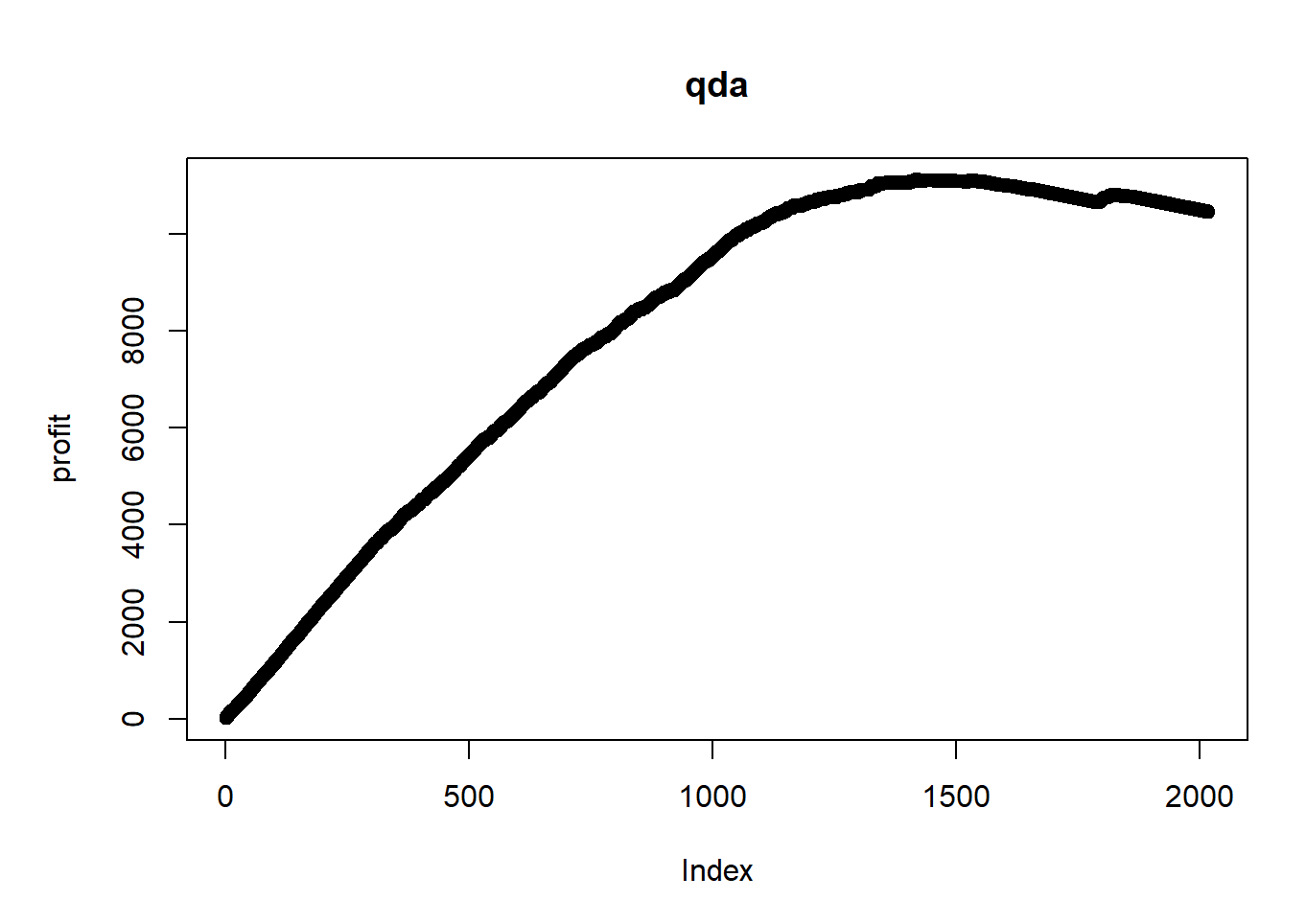
Accuracy

0.722002

[1] "Maximum profit earned with LDA model is: 11040"

Model 2 (QDA model):

Model 2 is a QDA model and the graphs are similar to that of the LDA model. The accuracy is 0.755 and maximum profit is 11111 which is slightly better than model 1. The number of actual and predicted non-valid donr has increased to 562. Valid and predicted valid donr has decreased to 962. The other two coefficients also change slightly.



y.valid.donr

pred.valid.donr 0 1

0 562 37

1 457 962

Accuracy

0.7552032

[1] "Maximum profit earned with QDA model is: 11111"

Model 3 (logistic regression model 1):

The logistic regression model also has a similar graph. The accuracy and maximum profit increased again, exceeding model 2. Compared with the QDA mode, the number of actual and predicted non-valid donr increased to 612 and actual and predicted valid donr increased to 962. The other two coefficients also changed slightly.

y.valid.donr

pred.valid.donr 0 1

0 612 20

1 407 979

Accuracy

0.7884044

[1] "Maximum profit earned with logistic model is: 11423.5"

Model 4 (logistic model 2):

Another logistic regression model was built to see if it would improve the model, but it turns out to be a little worse than model 3 with a lower maximum profit and no improvement in true predictions.

