A

Group Project Report

on

Warm Up: Predict Blood Donations

By

Group -3 Team Y

Chandu Yerragopu

Saran Prasad Balasubramaniam

Wen Xie

Under the guidance of
Dr. Paige Rutner, Ph.D.
Course: ISQS 6349 Predictive Analytics

TEXAS TECH UNIVERSITY
Rawls College *of* Business™

Table of contents

1 Introduction

Blood donation has been around for a long time. The first successful recorded transfusion was between two dogs in 1665, and the first medical use of human blood in a transfusion occurred in 1818. Even today, donated blood remains a critical resource during emergencies.

Today in the developed world, most blood donors are unpaid volunteers who donate blood for a community supply. In poorer countries, established supplies are limited and donors usually give blood when family or friends need a transfusion (directed donation). Many donors donate as an act of charity, but in countries that allow paid donation some donors are paid, and in some cases there are incentives other than money such as paid time off from work. Donors can also have blood drawn for their own future use (autologous donation). Donating is relatively safe, but some donors have bruising where the needle is inserted or may feel faint.

1.1 Dataset source

Our dataset is from a mobile blood donation vehicle in Taiwan. The Blood Transfusion Service Center drives to different universities and collects blood as part of a blood drive. We want to predict whether or not a donor will give blood the next time the vehicle comes to campus. Data is courtesy of Yeh, I-Cheng via the UCI Machine Learning repository:

Yeh, I-Cheng, Yang, King-Jang, and Ting, Tao-Ming, "Knowledge discovery on RFM model using Bernoulli sequence, "Expert Systems with Applications, 2008, doi:10.1016/j.eswa.2008.07.018.

It has the below predictor variables:

- Months since Last Donation: this is the number of monthis since this donor's most recent donation.
- Number of Donations: this is the total number of donations that the donor has made.
- Total Volume Donated: this is the total amound of blood that the donor has donated in cubuc centimeters.

- Months since First Donation: this is the number of months since the donor's first donation.

1.2 How data looks

Below is the format how training data looks like:

	A	B	C	D	E	F
1		Months since Last Donation	Number of Donations	Total Volume Donated (c.c.)	Months since First Donation	Made Donation in March 2007
2	619	2	50	12500	98	1
3	664	0	13	3250	28	1
4	441	1	16	4000	35	1
5	160	2	20	5000	45	1
6	358	1	24	6000	77	0
7	335	4	4	1000	4	0
8	47	2	7	1750	14	1
9	164	1	12	3000	35	0
10	736	5	46	11500	98	1
11	436	0	3	750	4	0
12	460	2	10	2500	28	1
13	285	1	13	3250	47	0
14	499	2	6	1500	15	1
15	356	2	5	1250	11	1
16	40	2	14	3500	48	1
17	191	2	15	3750	49	1

In this, the first column indicates a random number of the person donating the blood. And the next 4 are the predictor variables (independent variables). The last column “Made Donation in March 2007” is the columns which indicates a person donates blood or not.

Now, this is how test data looks like:

	A	B	C	D	E	F
1		Months since Last Donation	Number of Donations	Total Volume Donated (c.c.)	Months since First Donation	
2	659	2	12	3000	52	
3	276	21	7	1750	38	
4	263	4	1	250	4	
5	303	11	11	2750	38	
6	83	4	12	3000	34	
7	500	3	21	5250	42	
8	530	4	2	500	4	
9	244	14	1	250	14	
10	249	23	2	500	87	
11	728	14	4	1000	64	
12	129	13	3	750	16	
13	534	11	7	1750	62	
14	317	5	11	2750	75	
15	401	4	1	250	4	
16	696	4	4	1000	26	
17	192	11	1	250	11	

Here, in this data we need to predict the dependent variable whether the person donates blood in march 2007 or not.

1.3 Problem Description

The goal is to predict the last column, whether he/she donated blood in March 2007.

1.4 Significance of the problem

In the United States, the American Red Cross is a good resource for information about donating blood. According to their website:

- Every two seconds someone in the U.S. needs blood.
- More than 41,000 blood donations are needed every day.
- A total of 30 million blood components are transfused each year in the U.S.
- The blood used in an emergency is already on the shelves before the event occurs.
- Sickle cell disease affects more than 70,000 people in the U.S. About 1,000 babies are born with the disease each year. Sickle cell patients can require frequent blood transfusions throughout their lives.

- More than 1.6 million people were diagnosed with cancer last year. Many of them will need blood, sometimes daily, during their chemotherapy treatment.
- A single car accident victim can require as many as 100 pints of blood.

1.5 Applicability to Data Scientists

Good data-driven systems for tracking and predicting donations and supply needs can improve the entire supply chain, making sure that more patients get the blood transfusions they need.

