Partha Sai Madallapalli

Fundraising Write-Up

Direct-Mail Fundraising- Predictive Modelling Final Project

Business Objectives and Goals:

Using the facts provided, we take a sample of the data to **develop a classification model that can effectively capture donors so that the expected net profit is maximized**. Weighted sampling was used, under-representing the non-responders so that the sample has equal numbers of donors and non-donors**. What percentage of your data would you recommend for a mailing campaign?**

Data Sources and Data used:

The fundraising training file contains 3,000 records with approximately 50% donors (target = Donor) and 50% non-donors (target = No Donor). The descriptions for the 22 are listed below.

Variable	Description
zip	Zip code group (zip codes were grouped into five groups; Yes = the potential donor belongs to this zip group.) 00000–19999 ⇒ zipconvert1 20000–39999 ⇒ zipconvert2 40000–59999 ⇒ zipconvert3 60000–79999 ⇒ zipconvert4 80000–99999 ⇒ zipconvert5
homeowner	Yes = homeowner, No = not a homeowner
num_child	Number of children
income	Household income
female	No = male, Yes = female
wealth	Wealth rating uses median family income and population statistics from each area to index relative wealth within each state. The segments are denoted 0 to 9, with 9 being the highest-wealth group and zero the lowest. Each rating has a different meaning within each state

home_value	Average home value in potential donor's neighborhood in hundreds of dollars
med_fam_inc	Median family income in potential donor's neighborhood in hundreds of dollars
avg_fam_inc	Average family income in potential donor's neighborhood in hundreds
pct_lt15k	Percent earning less than $15K in potential donor's neighborhood
num_prom	Lifetime number of promotions received to date
lifetime_gifts	Dollar amount of lifetime gifts to date
largest_gift	Dollar amount of largest gift to date
last_gift	Dollar amount of most recent gift
months_since_donate	Number of months from last donation to July 2018
time_lag	Number of months between first and second gift
avg_gift	Average dollar amount of gifts to date
target	Outcome variable: binary indicator for response Yes = donor, No = non-donor

The reason behind using weighted sampling to produce a training set with equal numbers of donors and non-donors? And why not use a simple random sample from the original dataset? Is because **by creating a training set with an equal number of donors and non-donors using weighted sampling, it is possible to make sure that the model is trained on a balanced dataset, hence reducing bias towards the majority class. This method guarantees that the model gains an equivalent amount of knowledge from both classes, which improves efficiency and generalization, particularly when the classes are unbalanced.**

Type of Analysis performed:

This thorough analysis covers all the stages involved in predictive modelling, such as feature engineering, data partitioning, data summarization, random forests, KNN, logistic regression, and the Naive Bayes model. It also includes metrics for evaluation and interpretation. Let us examine it:

1. **Data Partitioning:** An 80-20 rule with a seed(12345) is used to divide the dataset into training and testing sets in order to ensure reproducibility.
2. **Exploratory data analysis:** Examine the predictors and evaluate their association with the response variable. Which might be good candidate predictors, feature Engineering, data summarization, visualization, addressing multicollinearity with correlation plot.

3. **Multicollinearity Check:** To determine whether predictor variables are multicollinear, utilize the Variance Inflation Factor (VIF). Variables with high VIF values are eliminated after the VIF values are analyzed.
4. **Model Building:** A predictive model is first constructed using logistic regression. Next, utilizing the Akaike Information Criterion (AIC), stepwise backward elimination is used to refine the model. This procedure aids in the selection of features while taking multicollinearity into account. thereafter, KNN, Naive Bayes model, and random forests were constructed.
5. **Model Evaluation:** The model's performance is evaluated using metrics such as accuracy, precision, recall (sensitivity), and F1 score. A confusion matrix is generated to visualize the model's predictive performance. Receiver Operating Characteristic (ROC) curve and Area Under the Curve (AUC) are used to evaluate the logistic regression model's discrimination ability.
6. **Select best model:** Based upon the above metrics which has the best accuracy I will select that for testing future fundraising test.

Exclusions

As part of feature engineering, I have removed “zipconvert2” as the VIF co-efficients are getting influenced by this variable due to high correlation. VIF co-efficient values below 5 indicate low multicollinearity and are generally acceptable for any data analysis and conversely, VIF co-efficient values above 5 indicate moderate to high multicollinearity, suggesting that the variable would be removed from further analysis. From the above the analysis variables “med_fam_inc”, and “avg_fam_inc” have high VIF co-efficient and even from the exploratory data analysis also suggest that these variables are highly correlated(multicollinearity).

Removed the variables which has the least significant using Backward elimination of AIC value. Post stepwise selection we are left with 5 features(homeowner + num_child + income + last_gift + months_since_donate) we can use this for further prediction of the target(response variable).

Multicollinearity is satisfied here, as VIF for the left over features(homeowner + num_child + income + last_gift + months_since_donate) post step wise(backward) selection has values less than 5(default value for VIF).

Variable transformations

The following variables are transformed using as.factor() function, such that zipconvert(2,3,4,5), and target variable:

Below is the structure post variable transformation.

```
'data.frame':  3000 obs. of  21 variables:
 $ zipconvert2        : Factor w/ 2 levels "No","Yes": 2 1 1 1 1 1 1 2 1 2
...
 $ zipconvert3        : Factor w/ 2 levels "No","Yes": 1 1 1 2 2 1 1 1 1 1
...
 $ zipconvert4        : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 1 1 1
...
 $ zipconvert5        : Factor w/ 2 levels "No","Yes": 1 2 2 1 1 2 1 1 2 1
...
```

```
 $ homeowner           : Factor w/ 2 levels "No","Yes": 2 1 2 2 2 2 2 2 2 2
...
 $ num_child           : int  1 2 1 1 1 1 1 1 1 1 ...
 $ income              : int  1 5 3 4 4 4 4 4 4 1 ...
 $ female              : Factor w/ 2 levels "No","Yes": 1 2 1 1 2 2 1 2 2 2
...
 $ wealth              : int  7 8 4 8 8 8 5 8 8 5 ...
 $ home_value          : int  698 828 1471 547 482 857 505 1438 1316 428
...
 $ med_fam_inc         : int  422 358 484 386 242 450 333 458 541 203 ...
 $ avg_fam_inc         : int  463 376 546 432 275 498 388 533 575 271 ...
 $ pct_lt15k           : int  4 13 4 7 28 5 16 8 11 39 ...
 $ num_prom            : int  46 32 94 20 38 47 51 21 66 73 ...
 $ lifetime_gifts      : num  94 30 177 23 73 139 63 26 108 161 ...
 $ largest_gift        : num  12 10 10 11 10 20 15 16 12 6 ...
 $ last_gift           : num  12 5 8 11 10 20 10 16 7 3 ...
 $ months_since_donate: int  34 29 30 30 31 37 37 30 31 32 ...
 $ time_lag            : int  6 7 3 6 3 3 8 6 1 7 ...
 $ avg_gift            : num  9.4 4.29 7.08 7.67 7.3 ...
 $ target              : Factor w/ 2 levels "Donor","No Donor": 1 1 2 2 1 1
1 2 1 1 ...
```

Business inputs

I have brief introduction of the business inputs about the meta data and how it is important for the Fundraising output. A national veterans' organization wishes to develop a predictive model to improve the cost-effectiveness of their direct marketing campaign. The organization, with its in-house database of over 13 million donors, is one of the largest direct-mail fundraisers in the United States. According to their recent mailing records, the overall response rate is 5.1%. Out of those who responded (donated), the average donation is $13.00. Each mailing, which includes a gift of personalized address labels and assortments of cards and envelopes, costs $0.68 to produce and send.

