IDS 572: Assignment 3

Target Marketing - Fundraising (Part 2)

Archan Patel - 661105271 - apate381@uic.edu

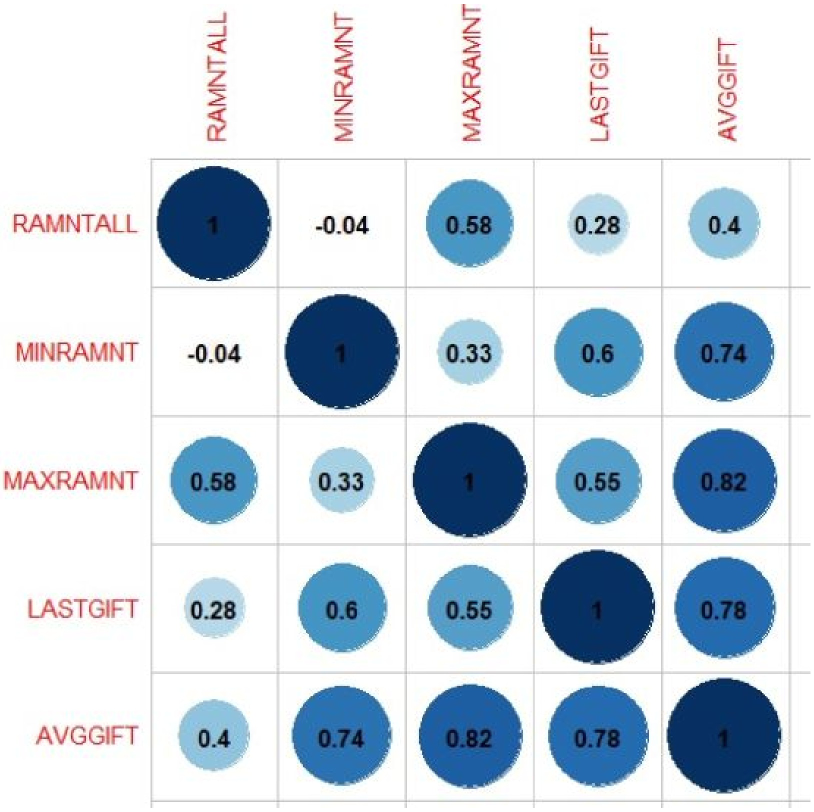
Jasbir Singh - 651837003 - jasingh3@uic.edu

Jeetyog Rangnekar - 656052696 - jrangn2@uic.edu

Assignment Question 1

Step 1

- In the previous assignment, we had selected a number of variables for the model to predict TARGET_B. For that phase, we had dropped the variables related to the donation amount. Now, for this task, we have to consider amount variables as well.
- So, variables RAMNT_3, RAMNT_4,...RAMNT_24, RAMNTALL, MINRAMNT, MAXRAMNT, AVGGIFT can be considered now. However, RAMNT_3 & RAMNT_4,... are individual donation amount for different promotion type. Since, individually they do not provide sufficient information we can combine them to get the total donation amount.
    - But since, RAMNTALL provides the same information, we can drop RAMNT_3, RAMNT_4,.. Variables.
    - Also, MINRAMNT, MAXRAMNT are highly correlated with AVGGIFT. Since, AVGGIFT provides average donation amount of the donor, it is much more significant variable. We can drop MINRAMNT & MAXRAMNT.



	RAMNTALL	MINRAMNT	MAXRAMNT	LASTGIFT	AVGGIFT
RAMNTALL	1	-0.04	0.58	0.28	0.4
MINRAMNT	-0.04	1	0.33	0.6	0.74
MAXRAMNT	0.58	0.33	1	0.55	0.82
LASTGIFT	0.28	0.6	0.55	1	0.78
AVGGIFT	0.4	0.74	0.82	0.78	1

- So, now we have AVGGIFT & RAMNTALL left. But by looking at the data, we can see that AVGGIFT * NGIFTALL = RAMNTALL. So basically, they provide the same information.
- Also, RAMNTALL is highly correlated with other selected variables while AVGGIFT is not. So it makes sense to drop RAMNTALL & keep AVGGIFT.

Step 2

- The next step is to develop Support Vector Machine model. Following is the summary of performance of different SVMs with different parameter values.