2 Literature

2.1 Article 1

Article Source: Predicting Blood Donations Using Machine Learning Techniques, Deepti Bahel, Prerana Ghosh, Arundhyoti Sarkar, Matthew A. Lanham

Summary of the article:

We study the performance of machine learning algorithms that have not been previously investigated to support this problem of blood donation prediction. We build models on clustered data sets using k-means clustering and not using clustering to see if performance is significantly improved using clustering or not. The motivation for this research is that blood demand is gradually increasing by the day due to needed transfusions due to accidents, surgeries, diseases etc. Accurate prediction of the number of blood donors can help medical professionals know the future supply of blood and plan accordingly to entice voluntary blood donors to meet demand. We found that in a non-clustered 5-fold cross-validated logit model led to the best test set AUC (72.6%), which beat other studies. Using k-Means clustering with k=5 consistently led to poorer results than non-clustering. Also, those focused on best specificity could achieve 97.34% using a clustered LDA model. Our current solution is within the top 8% of all current participants in the DataDriven.org blood prediction competition.

Keywords: Blood Donations, Health

2.2 Article 2

Article Source: Knowledge And Attitude Regarding Blood Donation In Rural Puducherry, India by Umakant G Shidam, Subitha Lakshminarayanan, Suman Saurabh, Gautam Roy

Summary of the article:

Out of 288 respondents, 229 subjects (79.5%) were aware that blood could be donated, and only 14.8% of them knew about the correct frequency of blood donation. Around 80% of these subjects felt that the victims of road traffic accident required blood transfusion. Among those who were aware that blood could be donated, 40 subjects (17.5%) had donated blood in past. Most of them had donated blood for their relatives (55%). Among

non-donors the most common reason for not donating blood was “never considered” (34.2%). However, three fourth of the non-donors have shown their willingness to donate blood in future. Electronic and print media were found to be the most common source of knowledge.

Though the awareness regarding blood donation was high, the practice of voluntary blood donation was remarka-bly low. Education and motivation through various media is rec-ommended to eliminate misbelieves and to reinforce positive atti-tudes towards blood donation.

Keywords: Blood donation, Knowledge, Attitude, Blood donor

2.3 Article 3

Article Source:

A Study to assess the Knowledge and Attitude regarding Blood Donation among the General Public in a Selected Urban Area of New Delhi by Rajlaxmi Nishant Kurian, Shilpi Sarkar

Summary of the article:

The study was conducted to assess the knowledge and attitude regarding blood donation among the general public in a selected urban area of New Delhi. A non-experimental research approach with descriptive survey design was used to meet the objectives. The tools used to collect the data were a structured questionnaire and an attitude scale on blood donation. It was found that among the subjects, 55% were in the age group of 21-30 years and 51% were males. Maximum (72%) lived in joint families and 27% belonged to nuclear families. 54% were graduates and 25% were high school pass. 49% of the subjects were earning more than Rs 25000. 54% were single and 38% were married. 45% were private employees. Maximum (74%) of the subjects had average knowledge; 11% displayed good knowledge and 15% had poor knowledge regarding blood donation. Regarding the attitude; 58% had strongly positive and 22% had positive attitude towards blood donation. 9% showed negative attitude and 11% had strongly negative attitude towards blood donation. The study findings revealed that maximum (74%) subjects had average knowledge regarding blood donation and maximum (58%) expressed positive attitude towards blood donation.

Keywords:

Knowledge, Attitude and general public.

2.4 Article 4

Article Source:

Investigating the factors affecting blood donation among Israelis by Ben Natan Merav RN, PhD (Lecturer), Gorkov Lena RN, BA (Nurse)

Summary of the article:

Aim: This study examined whether the Theory of Planned Behavior adds significantly to the prediction of intention and actual blood donation of the general Israeli population.

Background: In most developed countries and in Israel in particular there is a chronic shortage of blood for transfusions. This raises questions about methods of increasing blood donations.

Design: This is a correlational quantitative study.

Methods: A questionnaire was created based on a review of the literature and the Theory of

Planned Behavior. The questionnaire was distributed among a convenience sample of 190 Israeli Jewish men and women, aged 17–60.

Results: Israelis' perceived behavioral control of their blood donations, their subjective norms and their attitude regarding blood donation, predicted their intention to donate blood. It seems that intention predicted actual blood donations. A conspicuous finding is that members of the Ethiopian Jewish community displayed an extremely limited intention to donate blood.

Conclusions: The results of this study show that a number of various educational and practical strategies may be used to encourage the population to donate blood. These include: reducing perceived barriers, directing interventions specifically at the population most likely to donate blood and forming a reserve of regular donors.