Methodology used, background, benefits

Following data analysis is used, as part of the data summarization, I have used a function called "skim" to visualize and summarize the data together, additionally I have checked for data skewness, and pairplot for the high correlated variables, and various exploratory data analysis (**EDA**).

```
Number of missing values in Fundraising_data: 0
Number of missing values in Future_fundraising: 0
```

No missing data identified in the dataset's.

```
##  zipconvert2       zipconvert3       zipconvert4       zipconvert5
##  Length:3000       Length:3000       Length:3000       Length:3000
##  Class :character  Class :character  Class :character  Class :character
##  Mode  :character  Mode  :character  Mode  :character  Mode  :character
##
##
##
##   homeowner          num_child        income        female
##  Length:3000       Min.   :1.000  Min.   :1.000  Length:3000
##  Class :character  1st Qu.:1.000  1st Qu.:3.000  Class :character
##  Mode  :character  Median :1.000  Median :4.000  Mode  :character
##                    Mean   :1.069  Mean   :3.899
##                    3rd Qu.:1.000  3rd Qu.:5.000
##                    Max.   :5.000  Max.   :7.000
##      wealth       home_value     med_fam_inc     avg_fam_inc
##  Min.   :0.000  Min.   :   0.0  Min.   :   0.0  Min.   :   0.0
##  1st Qu.:5.000  1st Qu.: 554.8  1st Qu.: 278.0  1st Qu.: 318.0
##  Median :8.000  Median : 816.5  Median : 355.0  Median : 396.0
##  Mean   :6.396  Mean   :1143.3  Mean   : 388.4  Mean   : 432.3
##  3rd Qu.:8.000  3rd Qu.:1341.2  3rd Qu.: 465.0  3rd Qu.: 516.0
##  Max.   :9.000  Max.   :5945.0  Max.   :1500.0  Max.   :1331.0
##    pct_lt15k        num_prom     lifetime_gifts    largest_gift
##  Min.   : 0.00  Min.   : 11.00  Min.   :  15.0  Min.   :   5.00
##  1st Qu.: 5.00  1st Qu.: 29.00  1st Qu.:  45.0  1st Qu.:  10.00
##  Median :12.00  Median : 48.00  Median :  81.0  Median :  15.00
##  Mean   :14.71  Mean   : 49.14  Mean   : 110.7  Mean   :  16.65
##  3rd Qu.:21.00  3rd Qu.: 65.00  3rd Qu.: 135.0  3rd Qu.:  20.00
##  Max.   :90.00  Max.   :157.00  Max.   :5674.9  Max.   :1000.00
##    last_gift      months_since_donate    time_lag         avg_gift
##  Min.   :  0.00  Min.   :17.00       Min.   : 0.000  Min.   :  2.139
##  1st Qu.:  7.00  1st Qu.:29.00       1st Qu.: 3.000  1st Qu.:  6.333
##  Median : 10.00  Median :31.00       Median : 5.000  Median :  9.000
##  Mean   : 13.48  Mean   :31.13       Mean   : 6.876  Mean   : 10.669
##  3rd Qu.: 16.00  3rd Qu.:34.00       3rd Qu.: 9.000  3rd Qu.: 12.800
##  Max.   :219.00  Max.   :37.00       Max.   :77.000  Max.   :122.167
##     target
##  Length:3000
##  Class :character
##  Mode  :character
##
##
##
```

skim(Fundraising_data)

Data summary

Name	Fundraising_data
Number of rows	3000
Number of columns	21
Column type frequency:	
factor	7
numeric	14
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
zipconvert2	0	1	FALSE	2	No: 2352, Yes: 648
zipconvert3	0	1	FALSE	2	No: 2449, Yes: 551
zipconvert4	0	1	FALSE	2	No: 2357, Yes: 643
zipconvert5	0	1	FALSE	2	No: 1846, Yes: 1154
homeowner	0	1	FALSE	2	Yes: 2312, No: 688
female	0	1	FALSE	2	Yes: 1831, No: 1169
target	0	1	FALSE	2	No : 1501, Don: 1499

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
num_child	0	1	1.07	0.35	1.00	1.00	1.0	1.00	5.00	
income	0	1	3.90	1.64	1.00	3.00	4.0	5.00	7.00	
wealth	0	1	6.40	2.55	0.00	5.00	8.0	8.00	9.00	
home_value	0	1	1143.27	952.15	0.00	554.75	816.5	1341.25	5945.00	
med_fam_inc	0	1	388.36	173.73	0.00	278.00	355.0	465.00	1500.00	
avg_fam_inc	0	1	432.31	168.90	0.00	318.00	396.0	516.00	1331.00	
pct_lt15k	0	1	14.71	12.11	0.00	5.00	12.0	21.00	90.00	
num_prom	0	1	49.14	22.78	11.00	29.00	48.0	65.00	157.00	
lifetime_gifts	0	1	110.74	149.38	15.00	45.00	81.0	135.00	5674.90	
largest_gift	0	1	16.65	22.52	5.00	10.00	15.0	20.00	1000.00	
last_gift	0	1	13.48	10.48	0.00	7.00	10.0	16.00	219.00	
months_since_donate	0	1	31.13	4.10	17.00	29.00	31.0	34.00	37.00	
time_lag	0	1	6.88	5.60	0.00	3.00	5.0	9.00	77.00	
avg_gift	0	1	10.67	7.45	2.14	6.33	9.0	12.80	122.17	

Interpretation:

I used this summary of the data, to understand the descriptive statistics, data skewness through histogram and number of numeric and factor conversion data type variables, missing data by variable wise. For example, I notice that income variable has the right skewness, we can notice this kind of histogram and descriptive statistics in one table.

Partha Sai Madallapalli

Target variable split and balance check:

```
##
##    Donor No Donor
## 49.96667 50.03333
```

Interpretation:

Here the target variable is equally 50-50% split and understood that data is balanced.

