# IDS 572
# Assignment – 3
# Target Marketing – Fundraising (Part 2)

Bala Rohit Yeruva

Fall 2016

# 1. Response Modelling – Comparative Evaluation

As a continuation to Assignment-2, where we developed classification models to predict the donors, we have built two more classification models; Support Vector Model (SVM) and k-Nearest Neighbours (k-NN) and analysed the performance of the best models from all the techniques.

Support Vector Model (SVM):

As SVM handles only numeric attributes, all polynomial and nominal variables were transformed into numeric variables and all the numerical variables were checked if they can be normalized any further.

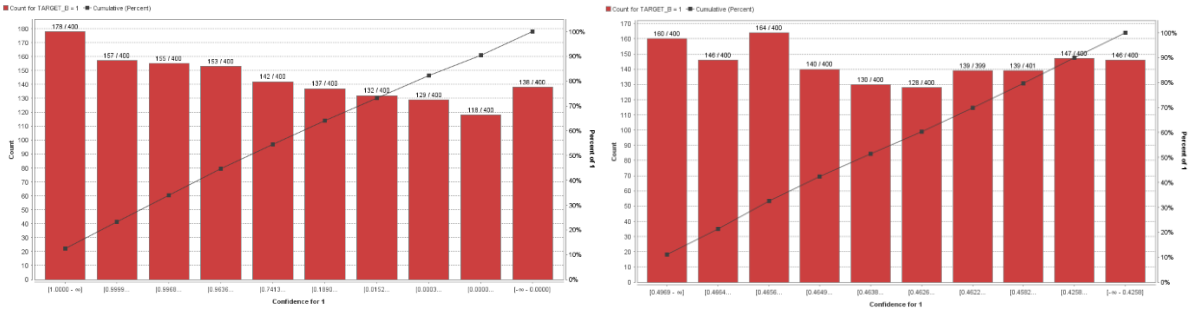
While figuring out the best possible parameters for our model, we found out that for any selected set of attributes, SVM performs relatively well on the training data than that on the validation data. The results on the validation dataset are such that the model either predicts 90% of the data as donors or non-donors. When a polynomial with degree greater than 2 or a Gaussian is used as the kernel type, it classifies the complete data as non-donors. Evidently, Radial kernel type outperforms almost all kernel types. After trying several combinations of kernel gamma and cost estimation (C, Lpos, Lneg) values, we were able to achieve better results when gamma was 0.005 and the cost estimation values were 1, 0.799 and 10 respectively. After finalizing the parameters, we have built models which were executed on different subsets of the variables. Table 1.1 shows the performance of these models.

| ***Subset*** | ***Accuracy on Training Data*** | ***Accuracy on Validation Data*** | ***Precision(1) on Training Data*** | ***Precision(1) on Validation Data*** | ***Recall(1) on Training Data*** | ***Recall(1) on Validation Data*** |
|---|---|---|---|---|---|---|
| All PCAs | 99.55% | 62.10% | 98.71% | 40.00% | 100% | 10.70% |
| W/O Neighborhood PCA | 99.97% | 63.22% | 99.90% | 38.41% | 100% | 3.68% |
| W/O Mail Response PCA | 99.55% | 62.18% | 98.75% | 40.36% | 100% | 10.77% |
| W/O Donor Interests PCA | 99.57% | 62.02% | 98.75% | 40.10% | 100% | 10.70% |
| No PCAs | 99.97% | 63.22% | 99.90% | 38.24% | 100% | 3.61% |

*Table 1.1*

From the above table, we could observe that removing second and the third PCS doesn't have much impact on the model accuracy. Further, we could see that the model with all the PCAs included has the better accuracy and relatively reasonable recall value for Class 1. Therefore, the first model above would be our best SVM model.

The following are the lift charts obtained when the above better model is run on the training and the validation dataset respectively.



k-Nearest Neighbours (k-NN):