y.valid.donr

pred.valid.donr 0 1

0 610 21

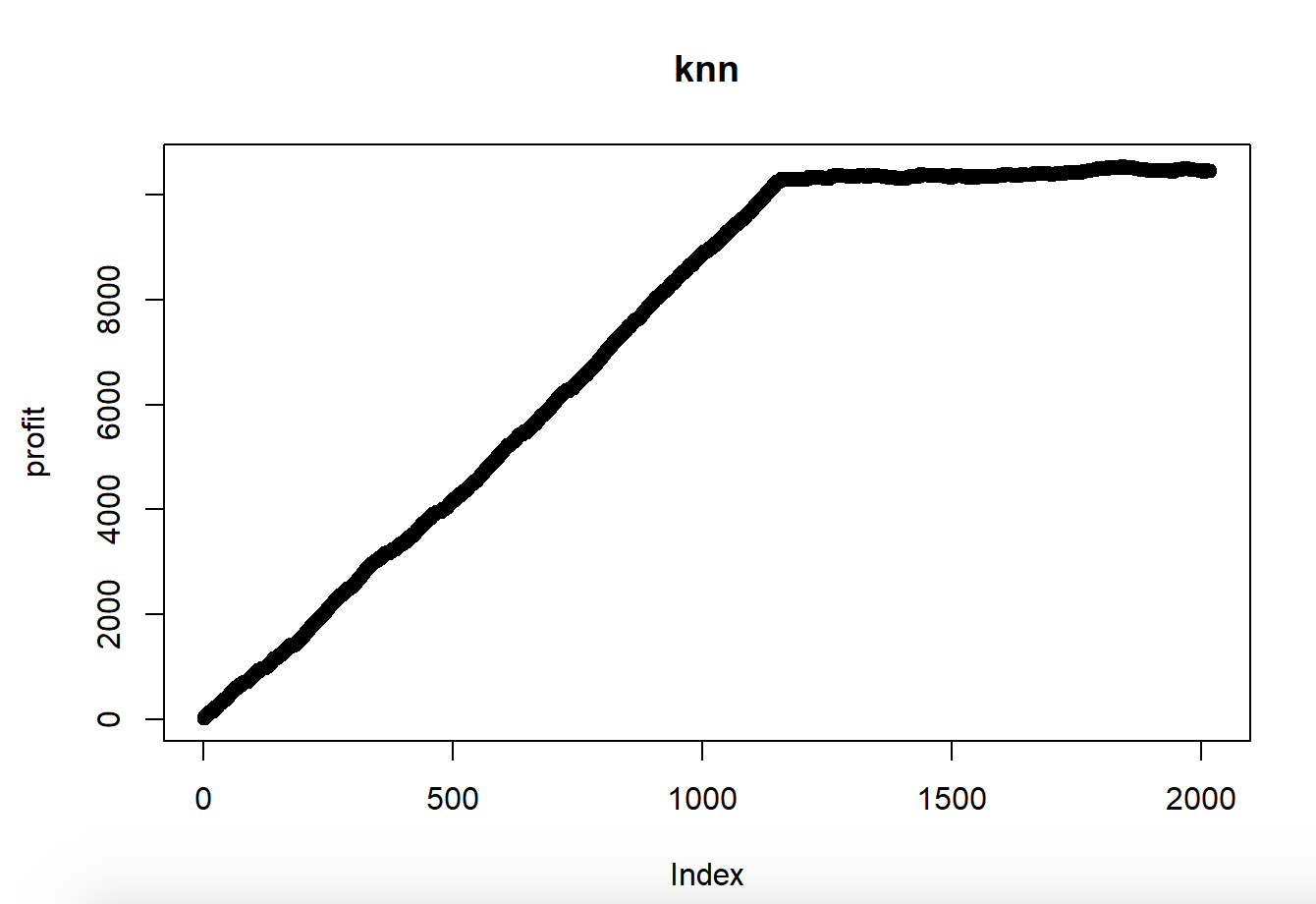
1 409 978

Accuracy

0.7869177

[1] "Maximum profit earned with logistic model is: 11407"

Model 5 (KNN model 1):

This KNN model has higher accuracy than the first three models, but the maximum profit is 10538.5 which is lower than the logistic models. The KNN model with relatively sharper change on the graphs than the first three models. The accuracy is 0.7923687 which is a little better than the logistic models. The number of actual valid donr and predicted non-valid donr has dramatically increased to 128, and number of actual non-valid donr and predicted valid donr has decreased dramatically to 291. The other two coefficients also have fairly large changes. The lower coefficients in the last column and row shows this may not be a good model since it does not have a high value on right validation on actual and predicted donr. 

y.valid.donr

pred.valid.donr 0 1

0 728 128

1 291 871

Accuracy

0.7923687

[1] "Maximum profit earned with logistic model is: 10538.5"

Model 6 (KNN model 2):

Another KNN model was built. It showed little improvement in maximum profits, accuracy and true prediction values. Actual and prediction valid donr with value 884 and non-valid donr with value 733- both higher than in model 5. Hence Model 6 is better than model 5. However, overall the value shows logistic models are better than KNN models.