Checking for data skewness:

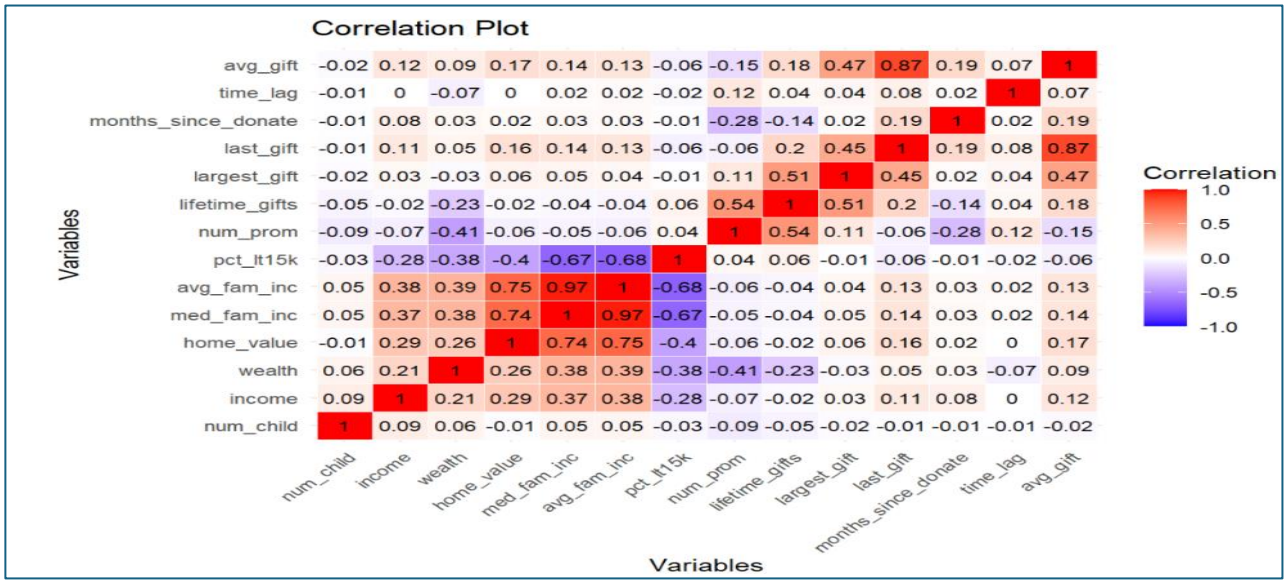
```
##          num_child             income             wealth         home_value
##        5.983103274      -0.002241026      -1.200892282        2.233753643
##           med_fam_inc        avg_fam_inc         pct_lt15k           num_prom
##          1.893381298        1.211320870        1.313551667        0.690437980
##      lifetime_gifts       largest_gift         last_gift months_since_donate
##         19.451027090       30.019347022        5.565885243       -1.006139839
##            time_lag           avg_gift
##         2.853331062        4.806070417
```

Interpretation:

variables num_child, home_value, med_fam_inc, avg_fam_inc, pct_lt15k, lifetime_gifts, largest_gift, last_gift, time_lag, and avg_gift are positively skewed to Right skewness (tail to the right), and variables "wealth" and "months_since_donate" left skewness (tail to the left) or negative skewness. The variable is often regarded as severely skewed if the absolute value of skewness is more than 1.

Correlation plot:-

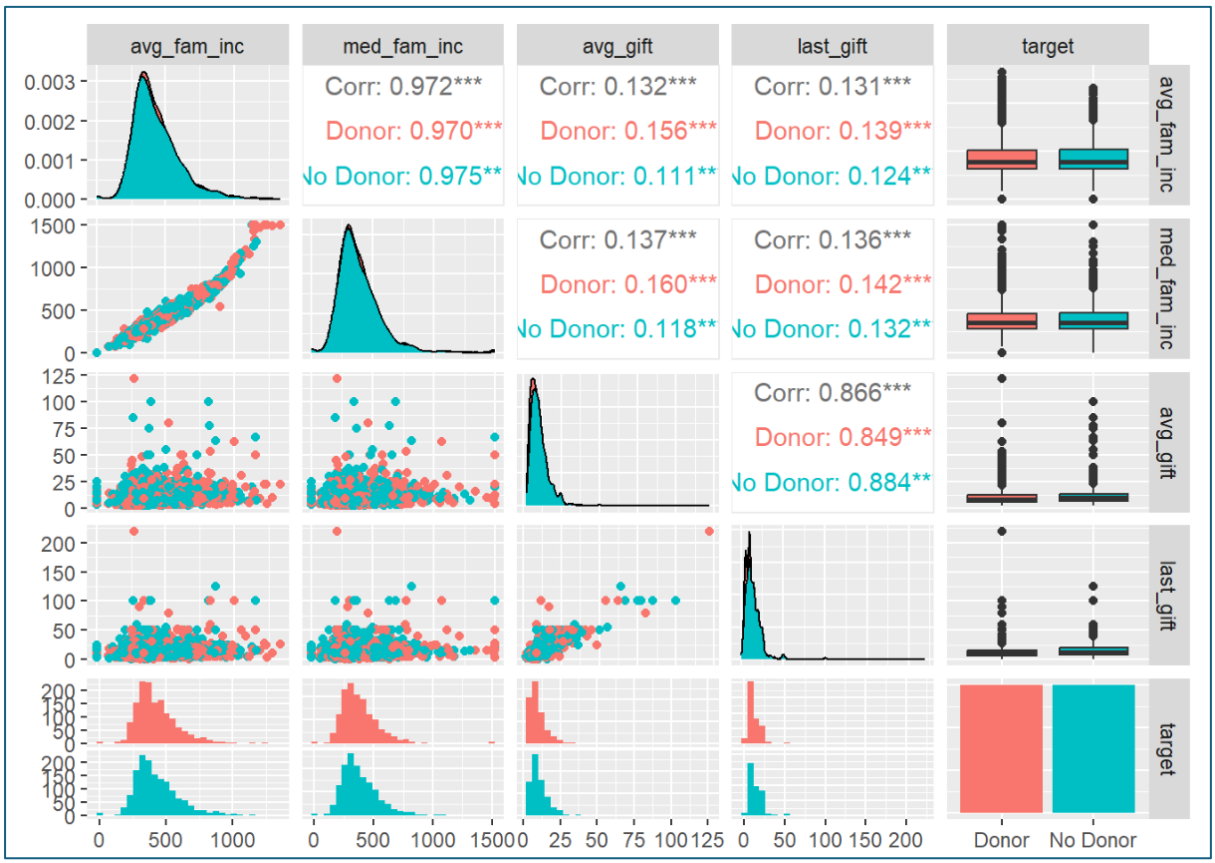
```
##       Variable1    Variable2 Correlation
## 1 avg_fam_inc med_fam_inc    0.9722713
## 2 med_fam_inc avg_fam_inc    0.9722713
## 3     avg_gift    last_gift    0.8664000
## 4    last_gift     avg_gift    0.8664000
```



Interpretation:

From the above correlation plot the highly correlated variables are avg_fam_inc, med_fam_inc, avg_gift, and last_gift.