Kernel Type	**Kernel Gamma/Degree *(as applicable)***	**C**	**Maximum Iterations**	**Recall Training**	**Recall Testing**
Radial	0.3	1.0	9000	100%	0%
Radial	0.75	0.5	10000	0%	0%
Polynomial	2.0	1.0	9000	100%	13.22%
Polynomial	2.0	100	9000	100%	13.22%
Dot	NA	2.0	9000	65.24%	21.69%
Dot	NA	0.5	9000	67.59%	23.57%

- As we can see from the above table, Support Vector Machine with Kernel Type: Dot with C=0.5 gives the best result with Recall 23.57% on testing dataset.

Assignment Question 2.1

- Since the data on which we built our model is weighted to 12682:3429 from the original dataset of 94.9: 5.1 ratio, we have adjusted the calculation to undo the effect of weighted sampling.

  Weighted Profit = $13-$0.68 = $12.32

Weighted Cost = ($0.68)

Adjusted-Profit = ($12.32 * 0.051) / (3429/16111) =$2.9526

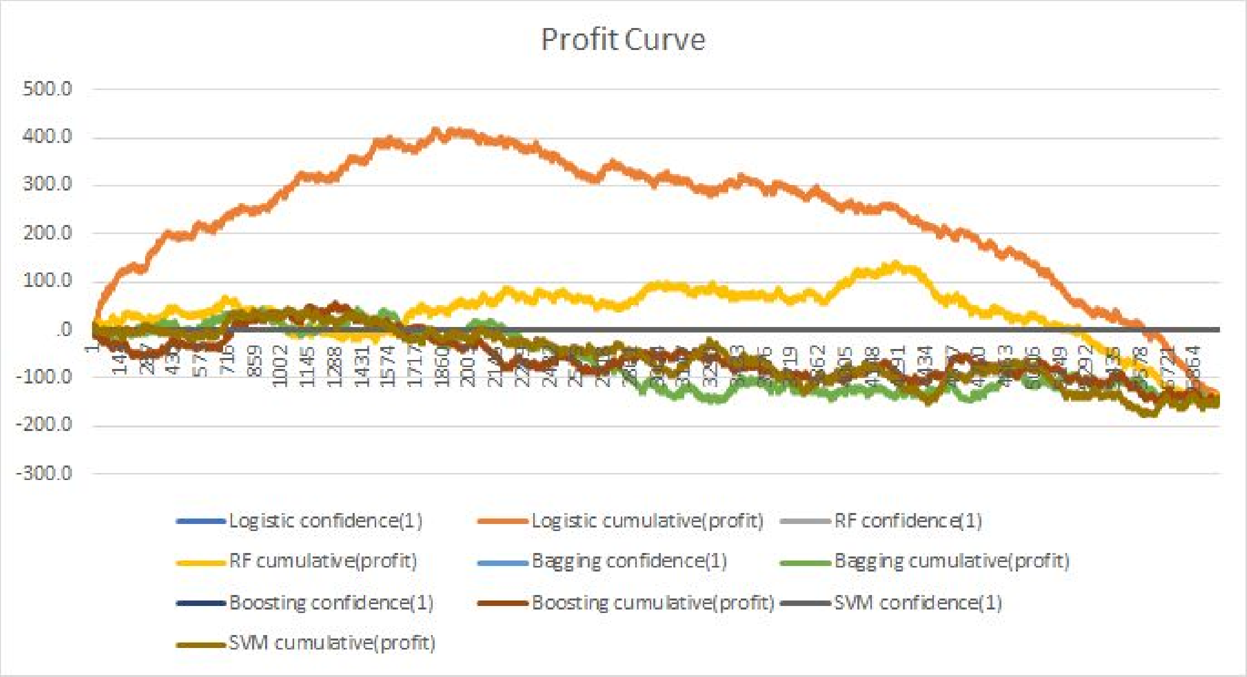
Adjusted-Cost = ($0.68 * 0.949) / (12682/16111) = $0.8198

- So, Formula for Net Profit would be: If TARGET_B = 1, then profit is $2.9526, otherwise $(-0.8198)
- Here is the summary of the profit obtained from different models.