y.valid.donr

pred.valid.donr 0 1

0 733 115

1 286 884

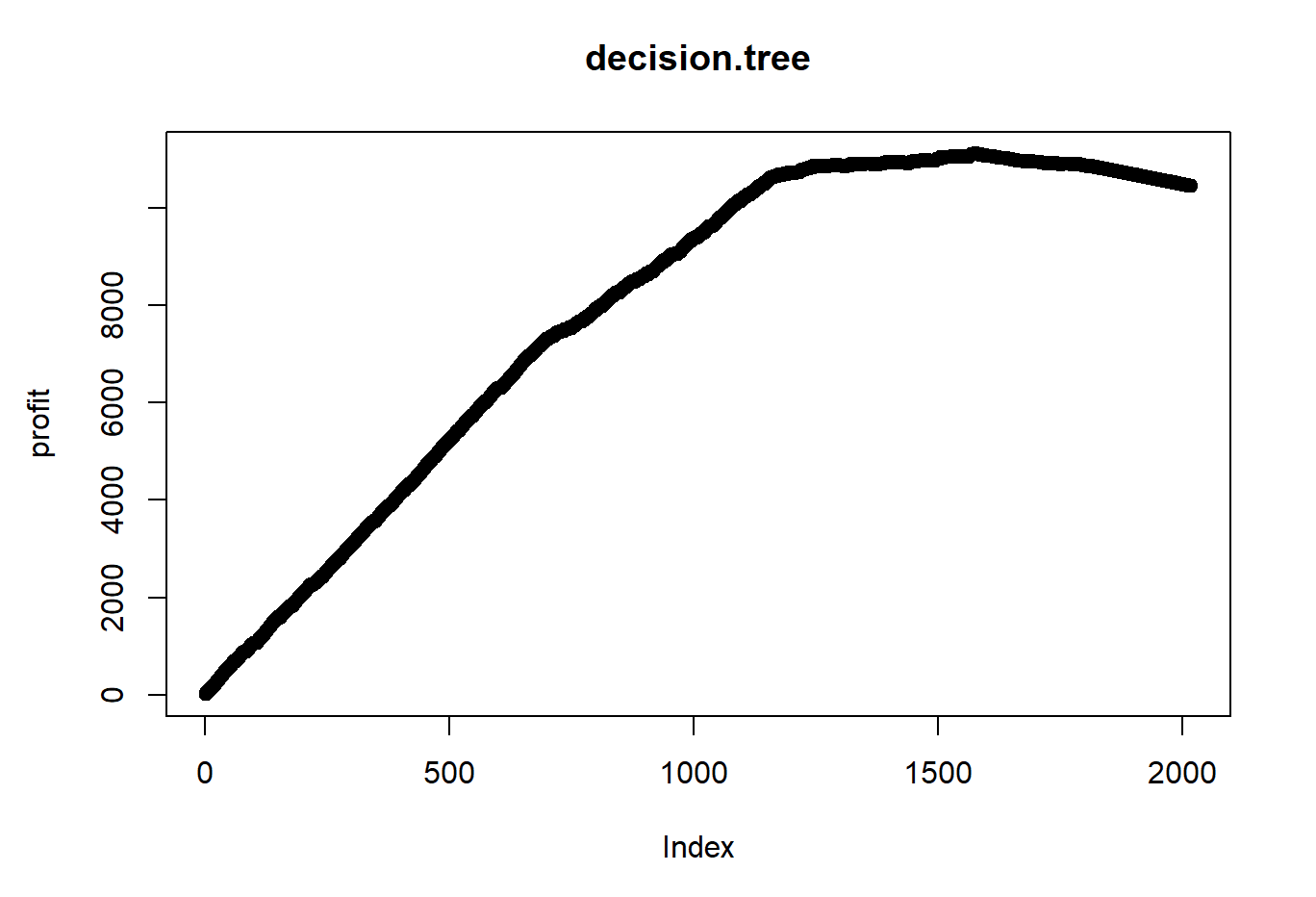
Accuracy

0.8012884

[1] "Maximum profit earned with logistic model is: 10549"

Model 7 (decision.tree model):

For this model, a decision tree was built which has a higher actual and predicted valid donr value than KNN model. However, the other models are better than this model. Also, the accuracy is lower than KNN models. The maximum profit is 11116 which is pretty high among all the models but not the best.



y.valid.donr

pred.valid.donr 0 1

0 667 73

1 352 926

Accuracy

0.7893954

[1] "Maximum profit earned with logistic model is: 11116"

Model 8 (svm model 1):

Model 8 has a similar graph as the first three models. This is not a great model, with both actual and predicted valid donr value 951 and actual and predicted non-valid donr. The accuracy and maximum profit earned with the logistic model is even lower than the decision tree model. This can not be the best model.

y.valid.donr

pred.valid.donr 0 1

0 555 48

1 464 951

Accuracy

0.7462834

[1] "Maximum profit earned with logistic model is: 10959.5"

Model 9 (svm model 2):

This model is also a svm model hence a similar graph as the last model. This model also has similar overall true prediction values with even lower accuracy which is 0.7190287. Even with a maximum profit of 11112, this model still does not have good performance.

y.valid.donr

pred.valid.donr 0 1

0 475 23

1 544 976

Accuracy

0.7190287

[1] "Maximum profit earned with logistic model is: 11112"

Model 10 (gam Model):

Model 10 has a similar true prediction value on donr with model 9 which is 1521. The accuracy is higher than svm models. The maximum profit earned is fairly high, just lower than logistic models. With that, this could be a good model.

y.valid.donr

pred.valid.donr 0 1

0 534 12

1 485 987

Accuracy

0.7537166

[1] "Maximum profit earned with logistic model is: 11367.5"

Model 11 (Random Forest Model):

Similar graph as the KNN and decision tree model graphs. Random forest model is similar to decision tree model and the principle of two models are similar. However, randomforest model have much larger value on both actual and predicted non-valid donr and lower actual and predicted valid donr value. This model has the highest accuracy value among all the models and the maximum profit is pretty high which reaches to 11113. This may be a good model and we will consider selecting this model.

y.valid.donr

pred.valid.donr 0 1

0 888 89

1 131 910

Accuracy

0.8909812

[1] "Maximum profit earned with logistic model is: 11113"

Model 12 (Boosting):

The boosting model has the highest true donr value of 897 predicted and actual non-valid donr value and 940 predicted and actual valid donr value and in total 1837 true donr value. Model 12 has the highest maximum profits which is 11947.5- this is much higher than the second best model, model 3. The accuracy is also very high. This could be the best model.

y.valid.donr

pred.valid.donr 0 1

0 897 89

1 131 940

Accuracy

0.9012093

[1] "Maximum profit earned with logistic model is: 11947.5"

**Classification Model Selection:**By comparing all the maximum profits, it shows that model 12 (Boosting model) has the highest maximum profit value (11947.5). Hence, we concluded model 12 is the best classification model.

**Prediction Models:**

Model 1 (Least Square Regression):

The summary of model 1 contains the coefficients are standard errors, t-values, and p-values which helps us to evaluate the data more efficiently. First, we look at the coefficient of the reg1, -0.039. This represents that for every 1 unit increase in reg1 there will be a 0.039 reduction in the damt. Then we look at the standard error and we find out that the inca has the largest standard error which is bad since it may have a large amount of errors. The t value measures the difference size relative to the variation in the dataset which means the larger the t-value the difference will be large. Agif has the largest t-value so this has the largest difference when compared to other variables. The p-value is 2.2e-16 which is smaller than the significance level 0.05 so the distribution of the data is significantly different from the normal distribution and their relationship to the response variable is significant. The adjusted R-squared is 0.5679, indicating that this model seems to explain 56.79% of the data variation. The standard deviation error of this model is 0.1696. And the mean prediction error for this model is 1.867 which we need to compare with the following models.