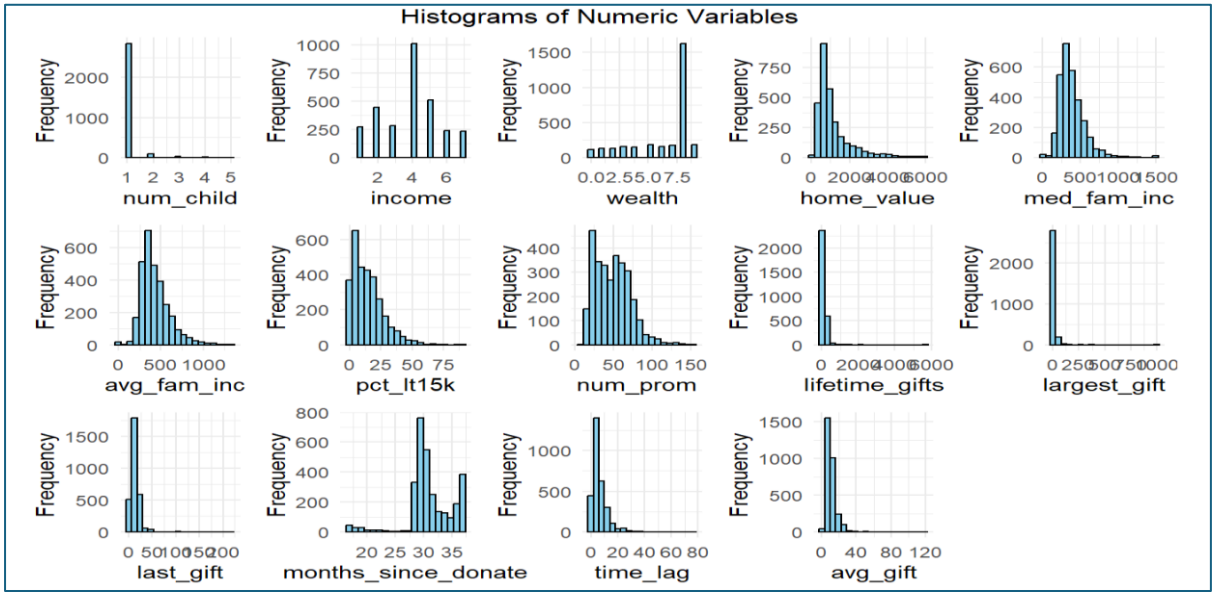
Pairwise plot for high correlated variables: -



Interpretation:

The variables avg_gift, and last_gift are positively skewed to Right skewness (tail to the right), I have noticed that their is high correlation between these variables along with the some correlation with the target variable(response variable).

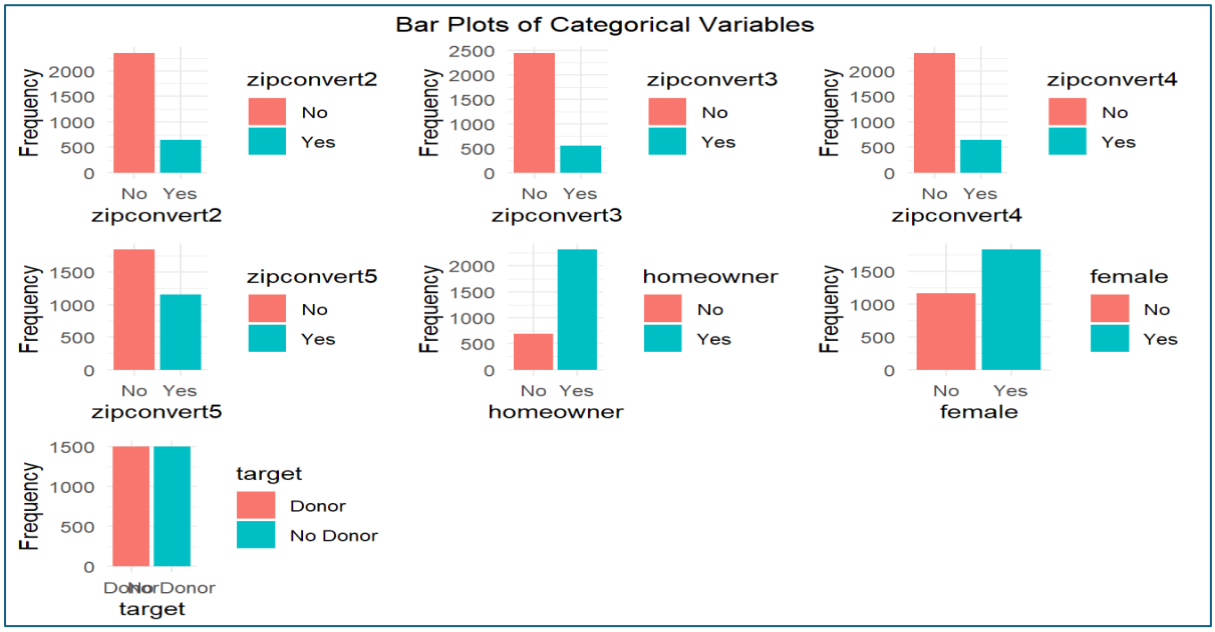
Histogram for numeric variables: -



Interpretation:

From the above graph it is understood that variable months_since_donate has high trend increase for the day between 25 to 30 and seen the fluctuations in the donation their after. Also, the avg_fam_inc for the potential donor's is above 500 to 700 for most of the donors.

categorical variable bar plots:-



Interpretation:

From the above plots its understood that variables “zipconvert3” has the highest number of non-donor’s nearly above 2000 who doesn’t belong to this zipconvert3 group. Variable “homeowner” implicates that more than 2000 donor’s have their own houses. “target” variable indicates that the data is equally distributed.

Model performance and Validation Results.