Model	**Parameters**	**Recall Training**	**Recall Testing**	**Training Profit**	**Testing Profit**
Logistic Regression	Lasso, Lambda=1.4E-6	97.52	92.73	1829.108	422.984
Logistic Regression	Lasso, Lambda=2.3E-7	96.93	90.83	1965.276	444.838
Bagging	Information Gain Tree Depth = 14	93.10	33.83	5041.967	44.525
Bagging	Gini Index Tree Depth = 18	73.67	38.36	3891.443	85.495
Random Forest	Information Gain Tree Depth =16 No. of Trees = 80	99.59	32.56	6430.783	262.331
Random Forest	Gini Index Tree Depth = 14 No. of Trees = 100	99.86	29.29	6470.639	233.637
Boosting	Information Gain Tree Depth = 20	100.00	19.29	5604.035	57.449
SVM	Dot C=0.5 Iterations = 9000	98.15	80.08	2956.793	97.098
SVM	Polynomial C=1.0 Iterations = 9000	100	61.64	6537.056	106.290

- From the table, it can be seen that Logistic Regression gives us the best model since it gives us the best profit on testing data as well as Recall on testing value is better.

- Below is the Profit Curve of all models (best performer from all model is considered). From the curve also, it is clear that Logistic Regression gives the best model among all.
- The profit curve of different models have been provided in Profit Curve.xlsx file.
- Also, here is the Excel file with the profit curve of different models

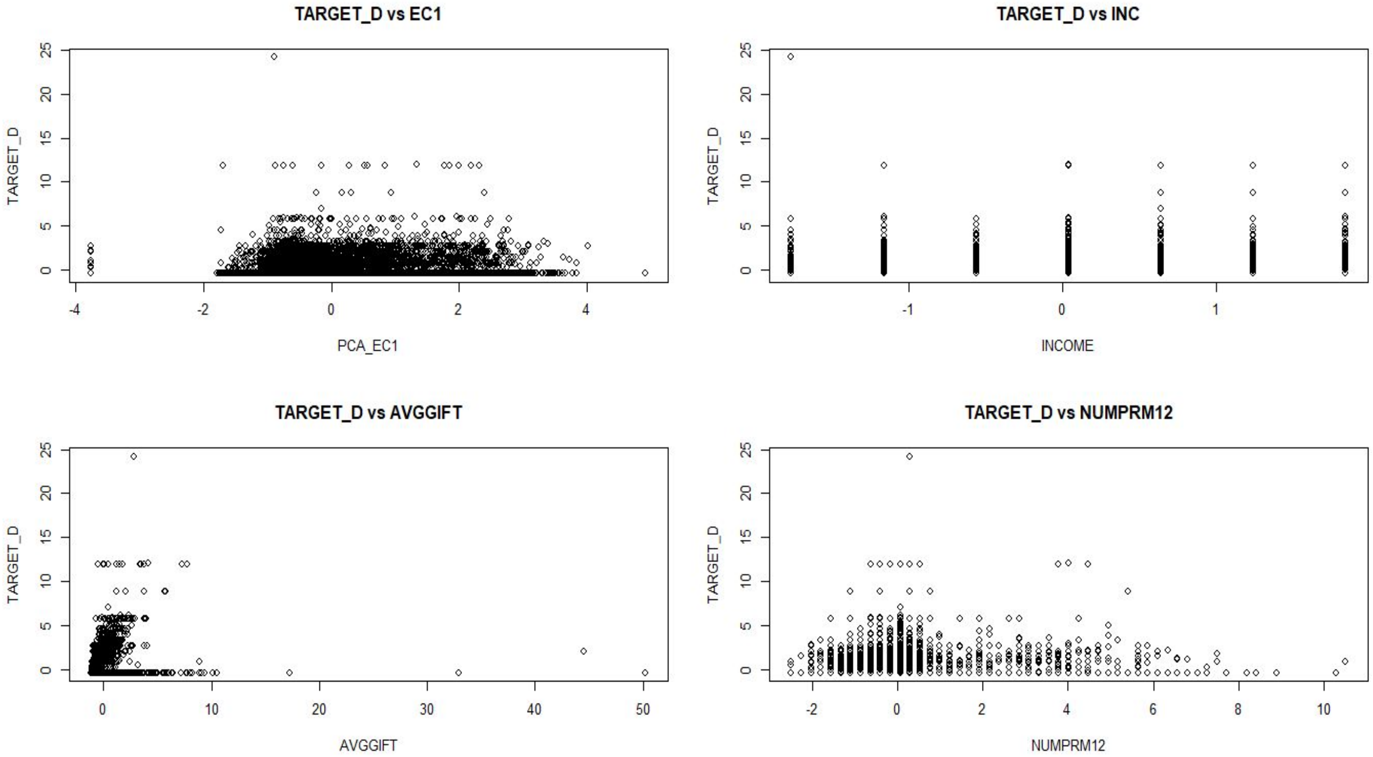


Assignment Question 2.2(A)