Call:

lm(formula = damt ~ reg1 + reg2 + reg3 + reg4 + home + chld +

hinc + genf + wrat + avhv + incm + inca + plow + npro + tgif +

lgif + rgif + tdon + tlag + agif, data = data.train.std.damt)

Residuals:

Min 1Q Median 3Q Max

-4.4624 -0.7966 -0.1533 0.5999 9.1086

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 14.18934 0.04735 299.660 < 2e-16 \*\*\*

reg1 -0.03923 0.03962 -0.990 0.32217

reg2 -0.07434 0.04294 -1.731 0.08352 .

reg3 0.32690 0.04041 8.089 1.04e-15 \*\*\*

reg4 0.63517 0.04158 15.275 < 2e-16 \*\*\*

home 0.23834 0.06073 3.925 8.99e-05 \*\*\*

chld -0.60477 0.03794 -15.939 < 2e-16 \*\*\*

hinc 0.50143 0.03984 12.587 < 2e-16 \*\*\*

genf -0.06318 0.02850 -2.217 0.02675 \*

wrat -0.00109 0.04150 -0.026 0.97905

avhv -0.04815 0.05136 -0.937 0.34864

incm 0.29408 0.05845 5.031 5.32e-07 \*\*\*

inca 0.04726 0.07161 0.660 0.50936

plow 0.24829 0.04341 5.719 1.23e-08 \*\*\*

npro 0.13613 0.04442 3.065 0.00221 \*\*

tgif 0.05965 0.04603 1.296 0.19517

lgif -0.05501 0.03843 -1.431 0.15251

rgif 0.51685 0.04387 11.783 < 2e-16 \*\*\*

tdon 0.07254 0.03493 2.077 0.03796 \*

tlag 0.02206 0.03366 0.655 0.51229

agif 0.67139 0.04048 16.585 < 2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.273 on 1974 degrees of freedom

Multiple R-squared: 0.5722, Adjusted R-squared: 0.5679

F-statistic: 132 on 20 and 1974 DF, p-value: < 2.2e-16

Adjusted R^2:

[1] 0.5678677

Mean Prediction Error:

[1] 1.866657

Standard Deviation Error:

[1] 0.1695524

Model 2 (best subset selection with *19*-fold cross-validation):

The summary of best subset selection did not offer much useful information. By looking at the graph and the R-squared values shown below, we find that the min RSS is 0.5722019, max adjusted R-squared is 0.5719149, min cp is 0.5710447 and min BIC is 0.5685088. The standard deviation error is 0.1693538 which is bigger than model 1. The mean squared error for this model is 1.857947 which is smaller than the model 1 and Adjusted R-square is larger than model 1 so this is better.

r^2:

[1] 0.2718288 0.3745233 0.4406249 0.4760923 0.5102874 0.5376199 0.5471627

[8] 0.5562790 0.5647500 0.5685088 0.5696678 0.5705746 0.5710447 0.5713117

[15] 0.5716925 0.5719149 0.5720135 0.5721088 0.5722019

Minimum RSS:

[1] 19

Maximum adjusted R^2:

[1] 16

Minimum cp:

[1] 13

Minimum BIC:

[1] 10

(Intercept) reg3 reg4 chld hinc

14.2931010 0.3736171 0.6889575 -0.5731359 0.5028017

rgif agif

0.4766170 0.6545193

Errors:

[1] 3.008661 2.499722 2.273558 2.149864 2.055742 1.980266 1.953428

[8] 1.905913 1.870818 1.857947 1.868510 1.870988 1.859918 1.858703

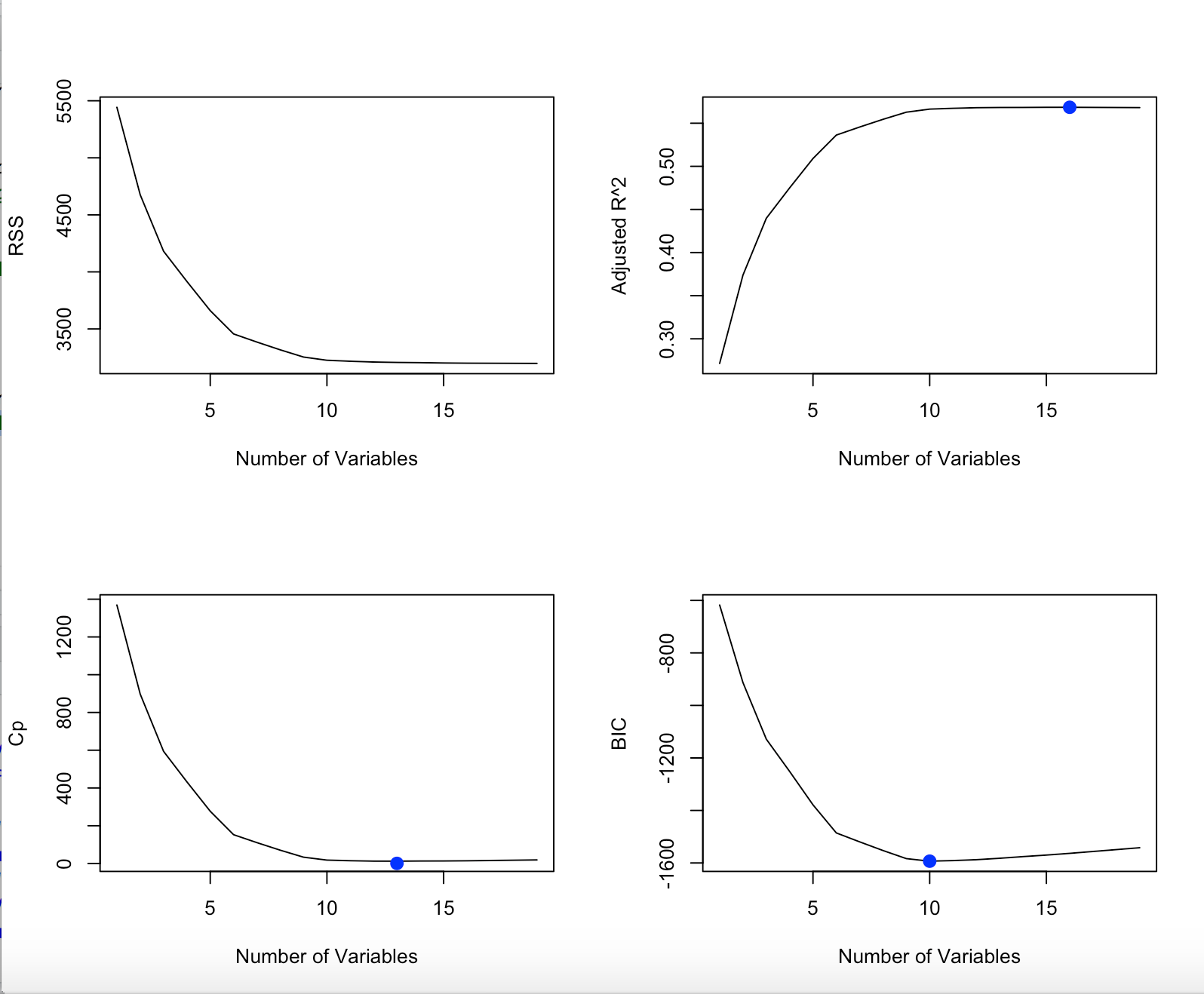
[15] 1.862669 1.861431 1.865514 1.867564 1.866597

Minimum error:

[1] 10

Standard Deviation Error:

[1] 0.1693538



Model 3 (best subset selection with *10*-fold cross-validation):

The standard deviation error is 0.1693538 which is bigger than model 1, a point of criticism for this model. The mean squared error for model 3 is 1.857947 which is the same as model 2 so this model is better than model 1.