1. Logistic Regression Modelling:-

checking multicollinearity: -

```
##
## Call:
## glm(formula = target ~ . - zipconvert2, family = binomial, data = train_data_split)
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -1.736e+00  5.176e-01  -3.354 0.000797 ***
## zipconvert3Yes     9.853e-02  1.335e-01   0.738 0.460643
## zipconvert4Yes     3.228e-02  1.283e-01   0.252 0.801362
## zipconvert5Yes     6.280e-02  1.213e-01   0.518 0.604576
## homeownerYes      -1.483e-01  1.059e-01  -1.401 0.161240
## num_child          3.324e-01  1.279e-01   2.600 0.009327 **
## income            -5.440e-02  2.876e-02  -1.892 0.058537 .
## femaleYes         -2.616e-02  8.587e-02  -0.305 0.760655
## wealth            -2.034e-02  1.997e-02  -1.018 0.308481
## home_value        -9.720e-05  7.962e-05  -1.221 0.222156
## med_fam_inc       -1.220e-03  1.063e-03  -1.147 0.251266
## avg_fam_inc        1.668e-03  1.136e-03   1.469 0.141852
## pct_lt15k         -3.696e-03  4.940e-03  -0.748 0.454325
## num_prom          -4.062e-03  2.569e-03  -1.581 0.113822
## lifetime_gifts     2.916e-04  4.033e-04   0.723 0.469762
## largest_gift      -2.219e-03  3.406e-03  -0.652 0.514625
## last_gift          1.381e-02  8.668e-03   1.593 0.111200
## months_since_donate  5.363e-02  1.126e-02   4.762 1.91e-06 ***
## time_lag          -1.160e-03  7.743e-03  -0.150 0.880878
## avg_gift           5.650e-03  1.237e-02   0.457 0.647882
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3327.0  on 2399  degrees of freedom
## Residual deviance: 3253.1  on 2380  degrees of freedom
## AIC: 3293.1
##
## Number of Fisher Scoring iterations: 4
```

```
##        zipconvert3        zipconvert4        zipconvert5          homeowner
##           1.583803           1.613826           2.030926           1.149109
##          num_child             income             female             wealth
##           1.026845           1.318902           1.017133           1.540787
##         home_value       med_fam_inc       avg_fam_inc       pct_lt15k
##           3.292901          19.255141          21.116940           2.069671
##          num_prom     lifetime_gifts       largest_gift         last_gift
##           1.943870           2.096937           1.990150           3.550115
## months_since_donate          time_lag           avg_gift
##           1.153031           1.045598           3.853753
```

Interpretation:

I have removed “zipconvert2” as the VIF co-efficients are getting influenced by this variable due to high correlation.

VIF co-efficient values below 5 indicate low multicollinearity and are generally acceptable for any data analysis and conversely, VIF co-efficient values above 5 indicate moderate to high multicollinearity, suggesting that the variable would be removed from further analysis. From the above the analysis variables “med_fam_inc”, and “avg_fam_inc” have high VIF co-efficient and even from the exploratory data analysis also suggest that these variables are highly correlated(multicollinearity).

```
vif(m2.log_wholedata_stepwise_backward)
```

```
##          homeowner          num_child             income          last_gift
##           1.132511           1.009586           1.154725           1.063439
## months_since_donate
##           1.056160
```

Interpretation:

Removed the variables which has the least significant using Backward elimination of AIC value. Post stepwise selection we are left with 5 features(homeowner + num_child + income + last_gift + months_since_donate) we can use this for further prediction of the target(response variable).

Multicollinearity is satisfied here, as VIF for the left over features(homeowner + num_child + income + last_gift + months_since_donate) post step wise(backward) selection has values less than 5(default value for VIF).

Confusion Matrix:-

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Donor No Donor
##   Donor      117       145
##   No Donor   173       165
##
##                Accuracy : 0.47
##                  95% CI : (0.4295, 0.5108)
##     No Information Rate : 0.5167
##     P-Value [Acc > NIR] : 0.99
##
##                   Kappa : -0.0645
##
##  Mcnemar's Test P-Value : 0.13
##
##             Sensitivity : 0.4034
##             Specificity : 0.5323
##          Pos Pred Value : 0.4466
##          Neg Pred Value : 0.4882
##              Prevalence : 0.4833
##          Detection Rate : 0.1950
##    Detection Prevalence : 0.4367
##       Balanced Accuracy : 0.4679
##
##        'Positive' Class : Donor
##
```

```
cat("Accuracy (log_reg):", accuracy_log_reg, "\n")
```

```
## Accuracy (log_reg): 0.47
```

```
cat("Precision (log_reg):", precision_log_reg, "\n")
```

```
## Precision (log_reg): 0.4466
```

```
cat("Recall (log_reg):", recall_log_reg, "\n")
```

```
## Recall (log_reg): 0.4034
```

```
cat("F1 Score (log_reg):", f1_score_log_reg, "\n")
```

```
## F1 Score (log_reg): 0.4239
```

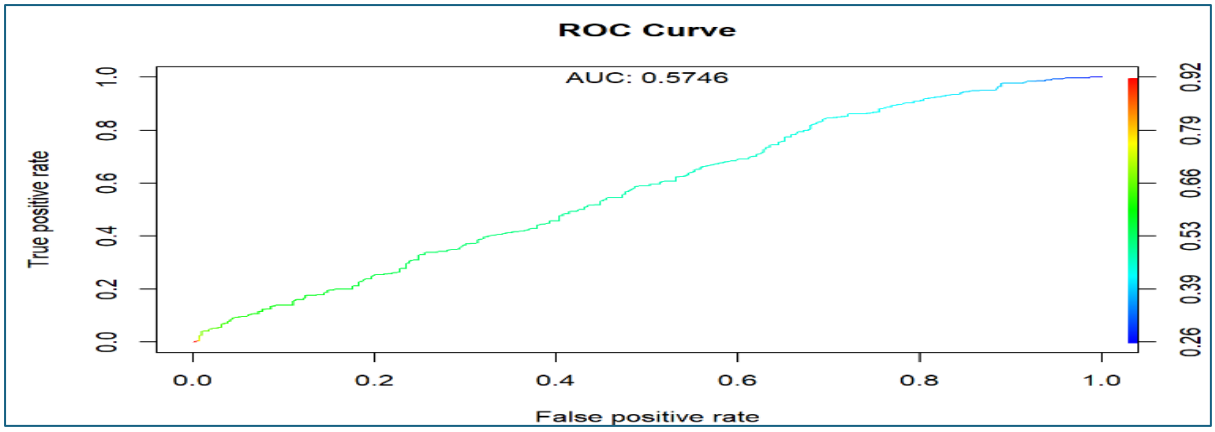
Interpretation:

From the above confusion matrix which shows that 117 instances were properly predicted as “Donor,” and 165 instances as “No Donor.” also, there were misclassifications: 173 cases of “No Donor” were wrongly projected as “Donor,” while 145 cases of “Donor” were wrongly forecasted as “No Donor.” The percentage of accurate predictions is used to determine the accuracy of the model.

From the summary:

Recall: 0.4034 suggests that the model correctly identifies about 40.34% of all actual positive samples.
Precision: 0.4466 that about 44.66% of the samples predicted as positive are actually positive.
F1 Score:0.4239 model has a average balance between precision and recall.

The model’s overall accuracy is 47%, indicating that it correctly classifies 47% of all instances, hence model is not best model for the case study and we will check other models like “Random Forest”, “K-Nearest Neighbor”, “Naive Bayes”.



Interpretation:

From the above ROC plot we can understand the performance and ability of the model in classification the instance (positive and negative), AUC of 0.5746 indicates that the model's predictions are, on average, 57.46% correct in terms of assigning higher probabilities to positive instances compared to negative instances. With 57.46% is considered the model has below averagely discriminating ability in this binary classification task, also its merely guessing the predictions. Hence, we can say that this model is not perfectly suitable for binary classification task and recommending further analysis with different model building.

2. Using Random Forest with variable importance:

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Donor No Donor
##   Donor       175       134
##   No Donor    115       176
##
##                Accuracy : 0.585
##                  95% CI : (0.5444, 0.6248)
##     No Information Rate : 0.5167
##     P-Value [Acc > NIR] : 0.0004536
##
##                   Kappa : 0.1708
##
##  Mcnemar's Test P-Value : 0.2539933
##
##             Sensitivity : 0.6034
##             Specificity : 0.5677
##          Pos Pred Value : 0.5663
##          Neg Pred Value : 0.6048
##              Prevalence : 0.4833
##          Detection Rate : 0.2917
##    Detection Prevalence : 0.5150
##       Balanced Accuracy : 0.5856
##
##        'Positive' Class : Donor
##
```

```
cat("Accuracy (_rf_tuned):", round(accuracy_rf_tuned,4), "\n")
```

```
## Accuracy (_rf_tuned): 0.585
```

```
cat("Precision (_rf_tuned):", precision_rf_tuned, "\n")
```

```
## Precision (_rf_tuned): 0.5663
```

```
cat("Recall (_rf_tuned):", recall_rf_tuned, "\n")
```

```
## Recall (_rf_tuned): 0.6034
```

```
cat("F1 Score (_rf_tuned):", f1_score_rf_tuned, "\n")
```

```
## F1 Score (_rf_tuned): 0.5843
```