- To combine the response model as well as donation amount, the idea is to find the best model for TARGET_D variable - in the similar way best model for TARGET_B was approached. Once, it has been obtained, we can multiply the probability of individuals being a donor to the expected amount of the donation. This will give us the total expected profit.

Assignment Question 2.2(B) & (C)

- The TARGET_D variable is a continuous valued label attribute which tells us about the donation amount donated by each donor. The TARGET_D variable has values only where TARGET_B (Response = 1) has a values of 1 (Donor). In order to build a model on TARGET_D, the first task was to select the relevant attributes which would speak about the giving history of the donor and also fit the model. To estimate the amount of donation we decided to train a Linear Regression model as TARGET_D is a continuous valued numeric attribute and not a categorical one. Also, the predictors are more likely to have a linear relationship with the TARGET_D variable. This can be seen from the below few scatterplots which we plotted using RStudio. Ignoring the outliers, we understand that a linearity exists between the predictors and the TARGET_D variable.

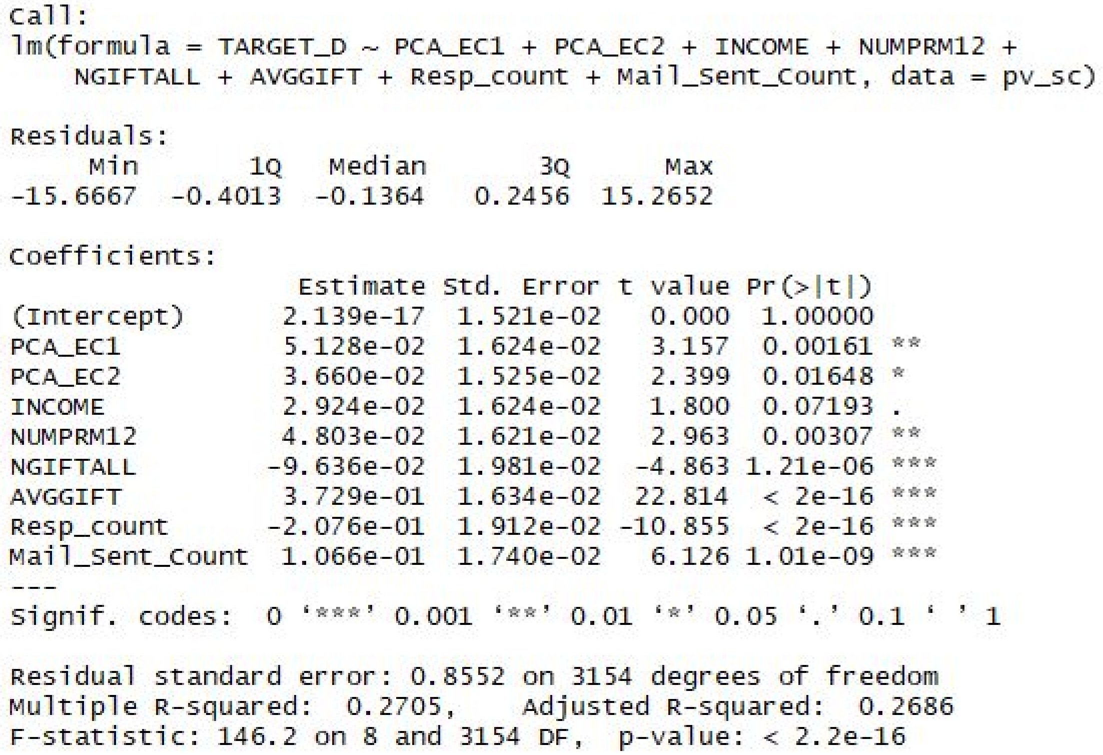


- Initially, we started with our dataset having 15006 observations and 55 variables. After selecting only those observations where our TARGET_B is 1 we were left with 3163 observations. As we were performing linear regression we decided, to consider only the numeric variables and dropped the categorical attributes as they would give poor results in a linear model. After this step our dataset was reduced to 38 variables and we then performed normalization on the values of TARGET_D and predictor variables in order to get them in the same normalized range.