Mean squared error:

[1] 1.857947

Standard Deviation Error:

[1] 0.1693538

Model 4 (bagging model 1):

This is a bagging model and this model is used to reduce the variance, but the main idea of bagging is that based on a large number of bootstrap samples it can create an aggregate fitted value. The random forest is used to lower the variance among models because it averages large amounts of trees. Bagging may have highly correlated predictors and rf has good predictive accuracy. As we look at the value below, the mean of squared residuals is 1.484916 and the variance explained is 60.37% which is pretty large, a good thing for this model. The standard deviation error is 0.1762078 which is bigger than model 3, a point of criticism for this model. The mean predicted error is 1.704569 which is smaller than the model 1, 2 and 3 so this is the best model so far. We still need to compare the following models.

Call:

randomForest(formula = damt ~ reg1 + reg2 + reg3 + reg4 + home + chld + hinc + genf + wrat + avhv + incm + inca + plow + npro + tgif + lgif + rgif + tdon + tlag + agif, data = data.train.std.damt, mtry = 20, importance = TRUE)

Type of random forest: regression

Number of trees: 500

No. of variables tried at each split: 20

Mean of squared residuals: 1.484916

% Var explained: 60.37

Mean Predicted Error:

[1] 1.704569

Standard Deviation Error:

[1] 0.1762078

Model 5 (bagging model 2):

The mean squared residuals for this model is 1.47818 and the variance explained is 60.55% which is pretty large and it’s larger than model 4, making it a better model than model 4. The standard deviation error is 0.173091 which is larger than the model 4 but the mean predicted error is 1.666061 which is smaller than the model 1, 2, 3 and 4 so this is the best model so far. We still need to compare the following models and find the best model.

Call:

randomForest(formula = damt ~ reg1 + reg2 + reg3 + reg4 + home + chld + hinc + genf + wrat + avhv + incm + inca + plow + npro + tgif + lgif + rgif + tdon + tlag + agif, data = data.train.std.damt, mtry = 5, ntree = 500)

Type of random forest: regression

Number of trees: 500

No. of variables tried at each split: 5

Mean of squared residuals: 1.47818

% Var explained: 60.55

Mean Predicted Error:

[1] 1.666061

Standard Deviation Error:

[1] 0.173091

Model 6 (Randomforest model):

The mean squared residuals for this model is 1.470602 and the variance explained is 60.75% which is pretty large and it’s larger than model 5 so this is better than model 5. The standard deviation error is 0.1732489 which is a little larger than the model. The mean predicted error is 1.671982 which is greater than the model 5 so this is not the best model.

Call:

randomForest(formula = damt ~ reg1 + reg2 + reg3 + reg4 + home + chld + hinc + genf + wrat + avhv + incm + inca + plow + npro + tgif + lgif + rgif + tdon + tlag + agif, data = data.train.std.damt, mtry = 6, importance = TRUE)

Type of random forest: regression

Number of trees: 500

No. of variables tried at each split: 6

Mean of squared residuals: 1.470602

% Var explained: 60.75

Mean Predicted Error:

[1] 1.671982

Standard Deviation Error:

[1] 0.1732489

Model 7 (Principal Components Regression):

For this model, we use the package “pls” and we have all the % variance explained, if we use the first principal component, we can explain 16.053% of the variation in the response variable. By adding in the second principal component, we can explain 27.87% of the variation in the response variable which increases, which is the desired effect of this addition. The standard deviation error is 0.1919335 which is larger than the other models. However, the mean predicted error is 1.597288 which is smaller than the 5 so this is the best model but we still need to compare other models.

Data: X dimension: 1995 20

Y dimension: 1995 1

Fit method: svdpc

Number of components considered: 20

TRAINING: % variance explained

1 comps 2 comps 3 comps 4 comps 5 comps 6 comps

X 16.05287 27.87 36.69 44.97 51.07 56.76

damt 0.04491 28.44 28.52 36.51 46.57 47.06

7 comps 8 comps 9 comps 10 comps 11 comps 12 comps

X 62.31 67.55 72.67 77.63 82.42 87.06

damt 48.95 49.27 49.40 49.60 49.80 51.22

13 comps 14 comps 15 comps 16 comps 17 comps 18 comps

X 90.43 92.69 94.79 96.28 97.61 98.65

damt 52.03 52.03 56.19 56.40 56.53 56.54

19 comps 20 comps

X 99.58 100.00

damt 57.19 57.22

Mean Predicted Error:

[1] 1.597288

Standard Deviation Error:

[1] 0.1919335

Model 8 (Partial Least Squares):

For model 8, we still use packages “pls” like model 7. If we use the first principal component, we can explain 10.8% of the variation in the response variable. By adding in the second principal component, we can explain 17.96% of the variation in the response variable. This is lower than model 7. The standard deviation error is 0.1705767 which is a little larger than the model 6. Also, the mean predicted error is 1.612741 which is greater than the 7 so this model does not perform as well.