Keywords: Blood donation, Israel ,Theory of Planned Behavior

2.5 Article 5

Article Source: Knowledge, Attitudes, and Motivations towards Blood Donation among King Abdulaziz Medical City Population

Summary of the article:

Background: Blood donation is remarkably safe medical procedure. However, attitudes, beliefs, and level of knowledge may affect it. Objectives. To measure the level of knowledge regarding blood donation, find out positive and negative attitudes, identify the obstacles, and suggest some motivational factors.

Methodology: A cross-sectional study was conducted at King Abdulaziz Medical City (KAMC). Participants were selected by convenient nonrandom sampling technique. A self-created questionnaire was used for data collection. Results. The study included 349 individuals. About 45.8% of the participants claimed that they have a history of blood donation. Reported causes for not donating blood were blood donation not crossing their mind (52.4%), no time for donation (45%), and difficulty in accessing blood donation center (41.3%). Reported motivating factors for donating blood were one day off (81.4%), mobile blood donation caravans in public areas (79.1%), token gifts (31.5%), and finally paying money (18.9%). Conclusion. People in the age group 31–50 years, males, higher education and military were more likely to donate blood as well as People who showed higher knowledge level and positive attitude towards blood donation. More educational programs to increase the awareness in specific targeted populations and also to focus on some motivational factors are recommended.

3 Data Visualization and Summary Statistics

The first step in any prediction problem is to load the data and then identify the varies trends in the given data.

By observing the data, we see it is cross-sectional in nature. We proceed with the below statistics of the data:

```
> train.data <- read.csv("E:/Data Science/PA/DrivenData Project/training.csv")
```

```
> traindata<- train.data[2:6]
> head(traindata)
  Months.since.Last.Donation Number.of.Donations Total.Volume.Donated..c.c..
1                          2                  50                       12500
2                          0                  13                        3250
3                          1                  16                        4000
4                          2                  20                        5000
5                          1                  24                        6000
6                          4                   4                        1000
  Months.since.First.Donation Made.Donation.in.March.2007
1                          98                           1
2                          28                           1
3                          35                           1
4                          45                           1
5                          77                           0
6                           4                           0
```

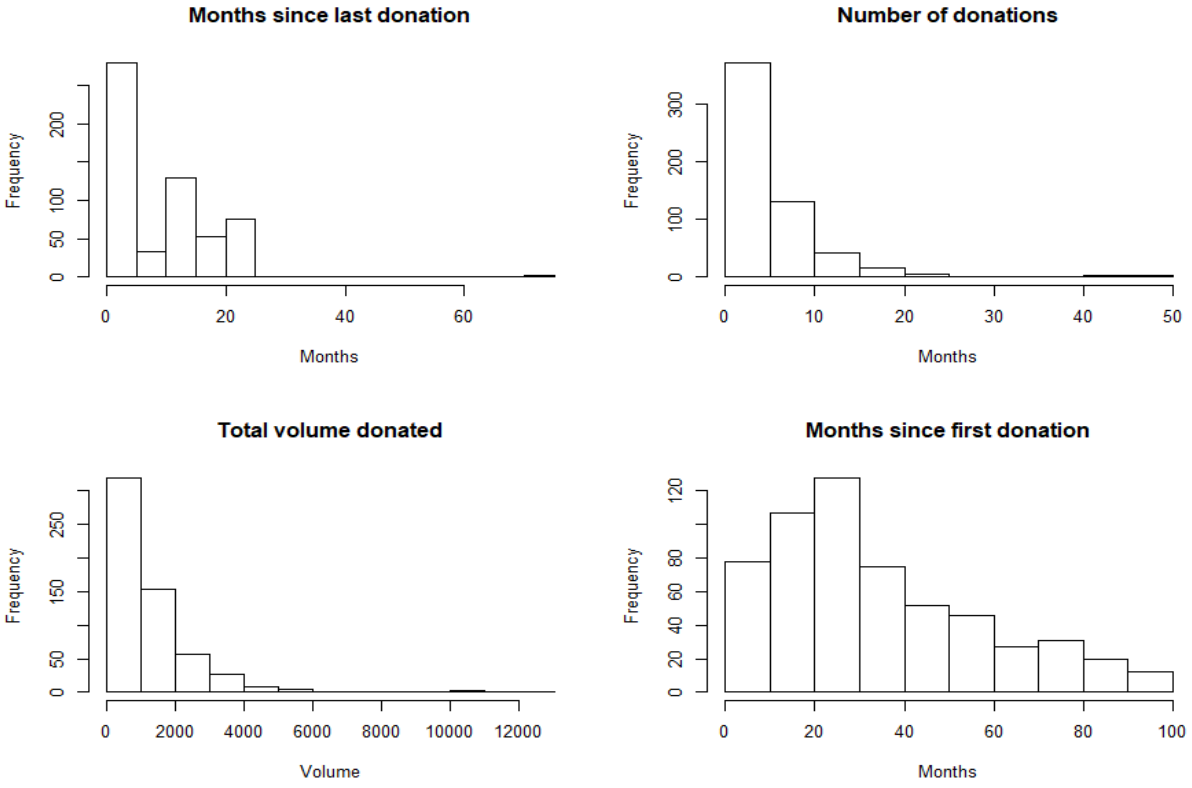
Summary of the data:

```
> summary(traindata)
 Months.since.Last.Donation Number.of.Donations Total.Volume.Donated..c.c..
 Min.   : 0.000             Min.   : 1.000      Min.   :  250
 1st Qu.: 2.000             1st Qu.: 2.000      1st Qu.:  500
 Median : 7.000             Median : 4.000      Median : 1000
 Mean   : 9.439             Mean   : 5.427      Mean   : 1357
 3rd Qu.:14.000             3rd Qu.: 7.000      3rd Qu.: 1750
 Max.   :74.000             Max.   :50.000      Max.   :12500
 Months.since.First.Donation Made.Donation.in.March.2007
 Min.   : 2.00               Min.   :0.0000
 1st Qu.:16.00               1st Qu.:0.0000
 Median :28.00               Median :0.0000
 Mean   :34.05               Mean   :0.2396
 3rd Qu.:49.25               3rd Qu.:0.0000
 Max.   :98.00               Max.   :1.0000
```