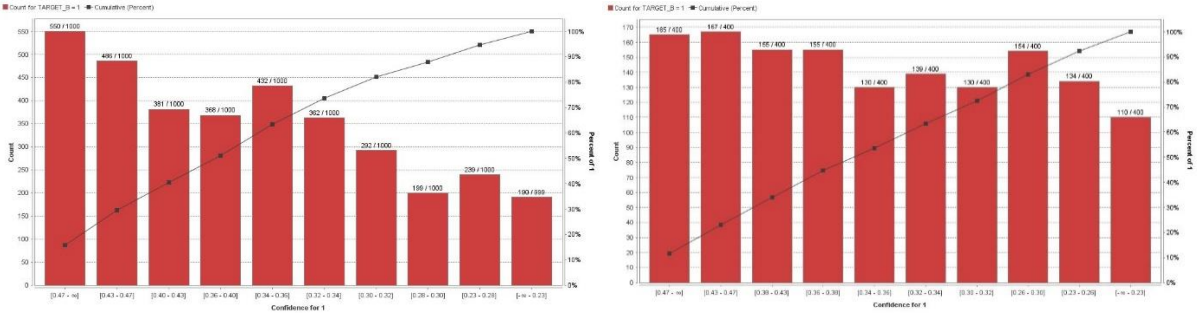
k-Nearest Neighbours is a nonparametric method used for detecting patterns. This method can be used for both classification and regression. In k-NN Classification, the dependent variable is classified based on the majority vote of its neighbours, with the response variable being classified to the most common class among the k nearest neighbours. While building this model, the parameters that best worked for was when we used k = 47. We then built several models which were executed on different subsets of the variables. Table 1.2 shows the performance of these models.

| ***Subset*** | ***Accuracy on Training Data*** | ***Accuracy on Validation Data*** | ***Precision(1) on Training Data*** | ***Precision(1) on Validation Data*** | ***Recall(1) on Training Data*** | ***Recall(1) on Validation Data*** |
|---|---|---|---|---|---|---|
| All PCAs | 66.38 % | 63.38% | 57.34% | 43.23% | 8.16% | 5.77% |
| W/O Neighborhood PCA | 74.93% | 60.20% | 73.02% | 39.33% | 37.57% | 19.60% |
| W/O Mail Response PCA | 74.28% | 60.12% | 73.56% | 39.71% | 39.17% | 20.92% |
| W/O Donor Interests PCA | 74.23% | 60.08% | 73.36% | 39.58% | 39.17% | 20.85% |
| No PCAs | 74.93% | 60.20% | 78.02% | 39.33% | 37.57% | 19.60% |

*Table 1.2*

From the above table, we could observe that variable selection does have a significant impact on the model accuracy. Further, we could see that the model without the Mail Response PCA has a better accuracy and a reasonable recall value for Class 1 when compared with the other models. Therefore, the third model above would be our best k-NN model.

The following are the lift charts obtained when the above better model is run on the training and the validation dataset respectively.



## Comparative Evaluation of all the best models from each technique:

After finalizing on the best models for SVM and k-NN, we compared the performance of these models against all the other best models developed using other techniques in Assignment-2. We could still observe that the model built using Boosted Trees was performing better than the models from SVM and k-NN.

| ***Model*** | ***Accuracy on Training Data*** | ***Accuracy on Validation Data*** | ***Recall(1) on Training Data*** | ***Recall(1) on Validation Data*** |
|---|---|---|---|---|
| Naïve Bayes | 79.48% | 52.00% | 86.02% | 58.51% |
| Decision Trees | 69.04% | 62.65% | 26.75% | 20.36% |
| Random Forest | 98.92% | 61.27% | 97.09% | 17.16% |
| Boosted Trees | 68.66% | **60.30%** | 86.36% | **73.25%** |
| Logistic Regression | 52.43% | 50.12% | 79.42% | 76.44% |
| SVM | 99.55% | 62.10% | 100% | 10.70% |
| k-NN | 74.28% | 60.12% | 39.17% | 20.92% |

*Table 1.3*

Therefore, we concluded to stick with **Gradient Boosted Trees** as our best model from all the techniques.

## 2.1. Calculating Net Profit – Maximizing Profit

According to the recent mailing records, the average donation from a donor is $13.00 and the cost of sending a mail is $0.68. Therefore, when a donor is predicted accurately, we gain a profit of $12.32 ($13.00 – $0.68) and when a donor is incorrectly predicted, we incur a loss of $0.68. However, these values are observed when the distribution of the classes were 5.1% donors and 94.9% non-donors.