Interpretation:

From the confusion matrix:

True Positives (TP): 175, meaning 175 instances were correctly predicted as "Donor." True Negatives (TN): 176, indicating 176 instances were correctly predicted as "No Donor." False Positives (FP): 134, representing instances wrongly predicted as "Donor" when they were actually

"No Donor." False Negatives (FN): 115, indicating instances wrongly predicted as "No Donor" when they were actually "Donor."

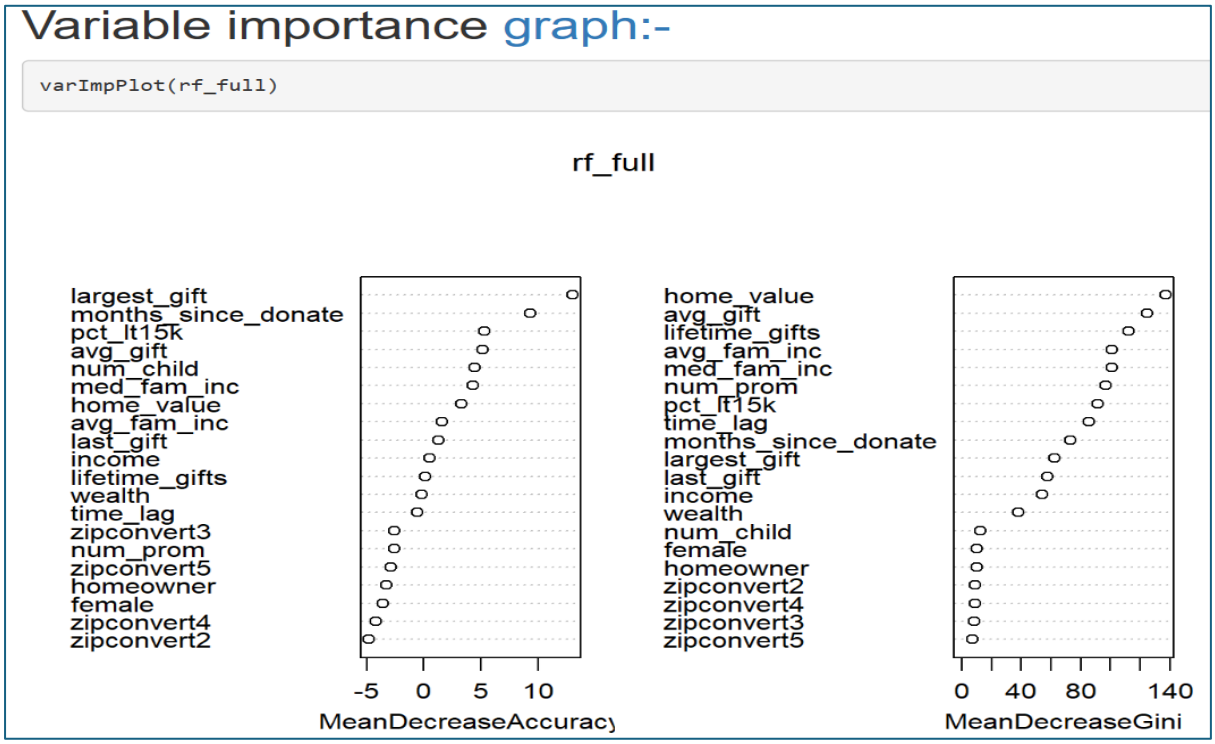
From the summary:

Recall: 0.6034 suggests that the model correctly identifies about 60.34% of all actual positive samples.
Precision: 0.5663 that about 56.63% of the samples predicted as positive are actually positive.
F1 Score:0.5843 model has a average balance between precision and recall.

The model's overall accuracy is 0.585, indicating that it correctly classifies 58.5% of all instances.

Variable importance graph:-

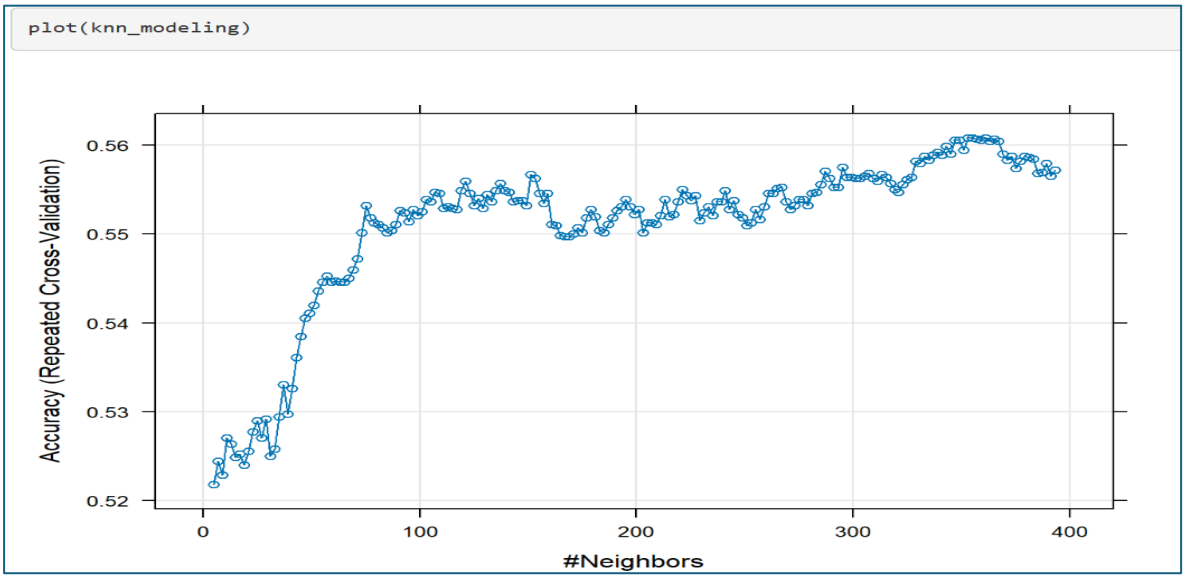
```
varImpPlot(rf_full)
```



Interpretation: From the above variable importance plot it is understood that variables at the top are most important for the random forest and being used in the model building.