- Variable Reduction using Backward and Forward subset selection:
  We started with building the model on all the 37 variables against the dependent variable

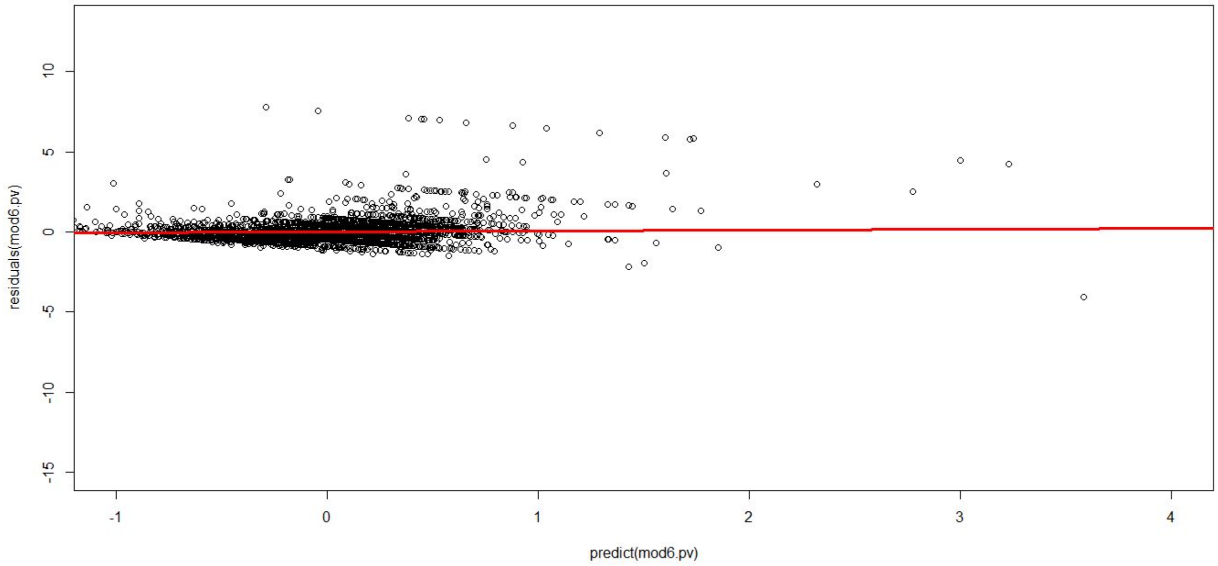
TARGET_D. The results we received was a Residual Standard Error of 0.8202 and a R-squared value of 32.72%. The statistically significant predictors were ODATEDW, NUMPROM, NUMPRM12, NGIFTALL, AVGGIFT, Resp_count, Mail_sent_count, RFA_Rate_Average and TARGET_B.