As the original data was sampled into 35% donors (over sampling) and 65% non-donors (under sampling), we need to calculate the appropriate weighted profits and weighted costs for the sampled data.

$$\text{Weighted Profit} = \text{Profit per donation} * \frac{(\%\ \text{of Actual Responders})}{(\%\ \text{of Responders in sampled data})}$$

⇒ Weighted Profit = 12.32 * (.051)/ (3499/9999) = **1.7955**

$$\text{Weighted Cost} = \text{Cost of solicitation} * \frac{(\%\ \text{of Actual Non}-\text{Responders})}{(\%\ \text{of Non}-\text{Responders in sampled data})}$$

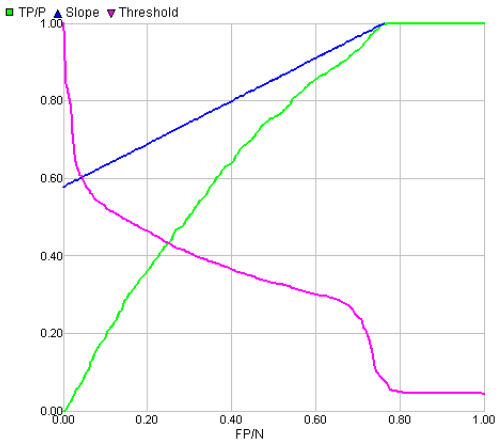
⇒ Weighted Cost = 0.68 * (.949) / (6500/9999) = **0.9927**

We used these weighted profit and costs to calculate net profit for each of the best models from different techniques. We primarily focused on maximizing the profits and hence, used the 'Find Threshold' operator on the validation data alone, to find the corresponding threshold value which gives us the maximum profit from a given model. Table 2.1 shows the performance and the maximum profits obtained from each model.

| ***Model*** | ***Accuracy on Training Data*** | ***Accuracy on Validation Data*** | ***Recall(1) on Training Data*** | ***Recall(1) on Validation Data*** | ***MaxProfit on Validation Data*** | ***Threshold on Validation Data*** |
|---|---|---|---|---|---|---|
| Naïve Bayes | 90.03% | 35.95% | 87.38% | 99.86% | 227.749 | 0.258 |
| Decision Trees | 68.93% | 40.00% | 24.03% | 93.26% | 327.762 | 0.23 |
| Random Forest | 99.20% | 37.75% | 97.72% | 98.33% | 250.143 | 0.259 |
| Boosted Trees | 68.66% | 51.50% | 86.36% | 99.58% | 702.676 | 0.342 |
| Logistic Regression | 52.43% | 35.98% | 79.42% | 100% | 428.947 | 0.36 |
| SVM | 99.55% | 36.00% | 100% | 100% | 117.42 | 0.427 |
| k-NN | 66.38% | 36.18% | 8.16% | 99.79% | 214.342 | 0.361 |

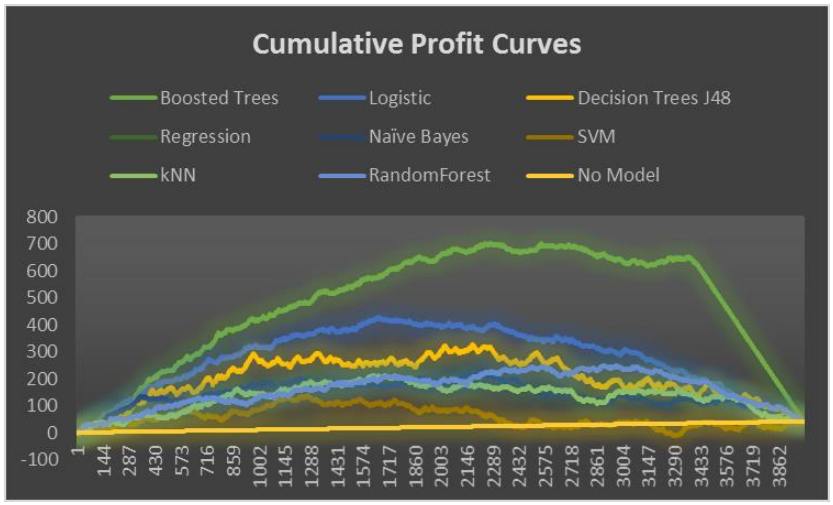
*Table 2.1*

From the above table, we could observe that all the models have less accuracy and high Recall value. This is because we concentrated on finding the maximum profit using the optimal threshold value rather than concentrating on the model accuracy. Further, the model built using Boosted Trees seems to give the maximum profit (threshold – 0.342) when compared to the other models. Below is the ROC curve obtained from this Boosted tree model.