Data: X dimension: 1995 20

Y dimension: 1995 1

Fit method: kernelpls

Number of components considered: 20

TRAINING: % variance explained

1 comps 2 comps 3 comps 4 comps 5 comps 6 comps

X 10.80 17.96 25.31 36.36 43.66 47.34

damt 49.65 55.30 56.51 56.93 57.11 57.18

7 comps 8 comps 9 comps 10 comps 11 comps 12 comps

X 50.51 56.08 62.01 65.06 68.43 70.04

damt 57.21 57.22 57.22 57.22 57.22 57.22

13 comps 14 comps 15 comps 16 comps 17 comps 18 comps

X 74.73 77.80 81.48 85.68 88.23 90.50

damt 57.22 57.22 57.22 57.22 57.22 57.22

19 comps 20 comps

X 94.94 100.00

damt 57.22 57.22

Mean Predicted Error:

[1] 1.612741

Standard Deviation Error:

[1] 0.1705767

Model 9 (Boosting):

For model 9, we used 5000 n.trees, then we got var and rel.inf. The rel.inf for agif is 14.1012 and for rgif is 12.9056. The standard deviation error is 0.1705982, which is larger than that of model 8. The mean predicted error for this model is 1.5398 which is smaller than the model 8 which means it’s better than other models but we will still compare the other models. According to the graph, agif has the largest relative influence and lgif has second largest relative influence.

var rel.inf

agif agif 14.1012124

rgif rgif 12.9055819

lgif lgif 10.0162333

avhv avhv 7.3083081

tgif tgif 7.0205169

npro npro 6.1455127

incm incm 5.4602817

chld chld 5.4451043

reg4 reg4 5.0240751

inca inca 4.6624669

wrat wrat 4.4197590

plow plow 3.8650056

hinc hinc 3.7144915

tdon tdon 3.3322890

tlag tlag 1.9924771

reg3 reg3 1.4944818

reg2 reg2 1.1414594

home home 0.8719222

genf genf 0.5947646

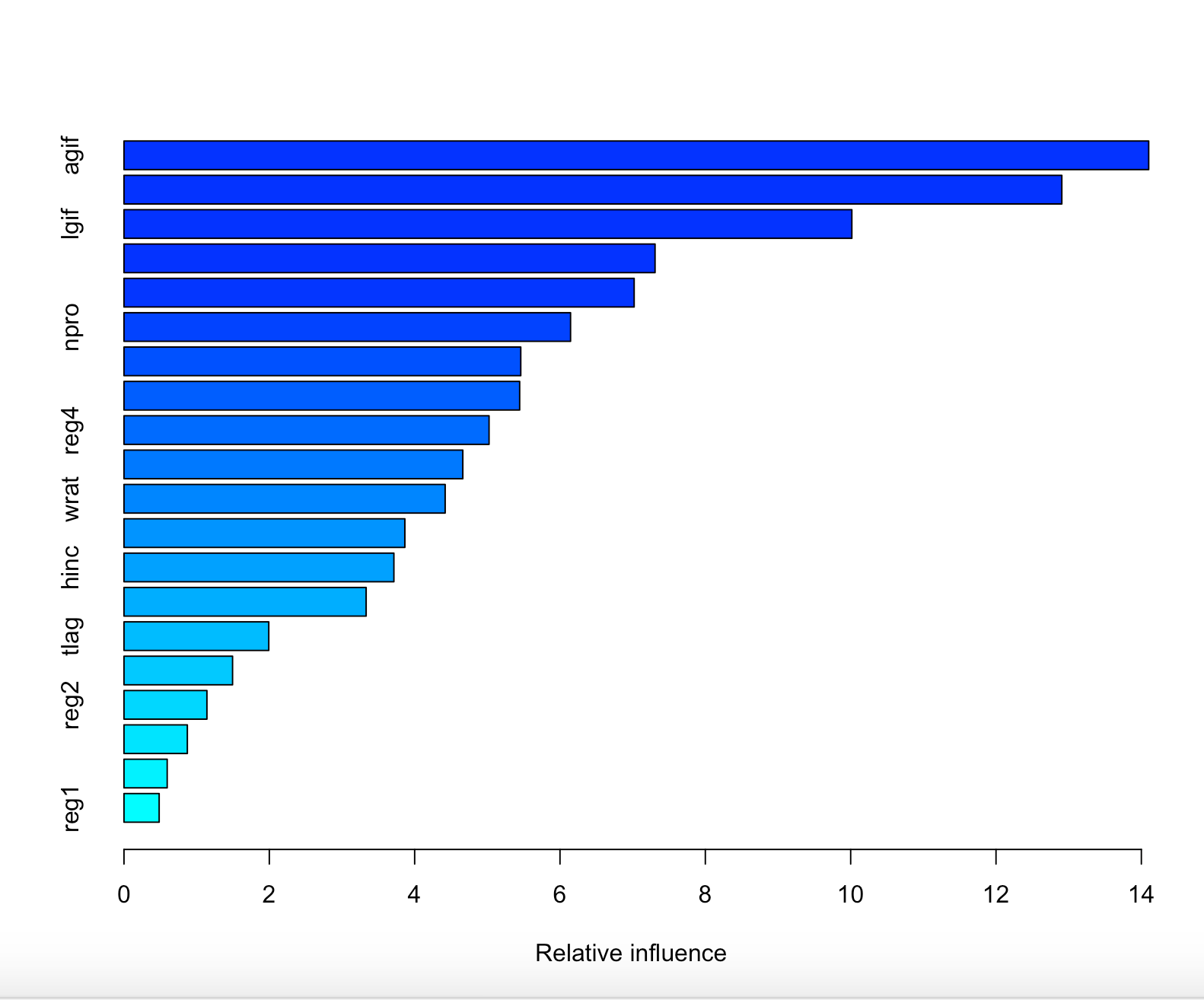
reg1 reg1 0.4840565

Mean Predicted Error:

[1] 1.539772

Standard Deviation Error:

[1] 0.1705982



Model 10 (Boosting model 2):

The standard deviation error is 0.1613 which is smaller than the model 9. Model 10 uses n.trees 100 instead of 5000 and the mean predicted error for this model is 1.403496 which is smaller than the model 9 which means it’s better than all other models but we will compare it with other models.

gbm(formula = damt ~ reg1 + reg2 + reg3 + reg4 + home + chld +

hinc + genf + wrat + avhv + incm + inca + plow + npro + tgif +

lgif + rgif + tdon + tlag + agif, distribution = "gaussian",

data = data.train.std.damt, n.trees = 100, interaction.depth = 4,

shrinkage = 0.1, verbose = F)

A gradient boosted model with gaussian loss function.