From the summary of the training data we can see that all the columns have some outliers in them.

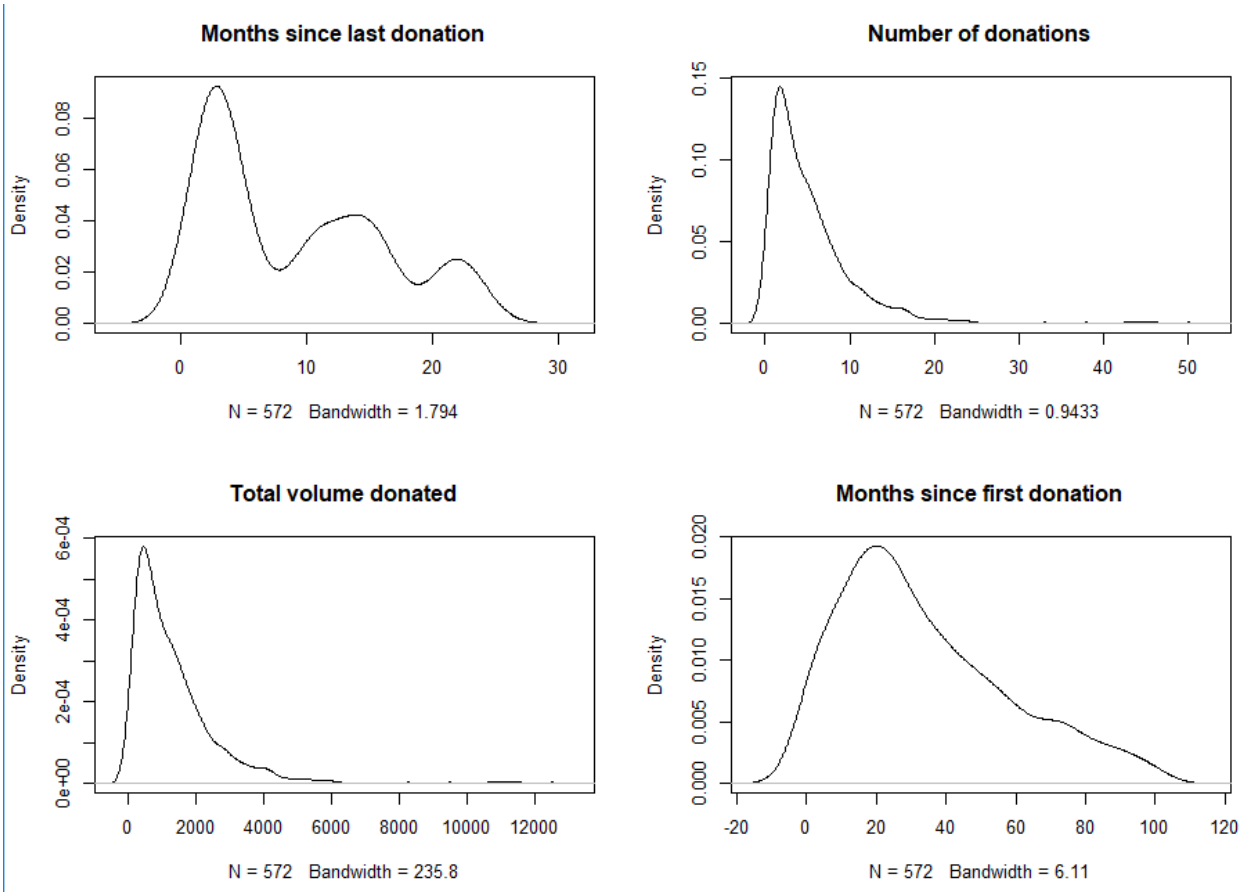
Histogram of the variables that are in the training dataset:

```
# 1) histogram of the independent values:

par(mfrow=c(2,2))
hist(train.data$Months.since.Last.Donation,main = "Months since last donation",
     xlab = "Months")
hist(train.data$Number.of.Donations,main = "Number of donations",
     xlab = "Months")
hist(train.data$Total.Volume.Donated..c.c..,main = "Total volume donated",
     xlab = "Volume")
hist(train.data$Months.since.First.Donation,main = "Months since first donation",
     xlab = "Months")
```