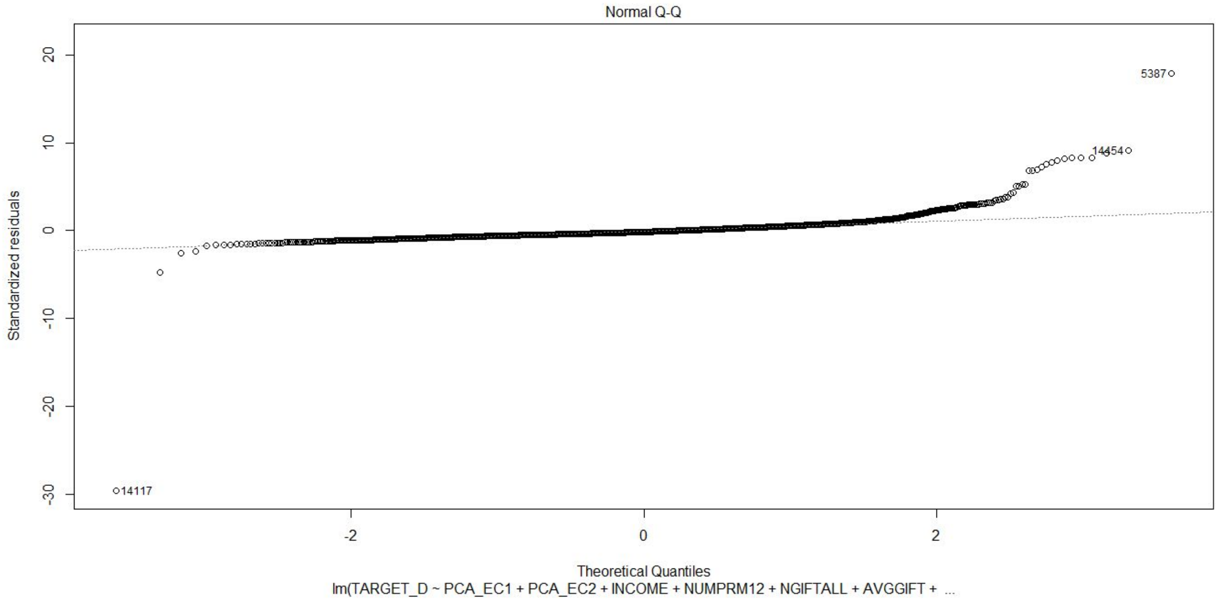
- We then followed a Forward selection process building subsequent models by adding each subset of predictors at a time and assessing the performance. After building subsequent models on different variables, we got the below mentioned set with an R-squared value of 26.86% and RSE of 0.8552. Though the previous model built on all the 37 variables gave a better RSE and R-squared value, it had a large number of statistically insignificant variables which had to be dropped from the model. We had to drop the ‘RFA_Rate_Average’ variable as it had a very high correlation of 0.86 with the ‘Mail_Sent_Count’ thereby providing similar proportion of information.



```
Call:
lm(formula = TARGET_D ~ PCA_EC1 + PCA_EC2 + INCOME + NUMPRM12 +
    NGIFTALL + AVGGIFT + Resp_count + Mail_Sent_Count, data = pv_sc)

Residuals:
     Min       1Q   Median       3Q      Max
-15.6667  -0.4013  -0.1364   0.2456  15.2652

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      2.139e-17  1.521e-02   0.000  1.00000
PCA_EC1          5.128e-02  1.624e-02   3.157  0.00161 **
PCA_EC2          3.660e-02  1.525e-02   2.399  0.01648 *
INCOME           2.924e-02  1.624e-02   1.800  0.07193 .
NUMPRM12         4.803e-02  1.621e-02   2.963  0.00307 **
NGIFTALL        -9.636e-02  1.981e-02  -4.863 1.21e-06 ***
AVGGIFT          3.729e-01  1.634e-02  22.814  < 2e-16 ***
Resp_count      -2.076e-01  1.912e-02 -10.855  < 2e-16 ***
Mail_Sent_Count  1.066e-01  1.740e-02   6.126 1.01e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8552 on 3154 degrees of freedom
Multiple R-squared:  0.2705,    Adjusted R-squared:  0.2686
F-statistic: 146.2 on 8 and 3154 DF,  p-value: < 2.2e-16
```

- After building the above model, we checked for the problem of multicollinearity using the Variance Inflation Factor (VIF) and omitting any outliers to get a better result. We got the below plots to get our model for variable selection:





- From the above model we shortlisted the following variables for the TARGET_D model:
  - **PCA_EC1**: Principal Components for Percent of Adults completed Education
  - **INCOME**: Household Income
  - **NUMPRM12**: Number of promotions received in the last 12 months
  - **NGIFTALL**: Number of lifetime gifts to date.

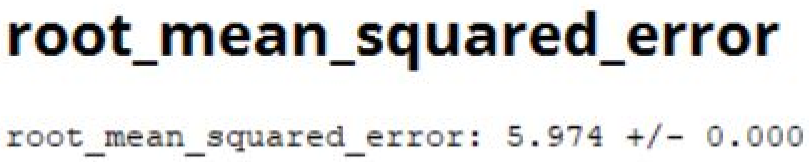
    - **AVGGIFT**: Average Dollar Amount of Gifts to date.
    - **Resp_Count**: Total number of responses provided by the individual.
    - **Mail_Sent_Count**: Total number of mails sent to the individual.