Profit Curve:

From the table 2.1, it is evident that Booted trees give the maximum profit. However, it is not advisable to conclude on the best profitable model by just looking at the numbers. It could be possible that the maximum profit was achieved at a narrow peak, which indicates that the model might not perform as better as it did on an unseen data. Hence, it is always recommended to plot the cumulative profit curve for all the models in order to conclude the most profitable as well as a robust model. The below graph shows the cumulative profit curves for all the best models.



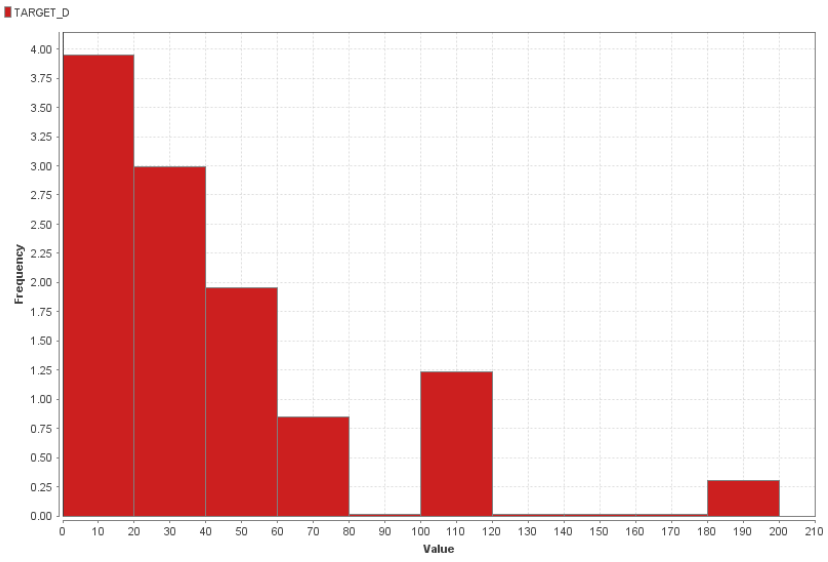
From the above graph, we can conclude that Boosted Trees has the maximum cumulative profit with a reasonable profit curve shape without and narrow peaks. Therefore, in all respects, we can conclude that the model built using Boosted Trees is our best model.

## 2.2. Regression Model

The next step in our analysis is to build a regression model which is used to predict the dollar amount donated by a donor. As the donation amount is only applicable to the donors, we have filtered the data such that it contains only donors.

Preparing/Selecting variables for Regression Model:

In selecting the variables for the regression model, we first analysed the distribution of our dependent variable; Donation Amount (Target_D). Since there were huge number of small donations when compared to large donations, we have used logarithmic scale. For the original distribution please refer to Appendix A.




From the distribution, we could see that there seemed to be a few extreme outliers (around $200). After analysing these data points, we could observe that the donors who donated these heavy donations didn't always do so. Their average donations were less than or equal to $20. This could indicate that while entering the data, it could be possible that $200 was entered instead of $20 and hence we have converted these values to $20.

We then visualized the correlation between the dependent variable and each of the independent variable using scatter plots. Out of all the independent variables, we could observe that the variables: RAMNT_3 – RAMNT_24, had a strong correlation with the donation amount. Variables such as LASTGIFT & AVGGIFT have a moderate correlation and AGE & AGE907 variables have a weak correlation with the donation amount. All remaining variables appear to have no correlation with our dependent variable. Below are the scatter plots between variables with strong, moderate and no correlation with the donation amount (TARGET_D).

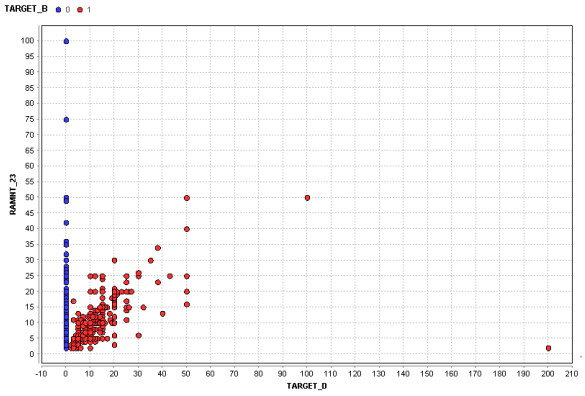



Fig.1. Scatter Plot between Target_D and RAMNT_23 (Strong Correlation)