100 iterations were performed.

There were 20 predictors of which 20 had non-zero influence.

Mean Predicted Error:

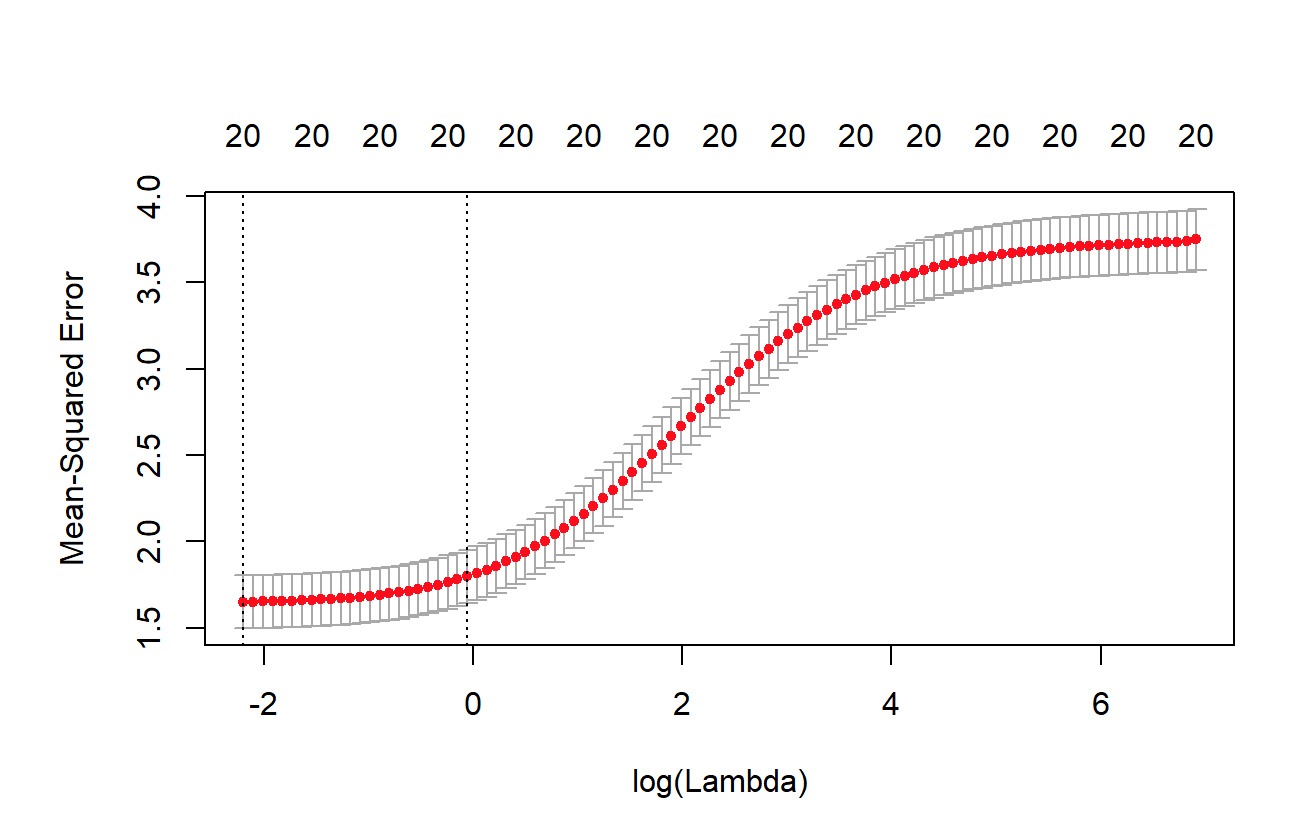
[1] 1.403496

Standard Deviation Error:

[1] 0.1613256

Model 11 (ridge):

The graph below shows the correlation between the mean squared error and lambda. The standard deviation error is 0.591238 which is the smallest so far. For model 11, our best lambda is 0.11076 and we will use this value to calculate the mean predicted error which is 1.871521. This MSE is bigger than model 1 so it is not the best.



Best lameda:

[1] 0.1107589

Mean Predicted Error:

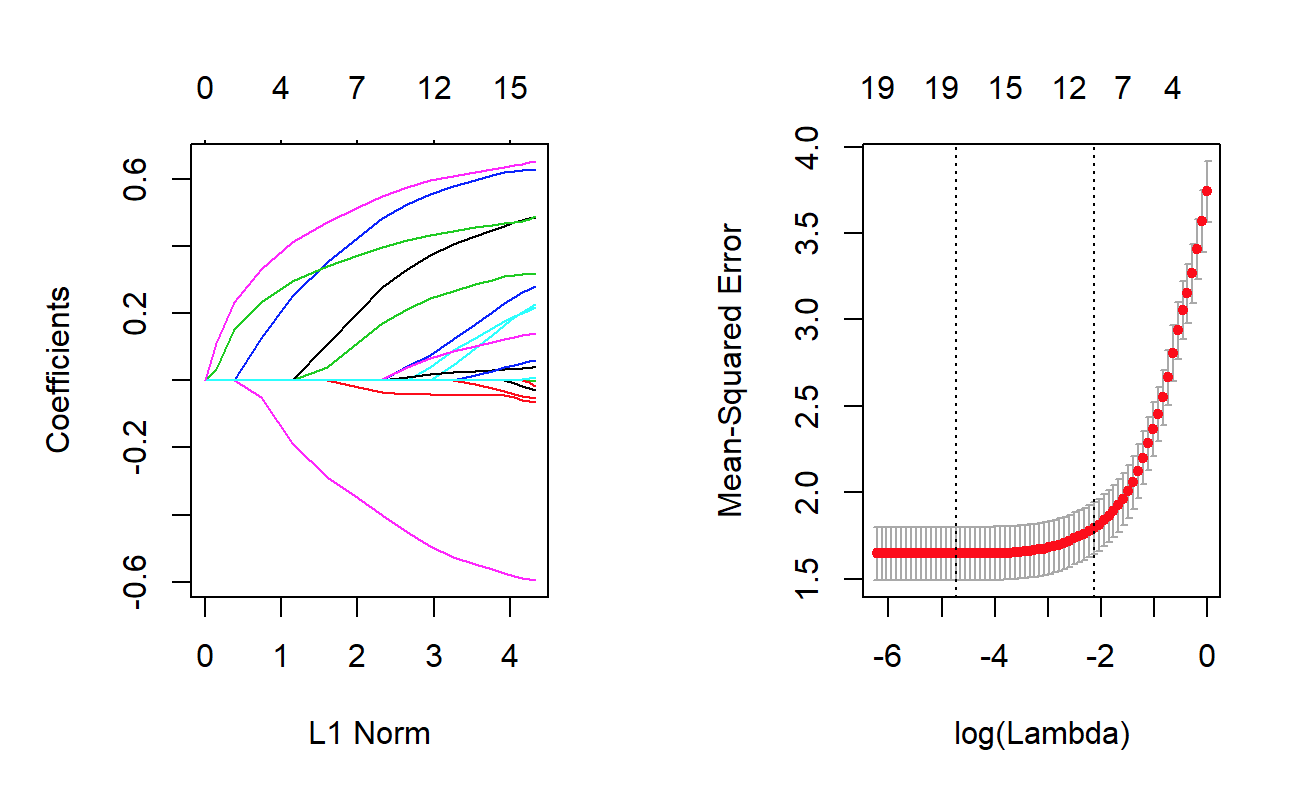
[1] 1.871521

Standard Deviation Error:

[1] 0.591238

Model 12 (lasso model):

The left graph shows the relation between coefficients and L1 Norm. The right graph shows the correlation between the mean squared error and lambda. The standard deviation error is 0.169354 which is fairly large. For model 12, our best lameda is 0.00878 which is smaller than model 10 and we will use this value to calculate the mean predicted error which is 1.85981. This MSE is smaller than the model 10 but still so large so this model is not the best. The following values are lasso coefficients.



Best lameda:

[1] 0.00877745

Mean Predicted Error:

[1] 1.85981

(Intercept) (Intercept) reg1 reg2 reg3 reg4 home chld hinc genf

14.194530356 0.000000000 -0.027292544 -0.065641610 0.319922718 0.630135530 0.217520541 -0.595569736 0.488308561 -0.052798906

wrat avhv incm inca plow npro tgif lgif rgif tdon

0.000000000 -0.001084543 0.282511967 0.000000000 0.226566028 0.138616732 0.043587827 -0.017263891 0.490316143 0.061421249

tlag agif

0.011348016 0.655682444

Standard Deviation Error:

0.169354

**Results:**

Classification Models: Prediction Models:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Maximize profits |  | Mean Prediction Error | Standard Deviation Error |
| Model 1 | 11040 | Model 1 | 1.866657 | 0.1695524 |
| Model 2 | 11111 | Model 2 | 1.857947 | 0.1693538 |
| Model 3 | 11423.5 | Model 3 | 1.857947 | 0.1693538 |
| Model 4 | 11407 | Model 4 | 1.704569 | 0.1762078 |
| Model 5 | 10538.5 | Model 5 | 1.666061 | 0.173091 |
| Model 6 | 10549 | Model 6 | 1.671982 | 0.1732489 |
| Model 7 | 11116 | Model 7 | 1.597288 | 0.1919335 |
| Model 8 | 10959.5 | Model 8 | 1.612741 | 0.1705767 |
| Model 9 | 11112 | Model 9 | 1.539772 | 0.1705982 |
| Model 10 | 11367.5 | Model 10 | 1.403496 | 0.1613256 |
| Model 11 | 11113 | Model 11 | 1.871521 | 0.591238 |
| Model 12 | 11947.5 | Model 12 | 1.85981 | 0.169354 |

For classification models, by comparing all the data above, we could find that model 12, which is a boosting model, has the highest maximum profits and hence is the best model. For prediction models, model 10 which is also a boosting model, with much smaller n.trees, is the best model since it has the lowest mean prediction error and standard deviation value.

After we look at all the models above, we find that for both classification and prediction models, boosting is the best model. For donar classification, the boosting model achieves maximum profit is $11947.5. For the prediction model, it has 1.403496 mean prediction error, and standard error 0.1624. The response rate is 15.05%, and the recent mailing records show that the overall response rate is 10%. The boosting model increases the response rate by 50.5% and each donor donates $14.50 on average and each mailing costs $2.00. The average donation amount based on all of the data set and the average that is equal to 1, then we have the average donation amount is $14.42 which is really close to 14.5 so both boosting classification model and prediction model works pretty well. If we calculate the current mailing plan we have expected profit is $14.5 \* 0.1-2 = -0.55. Based on the boosting model the expected profit is $14.42 \*0.1505 -2 = 0.17021. For the current mailing plan, people will lose the profits. However, if people use the boosting models they will get the positive profits which is better.

Mailing and maximum profit:

The adjust for this mailing rate we use 0.8/(0.5/0.1) = 0.16 and adjust the “non-mailing rate” we use is (1-0.8.1)= 0.44 and the scale mailing rats is a proportion which is 0.16/(0.16 +044) = 0.267. Hence, the optimal test mailing rate is 0.267.

Number of Mailings and Maximum Profit:

[1] 1386.0 11947.5

[1] "Response rate for profit calculation (based on weighted sampling) for the test data is 0.267 or 26.7%."

**Conclusion:**

With the several models built in this project there were several similarities and differences. Ultimately, our goal to create a model to better predict a profit that will help the charitable organization with fundraising was reached. The different experimentation leads to a clear conclusion that a boosting model leads to the highest profit.

Going forward, a point that could lead to improvement is to increase the size of the data set. The data set only represents existing donors. However, this is a bit limiting because it does not show the possibility of expanding the charity’s donor base to help increase donations. If there were to be two data sets, one of existing donors and one of individuals who were “cold called”, we could better predict the response rate and ultimately the profit of the group of people who are cold called. In this sense, “cold called,” is representing individuals that the charity is reaching out to for the first time by mail and the charity has no previous history of donation with. While this is likely to be far less successful than the donations from existing donors, it will still add some more money to the bottom line.

Another way this could be improved is to add a variable that represents the economy in the year of donation. People are much more likely to donate in a booming economy than in a recession, so to add this index, one could better predict donations and profit. This would be very challenging to do, but the charity could begin to add this into their data sets and then turn this over to data analysts.