3. KNN modelling:

```
plot(knn_modeling)
```



```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Donor No Donor
##   Donor      201      180
##   No Donor    89      130
##
##                Accuracy : 0.5517
##                  95% CI : (0.5109, 0.592)
##     No Information Rate : 0.5167
##     P-Value [Acc > NIR] : 0.04687
##
##                   Kappa : 0.1113
##
##  Mcnemar's Test P-Value : 4.079e-08
##
##             Sensitivity : 0.6931
##             Specificity : 0.4194
##          Pos Pred Value : 0.5276
##          Neg Pred Value : 0.5936
##              Prevalence : 0.4833
##          Detection Rate : 0.3350
##    Detection Prevalence : 0.6350
##       Balanced Accuracy : 0.5562
##
##        'Positive' Class : Donor
##
```

```
# Print the results
cat("Accuracy_knn_modeling:", accuracy_knn_modeling, "\n")
```

```
## Accuracy_knn_modeling: 0.5517
```

```
cat("Precision_knn_modeling:", precision_knn_modeling, "\n")
```

```
## Precision_knn_modeling: 0.5276
```

```
cat("Recall_knn_modeling:", recall_knn_modeling, "\n")
```

```
## Recall_knn_modeling: 0.6931
```

```
cat("F1 Score_knn_modeling:", f1_score_knn_modeling, "\n")
```

```
## F1 Score_knn_modeling: 0.5991
```

Interpretation:

The model correctly identified 201 instances of Donor and 130 instances of No Donor. However, it misclassified 89 instances as Donor when they were actually No Donor, and 180 instances as No Donor when they were actually Donor.

From the summary:

Recall: 0.6931 suggests that the model correctly identifies about 69.31% of all actual positive samples.
Precision: 0.5276 that about 52.76% of the samples predicted as positive are actually positive.
F1 Score:0.5991 model has a average balance between precision and recall.

The model's overall accuracy is 55.17%, indicating that it correctly classifies 55.17% of all instances.

4. Naive bayes Modelling:

```
## Naive Bayes
##
## 2400 samples
##    5 predictor
##    2 classes: 'Donor', 'No Donor'
##
## No pre-processing
## Resampling: Cross-Validated (15 fold, repeated 3 times)
## Summary of sample sizes: 2239, 2240, 2240, 2241, 2240, 2240, ...
## Resampling results across tuning parameters:
##
##   usekernel  Accuracy   Kappa
##   FALSE      0.5410850  0.07849578
##    TRUE      0.5274866  0.05016914
##
## Tuning parameter 'laplace' was held constant at a value of 0
## Tuning
##  parameter 'adjust' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were laplace = 0, usekernel = FALSE
##  and adjust = 1.
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction Donor No Donor
##   Donor       231       234
##   No Donor     59        76
##
##                  Accuracy : 0.5117
##                    95% CI : (0.4709, 0.5524)
##      No Information Rate : 0.5167
##      P-Value [Acc > NIR] : 0.6127
##
##                     Kappa : 0.0409
##
##  Mcnemar's Test P-Value : <2e-16
##
##              Sensitivity : 0.7966
##              Specificity : 0.2452
##           Pos Pred Value : 0.4968
##           Neg Pred Value : 0.5630
##               Prevalence : 0.4833
##           Detection Rate : 0.3850
##     Detection Prevalence : 0.7750
##        Balanced Accuracy : 0.5209
##
##         'Positive' Class : Donor
##
```

```
# Print the results
cat("Accuracy_nb_model_fit:", accuracy_nb_model_fit, "\n")
```

```
## Accuracy_nb_model_fit: 0.5117
```

```
cat("Precision_nb_model_fit:", precision_nb_model_fit, "\n")
```

```
## Precision_nb_model_fit: 0.4968
```

```
cat("Recall_nb_model_fit:", recall_nb_model_fit, "\n")
```

```
## Recall_nb_model_fit: 0.7966
```

```
cat("F1 Score_nb_model_fit:", f1_score_nb_model_fit, "\n")
```

```
## F1 Score_nb_model_fit: 0.612
```

Interpretation:

The model correctly identified 231 instances of Donor and 76 instances of No Donor. However, it misclassified 59 instances as Donor when they were actually No Donor, and 234 instances as No Donor when they were actually Donor.

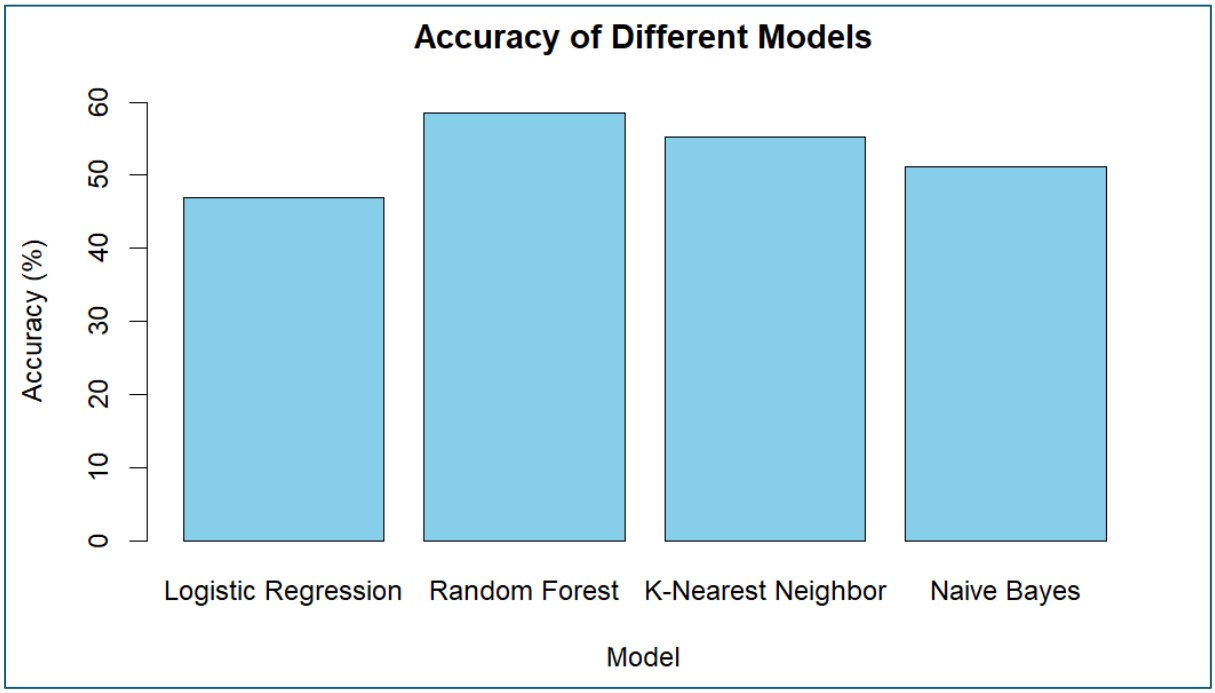
From the summary:

Recall: 0.7966 suggests that the model correctly identifies about 79.66% of all actual positive samples Precision: 0.4968 that about 49.68% of the samples predicted as positive are actually positive. F1 Score:0.612 model has a average balance between precision and recall.