Density plot of the variables that are in the training dataset:

```
# 2) density plot:
plot(density(train.data$Months.since.Last.Donation),main = "Months since last donation")
plot(density(train.data$Number.of.Donations),main = "Number of donations")
plot(density(train.data$Total.Volume.Donated..c.c..),main = "Total volume donated")
plot(density(train.data$Months.since.First.Donation),main = "Months since first donation")
```


Months since last donation
Density
0.00
0.02
0.04
0.06
0.08
0
10
20
30
N = 572 Bandwidth = 1.794
Number of donations
Density
0.00
0.05
0.10
0.15
0
10
20
30
40
50
N = 572 Bandwidth = 0.9433
Total volume donated
Density
0e+00
2e-04
4e-04
6e-04
0
2000
4000
6000
8000
10000
12000
N = 572 Bandwidth = 235.8
Months since first donation
Density
0.000
0.005
0.010
0.015
0.020
-20
0
20
40
60
80
100
120
N = 572 Bandwidth = 6.11

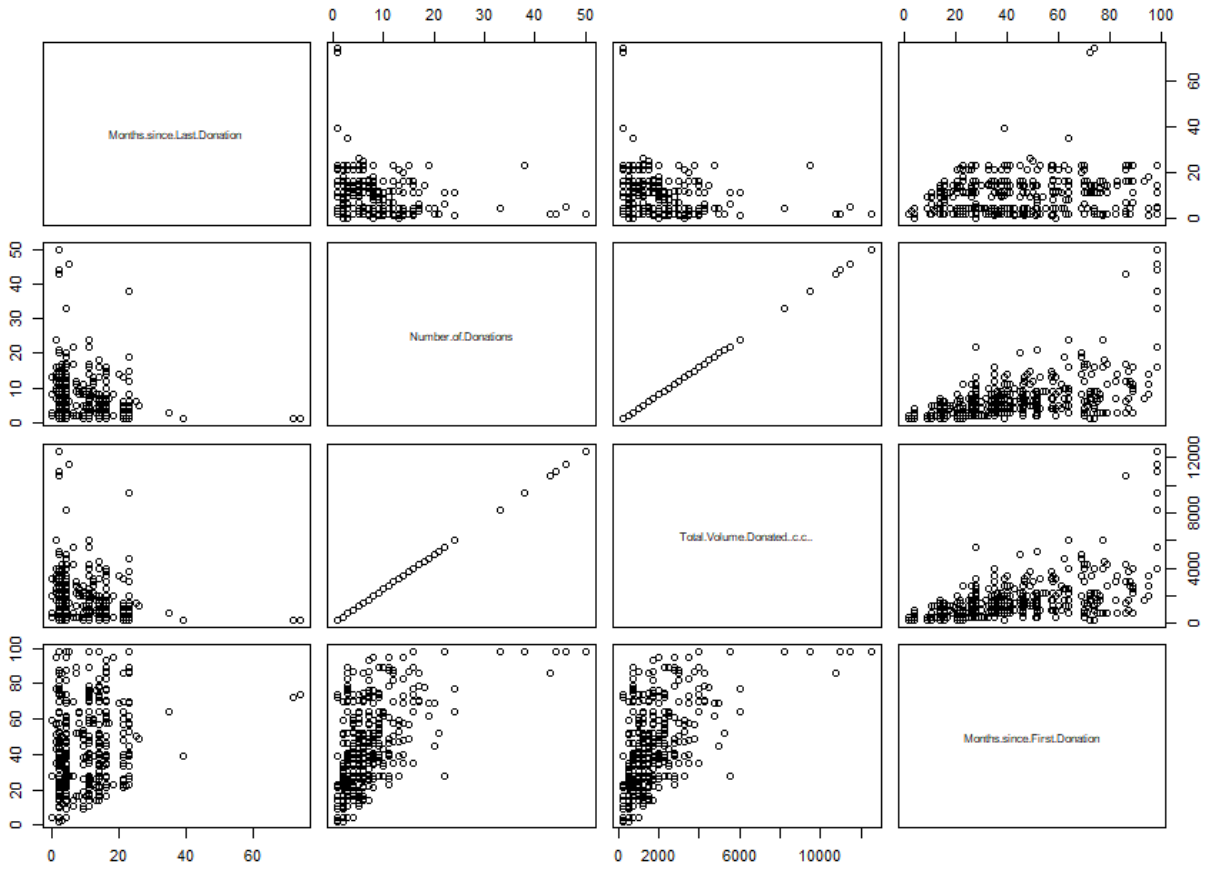
4 Data Cleaning

As the data downloaded for this project from the competition website is already almost clean.