- The non-donors (TARGET_B=0) were not considered in developing the model since they bear no significance in predicting the donation amount. Also, few outliers were identified in the TARGET_D. The identification method used was 68-95-99.7 rule. So values with more than 3 standard deviation away from the median were dropped from developing the model.

- Now that we have obtained the model for TARGET_D, we joined our TARGET_B best model (Logistic Regression) with TARGET_D model using the CONTROLN attribute which uniquely identifies each individual.

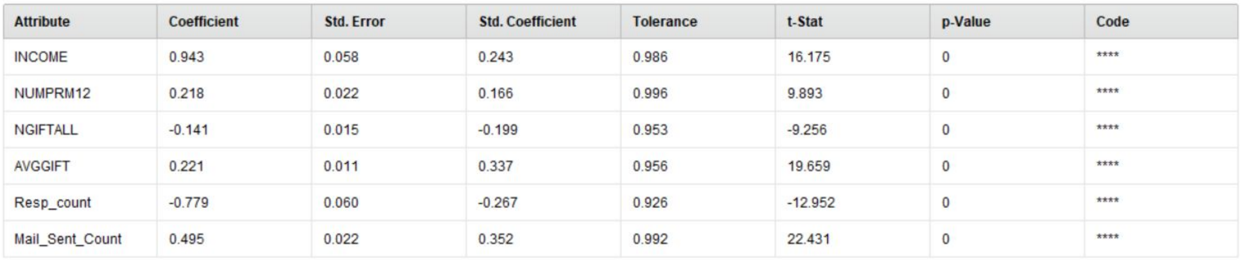
- Once joined, we multiplied the Confidence (1) value which we got from the Linear Regression Model to the prediction amount of TARGET_D. And the expected profit was calculated for the expected donation amount >0.82. (We can ignore the donation amount <0.82 for the profit calculation, since our cost in 0.82)

- After running the dataset on the model, we got the expected profit of 14524.4 in the training data & 9249.4 in the test data. Also, the RMSE value recorded was 5.974. Also, for the confirmation, p-value for all used variables was checked and they all were found statistically significant.



root_mean_squared_error

```
root_mean_squared_error: 5.974 +/- 0.000
```



Attribute	Coefficient	Std. Error	Std. Coefficient	Tolerance	t-Stat	p-Value	Code
INCOME	0.943	0.058	0.243	0.986	16.175	0	****
NUMPRM12	0.218	0.022	0.166	0.996	9.893	0	****
NGIFTALL	-0.141	0.015	-0.199	0.953	-9.256	0	****
AVGGIFT	0.221	0.011	0.337	0.956	19.659	0	****
Resp_count	-0.779	0.060	-0.267	0.926	-12.952	0	****
Mail_Sent_Count	0.495	0.022	0.352	0.992	22.431	0	****

- Given below is the total expected profit and total actual profit for different models on the testing dataset.

Model	**Expected Profit**	**Actual Profit**
Logistic Regression	9249.4	14524.4
SVM	13905	14524.4
Random Forest	3094.6	14524.4
Baggin	3094.6	14524.4
Boosting	3094.6	14524.4

- From the above table, we can see that SVM gives the better profit on testing dataset than any other model.
- Thus, we can say that SVM is a better model here than Logistic Regression (which we found best in question 2.1)
- However, here the overall profit is more than what we found in question 2.1, because earlier, we found the profit from all donors and the average donation amount, thus the profit was found to be low. However, in this case, the profit was found only for the individual with TARGET_B=1 and with the expected donation amount - which in turn gives higher profit.

Assignment Question 3

- We have used the prediction model using SVM for predicting the total profit (From question 2.2 (C))
- In order to apply the model on unseen dataset in pva_futureData_forScoring.csv file , we have used the repository of the pre-processed dataset on our model.
- However, since here, the data corresponds to 5% response rate, the prediction will have to be made on the cost & profit of weighted sampling. We cannot consider unweighted profit & cost.
- So Weighted Profit = $12.32 & Weighted Cost = $0.68
- The total predicted profit we are getting on the validation data is **4730.9**
- Along with this report, TestMaxProfit file has been provided, which has the data for expected donation amount for individuals against their CONTROLN identifier.

  Other 3 files (TrainMaxProfit, TestAggregate, TrainAggregate) provide the details about same details for training data as well as the aggregate profit for test & train data.