The model's overall accuracy is 0.5117%, indicating that it correctly classifies 51.17% of all instances.

Model Comparison to find the best accuracy:

```
##          Logistic Regression Model Random Forest Model K-Nearest Neighbor Model
## Accuracy                      0.47              0.585                    0.5517
##          Naive Bayes Model
## Accuracy            0.5117
```



Interpretation:

From the above model comparison, I understand the accuracy is highest for “Random Forest” Model hence recommending this as a predictive modelling for this task which can be used to develop a classification model that can effectively capture donors so that the expected net profit is maximized.

Testing on Future Fundraising Test File: -

````
## Testing on Future_fundraising test file:-

```{r}
set.seed(12345)

formula_rf <- as.formula(paste("target ~ homeowner + num_child + income + last_gift + months_since_donate"))

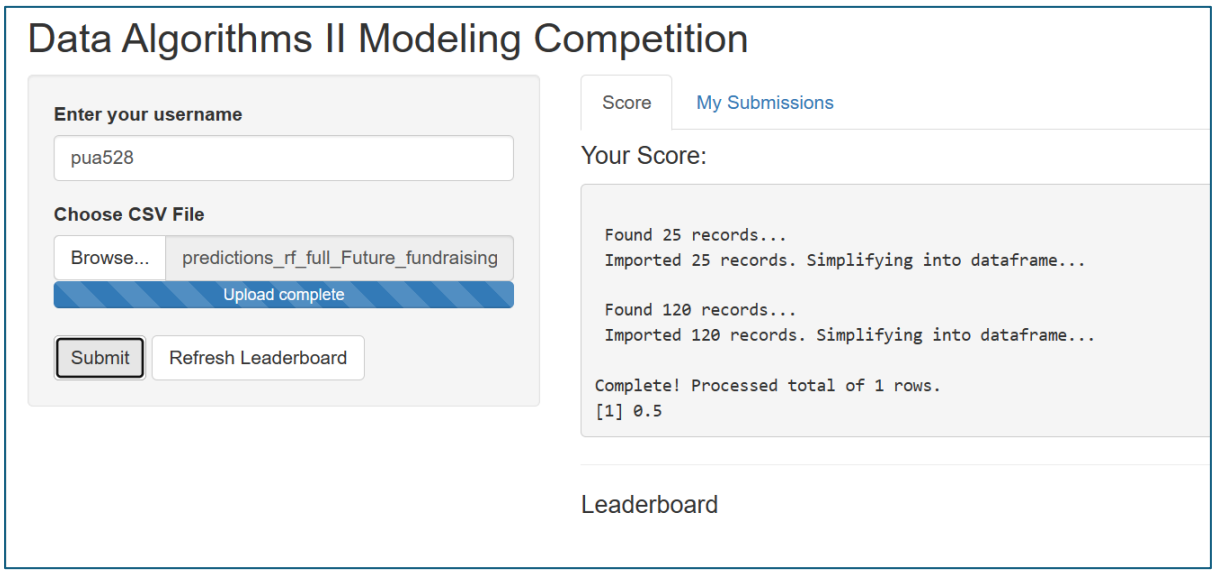
rf_full_future_fund = randomForest(formula_rf, data = train_data_split, mtry = 5, ntree = 1000,
    importance = T)


```

```{r}
pred_probs_rf_full_modeling_Future_fundraising = predict(rf_full_future_fund, Future_fundraising)
```

```{r}
# Write header to CSV file
header <- "value"
write(header, file = "predictions_rf_full_Future_fundraising_final.csv")

# Append data to CSV file
write.table(pred_probs_rf_full_modeling_Future_fundraising, file =
"predictions_rf_full_Future_fundraising_final.csv", col.names = FALSE, row.names = FALSE, append = TRUE, sep
= ",")
````


````

Cut-Off Analysis:

Cut-off analysis:

```
pred_probs_rf_full_modeling_Future_fundraising <- predict(rf_full_future_fund, Future_fundraising, type = "prob")[, "Donor"]


mailing_cost <- 0.68
average_donation_value <- 13
donor_response_rate <- 0.051
```

```
cutoff_values <- seq(0, 0.8, by = 0.01)

calculate_targeted_counts <- function(cutoff) {
  sum(pred_probs_rf_full_modeling_Future_fundraising > cutoff)
}

targeted_counts <- sapply(cutoff_values, calculate_targeted_counts)
expected_donors <- targeted_counts * donor_response_rate
total_costs <- targeted_counts * mailing_cost
total_benefits <- expected_donors * average_donation_value
net_benefits <- total_benefits - total_costs

cutoff_analysis_df <- data.frame(
  cutoff = cutoff_values,
  total_cost = total_costs,
  total_benefit = total_benefits,
  net = net_benefits
)


optimal_cutoff <- cutoff_analysis_df[cutoff_analysis_df$net == max(cutoff_analysis_df$net), "cutoff"]
optimal_cutoff <- optimal_cutoff[1]
cat("Optimal Cutoff: ", optimal_cutoff,"\n")
```

```
## Optimal Cutoff:  0.79
```

Recommendations:

```
probability_threshold <-optimal_cutoff

num_targeted <- sum(pred_probs_rf_full_modeling_Future_fundraising > probability_threshold)

percentage_to_use <- (num_targeted / length(pred_probs_rf_full_modeling_Future_fundraising)) * 100

cat("Percentage of data to use for the mailing campaign: ", percentage_to_use, "%\n")
```

```
## Percentage of data to use for the mailing campaign:  20 %
```

```
total_costs <- num_targeted * mailing_cost
expected_donors <- num_targeted * donor_response_rate
total_benefits <- expected_donors * average_donation_value
net_benefits <- total_benefits - total_costs

cat("Total Costs: $", total_costs, "\n")
```

```
## Total Costs: $ 16.32
```

```
cat("Expected Donors: ", expected_donors, "\n")
```

```
## Expected Donors:  1.224
```

```
cat("Total Benefits: $", total_benefits, "\n")
```

```
## Total Benefits: $ 15.912
```

```
cat("Net Benefits: $", net_benefits, "\n")
```

```
## Net Benefits: $ -0.408
```

Interpretation:

From the above analysis it is understood that 20% of data to use for the mailing campaign.