It is a good idea to see if any of the data are highly correlated since highly correlated features may affect our final model performance.

```
> cor(test.data[,2:5])
                         Months.since.Last.Donation Number.of.Donations Total.Volume.Donated..c.c..
Months.since.Last.Donation                  1.0000000         -0.2257519                 -0.2257519
Number.of.Donations                        -0.2257519          1.0000000                  1.0000000
Total.Volume.Donated..c.c..                -0.2257519          1.0000000                  1.0000000
Months.since.First.Donation                 0.0994124          0.6827566                  0.6827566
                         Months.since.First.Donation
Months.since.Last.Donation                 0.0994124
Number.of.Donations                        0.6827566
Total.Volume.Donated..c.c..                0.6827566
Months.since.First.Donation                1.0000000
```

Plot of the correlation of the data:



Covariance of the data:

```
> cov(traindata)
                                Months.since.Last.Donation Number.of.Donations
Months.since.Last.Donation                      66.8380405          -7.4957428
Number.of.Donations                             -7.4957428          32.9477174
Total.Volume.Donated..c.c..                  -1873.9356884        8236.9293478
Months.since.First.Donation                     37.0195864          86.5158514
Made.Donation.in.March.2007                     -0.9123732           0.5409783
                                Total.Volume.Donated..c.c.. Months.since.First.Donation
Months.since.Last.Donation                       -1873.9357                   37.0195864
Number.of.Donations                               8236.9293                   86.5158514
Total.Volume.Donated..c.c..                    2059232.3370                21628.9628623
Months.since.First.Donation                      21628.9629                  586.9800694
Made.Donation.in.March.2007                        135.2446                   -0.2051268
                                Made.Donation.in.March.2007
Months.since.Last.Donation                       -0.9123732
Number.of.Donations                               0.5409783
Total.Volume.Donated..c.c..                     135.2445652
Months.since.First.Donation                      -0.2051268
Made.Donation.in.March.2007                       0.1825000
```

From the correlation and covariance values we can see that there is a strong correlation between Number.of. Donations and Total.Volume.Donated..c.c.. which makes no sense. So, lets remove the Total.Volume.Donated..c.c.. column from the training data set.

Now, let us proceed with the application of the various predictive models to the above data.

5 Application of Predictive Techniques

We proceed with the application of the predictive models to fit the data and thus use the model to predict whether a person will donate blood or not using that model. As the data is cross-sectional in nature and the prediction is a class based (whether the person donates blood or not), we identified the below models as the best fit for this data.

In each of the below sections, we explain how we fit the model to the data, evaluation the goodness of the fit, calculate various statistics of the data, understand the errors, Calculate AIC, BIC, how to minimize those errors, and save the final output of the model in a .csv file. We have uploaded these .csv files to the competition website and then obtained the ranks. As we applied various techniques by trying to reduce the errors, the rank improved.

The following are the library that have been used in the creating the models.

```
#library used for the project:
library(randomForest)
library(caret)
library(e1071)
library(rpart)
library(forecast)
library(dplyr)
library(plyr)
library(rpart)
library(pROC)
library(Hmisc)
```

5.1 Technique 1 – Logistic Regression

Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

Two models are created using the logistic regression.