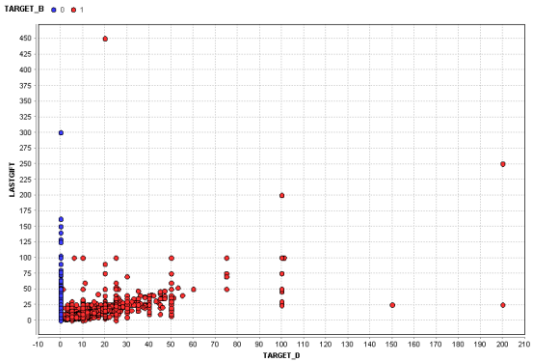



Fig.2. Scatter Plot between Target_D and LASTGIFT (Moderate Correlation)

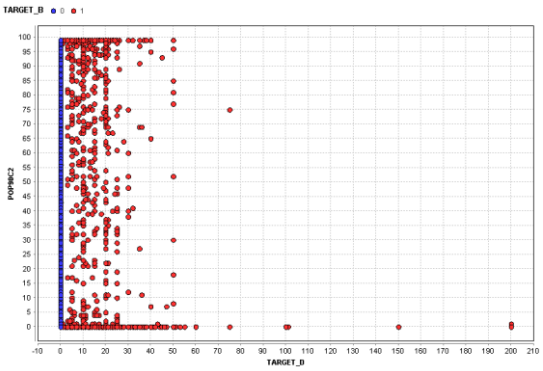



Fig.3. Scatter Plot between Target_D and POP90C1 (No Correlation)

After analysing all the scatter plots, we have decided to select only those variables which have at least weak correlation with the dependent variable. However, before building the model using these variables, we need to make sure that out of these selected models, there are no co-linear variables. Hence, we have calculated the correlation matrix for the selected independent variables and observed that AVGGIFT and LASTGIFT have a high correlation – 0.852.

| Attributes | LASTGIFT | AVGGIFT | Dollars |
|---|---|---|---|
| LASTGIFT | 1 | 0.852 | 0.612 |
| AVGGIFT | 0.852 | 1 | 0.467 |
| Dollars | 0.612 | 0.467 | 1 |

We could also see that between these two variables, LASTGIFT has a higher correlation with our dependent variable. Therefore, we have removed AVGGIFT from the selected independent variables which will be used for building our regression models.

After finalizing on the selected subset of variables, we have built the regression models using Simple Linear Regression and k-Nearest Neighbour and then based on their performance, we concluded our best regression model.

Simple Linear Regression:

The first regression model that we have built is a simple linear regression model. The parameters that gave us the best results was when using 'M5 prime' as the feature selection, eliminating any co-linear features among the subset of variables and minimum tolerance equal to 0.05. For validating our model, instead of using split validation, we have used K-fold cross validation which is a better cross validation technique for a regression model. The below table shows our model performance against the variation of K number of folds.

| GLM | K = 10 (K-fold) | K = 100 (K-fold) | K = 1000 (K-fold) |
|---|---|---|---|
| TRA | **RMS**: 9.191 +/- 0.320 | **RMS**: 9.241 +/- 0.120 | **RMS**: 9.247 +/- 0.038 |
| | **R**: 0.646 +/- 0.029 | **R**: 0.642 +/- 0.010 | **R**: 0.642 +/- 0.003 |
| | **Rsqr**: 0.418 +/- 0.039 | **Rsqr**: 0.413 +/- 0.013 | **Rsqr**: 0.412 +/- 0.004 |
| VAL | **RMS**: 9.892 +/- 3.201 | **RMS**: 8.468 +/- 6.068 | **RMS**: 6.659 +/- 7.754 |
| | **R**: 0.646 +/- 0.142 | **R**: 0.697 +/- 0.176 | **R**: 0.705 +/- 0.337 |
| | **Rsqr**: 0.438 +/- 0.174 | **Rsqr**: 0.516 +/- 0.218 | **Rsqr**: 0.611 +/- 0.360 |

*Table 2.2*

From the table above, we could observe that when the number of folds is equal to 1000, the value of R square for the validation data is the highest and it is equal to 0.611 which implies that about 61% of the variance in the dependent variable is explained by the model or the independent variables. Therefore, this model would be our best Simple Linear Regression Model.

K-Nearest Neighbors:

The second regression model that we have built was using k-Nearest Neighbor method. This method can be used for both classification and regression. In k-NN regression, the value of the dependent variable is the average value of the k nearest neighbours. The below table shows our model performance with the variation of k (in K-NN).

| k-NN | k = 10 | k = 100 | k = 200 | k = 300 |
|---|---|---|---|---|
| TRA | **RMS**: 8.526 +/- 0.195 | **RMS**: 10.489 +/- 0.182 | **RMS**: 11.062 +/- 0.181 | **RMS**: 11.343 +/- 0.182 |
| | **R**: 0.736 +/- 0.009 | **R**: 0.676 +/- 0.009 | **R**: 0.676 +/- 0.009 | **R**: 0.669 +/- 0.010 |
| | **Rsqr**: 0.542 +/- 0.014 | **Rsqr**: 0.457 +/- 0.013 | **Rsqr**: 0.456 +/- 0.013 | **Rsqr**: 0.448 +/- 0.013 |
| VAL | **RMS**: 9.458 +/- 1.621 | **RMS**: 10.485 +/- 1.555 | **RMS**: 11.012 +/- 1.540 | **RMS**: 11.274 +/- 1.554 |
| | **R**: 0.638 +/- 0.070 | **R**: 0.667 +/- 0.063 | **R**: 0.672 +/- 0.067 | **R**: 0.669 +/- 0.069 |
| | **Rsqr**: 0.411 +/- 0.087 | **Rsqr**: 0.449 +/- 0.082 | **Rsqr**: 0.457 +/- 0.087 | **Rsqr**: 0.452 +/- 0.091 |

*Table 2.3*

From the above table, we could see that when k = 200, the model has the highest R square value on the validation data. After finalizing this k value from the above table, even here, we have used the K-fold cross validation instead of split validation.

| k-NN (k = 200) | K = 10 (K-fold) | K = 100 (K-fold) | K = 1000 (K-fold) |
|---|---|---|---|
| TRA | **RMS**: 8.526 +/- 0.195 | **RMS**: 8.508 +/- 0.064 | **RMS**: 8.506 +/- 0.021 |
| | **R**: 0.736 +/- 0.009 | **R**: 0.737 +/- 0.003 | **R**: 0.737 +/- 0.001 |
| | **Rsqr**: 0.542 +/- 0.014 | **Rsqr**: 0.543 +/- 0.004 | **Rsqr**: 0.543 +/- 0.001 |
| VAL | **RMS**: 9.458 +/- 1.621 | **RMS**: 8.520 +/- 4.328 | **RMS**: 6.500 +/- 7.009 |
| | **R**: 0.638 +/- 0.070 | **R**: 0.685 +/- 0.137 | **R**: 0.714 +/- 0.331 |
| | **Rsqr**: 0.411 +/- 0.087 | **Rsqr**: 0.488 +/- 0.176 | **Rsqr**: 0.619 +/- 0.355 |

*Table 2.4*

From the above table, we could observe that our R square value has improved significantly increased when we use K = 1000 (number of folds). We could thus conclude that the model with k = 200 (in k-NN) and K = 1000 (K-fold) is our best k-NN regression model.

Based on the performance (R-square) from the two models, we could observe that both the models' performance was appropriately equal. However, since K-NN is a memory intensive model which requires large amounts of memory to analyse large datasets, it is not usually recommended for large dataset as in our case. Therefore, we have concluded that Simple Linear Regression would be our best regression model among the two models.

Combining the Response Model and the Regression Model:

After concluding the respective best models for the regression and response model, we need to estimate a way to combine the two models in order to predict the future donors and their donation amounts. One way to combine these two models is to multiply the predicted dollar amount from the regression model with the confidence score for the donors which is calculated by the response model. This product would give us an accurate estimation of the donation amount with respect to the probability that that particular person will or may donate.

After combining the information from both the models as described above, we then need to find a threshold to identify the target donors. For this we can follow either of the below approaches:

- Use the threshold value calculated by the response model while calculating the maximum cumulative profit and multiply only those data points for which this model predicted them to be the donors.
- Multiply all the data points and then target only those customers whose predicted donor amount is greater than $0.68. We need to ensure this because if the profit from a donation is less than $0.68 which is the cost of one solicitation, it results in a loss rather than a profit.

Hence, we can follow either one of the above approaches or whichever results in more overall predicted donation amounts or profit.

# 3. Testing

From all the above best models, we have chosen the model built using the Boosted Trees in Section 2.1 to predict/target the donors from the future dataset. The threshold that we used to identify the donors is same as that of the best Boosted Tree model which is 0.342. The identified targets can be found in the excel: "Targeted Donors".

# Appendix

Appendix A