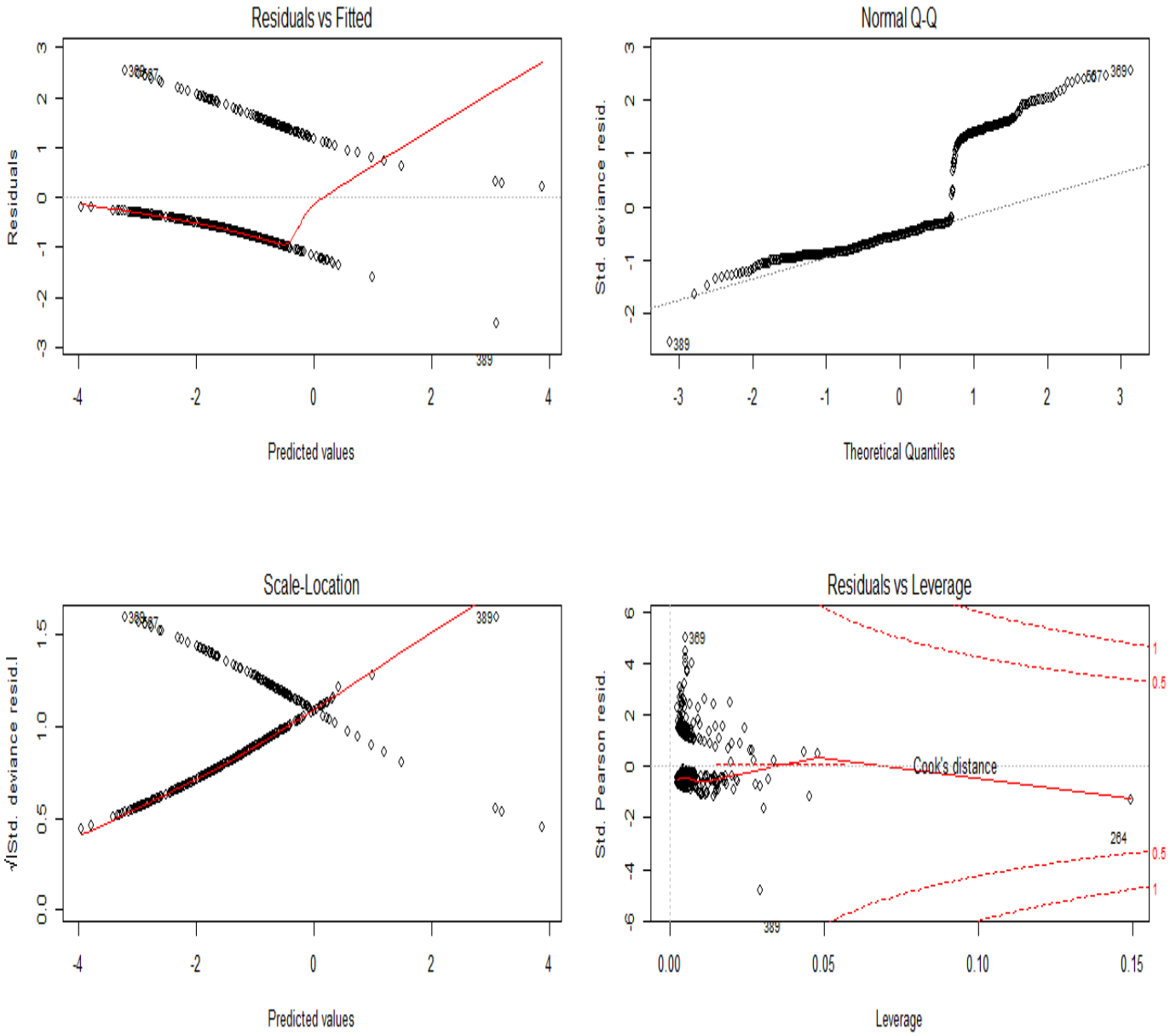
Model 1:

The first one is where logistic regression is applied for the three variables Months.since.Last.Donation, Number.of.Donations and Months.since.First.Donation against the dependent variable Made.Donation.in.March.2007.

```
log.model1 <- glm(Made.Donation.in.March.2007 ~ Months.since.Last.Donation +
                    Number.of.Donations +
                    Months.since.First.Donation , data = train.data,family = "binomial")
log.model1
```

```
> log.model1

Call:  glm(formula = Made.Donation.in.March.2007 ~ Months.since.Last.Donation +
    Number.of.Donations + Months.since.First.Donation, family = "binomial",
    data = train.data)

Coefficients:
                (Intercept)   Months.since.Last.Donation          Number.of.Donations
                   -0.58564                     -0.09103                      0.12992
Months.since.First.Donation
                   -0.01880

Degrees of Freedom: 575 Total (i.e. Null);  572 Residual
Null Deviance:      634.3
Residual Deviance: 556.6        AIC: 564.6
```

Plot for the model1:



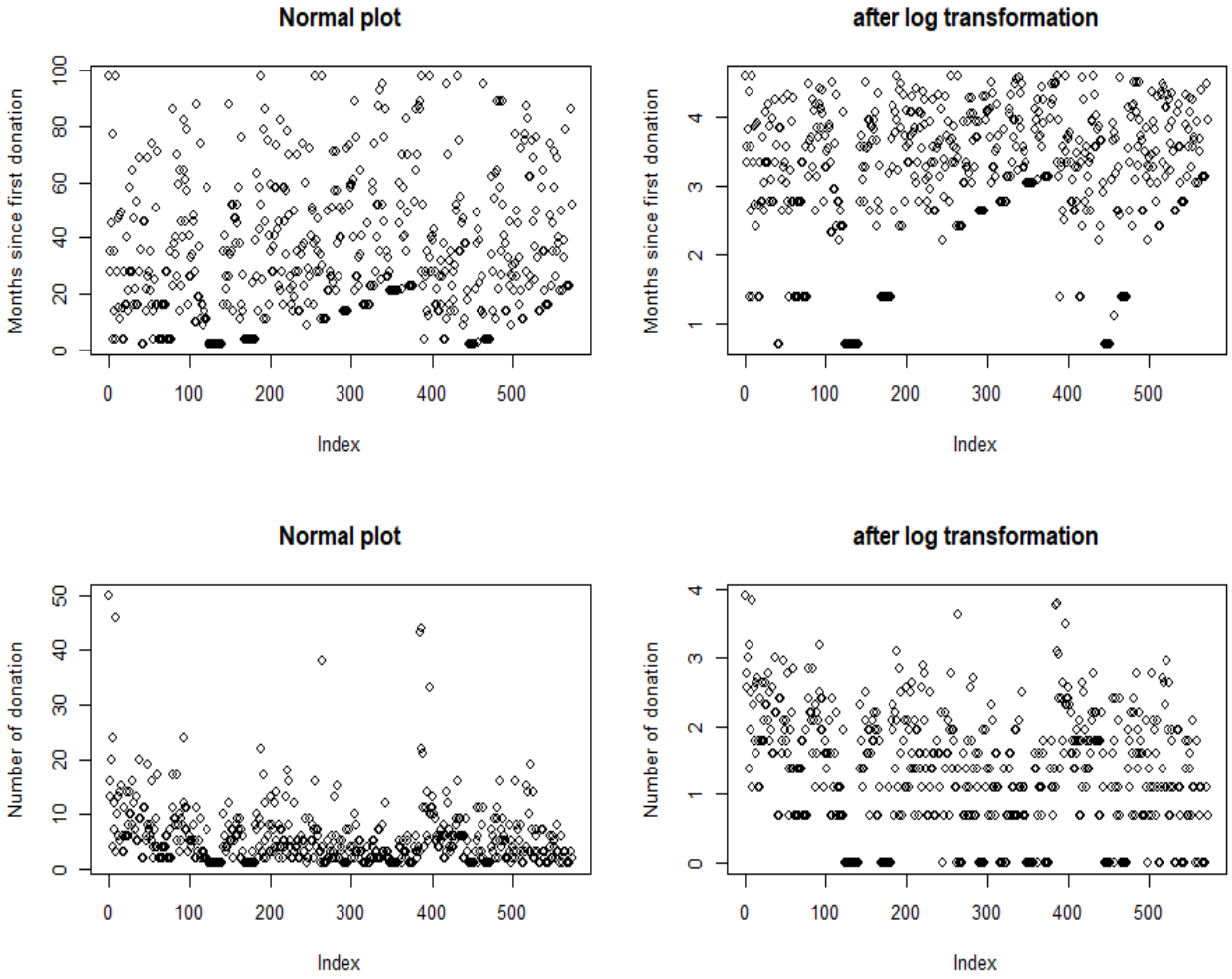
Model 2:

The log transformation can be used to make highly skewed distributions less skewed. This can be valuable both for making patterns in the data more interpretable and for helping to meet the assumptions of inferential statistics.

```
> log.model2 <- glm(Made.Donation.in.March.2007 ~ log(Months.since.Last.Donation) +
+                     log(Number.of.Donations) +
+                     log(Months.since.First.Donation) , data = train.data,family = "binomial")
Error in glm.fit(x = c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  :
  NA/NaN/Inf in 'x'
>
```

As Months.Since.Last.Donation column has zero values in it we are unable to log for this. Hence, we will have to do the log transformation for the other two columns.

Lets the two columns after taking log transformation and compare them with the normal plot of the variables.

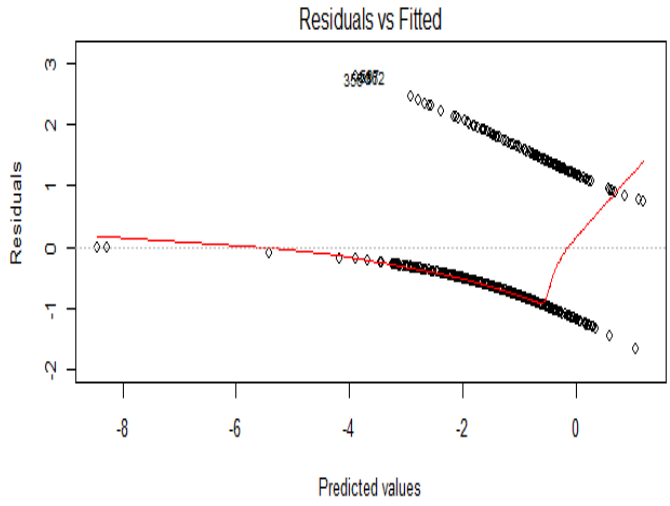


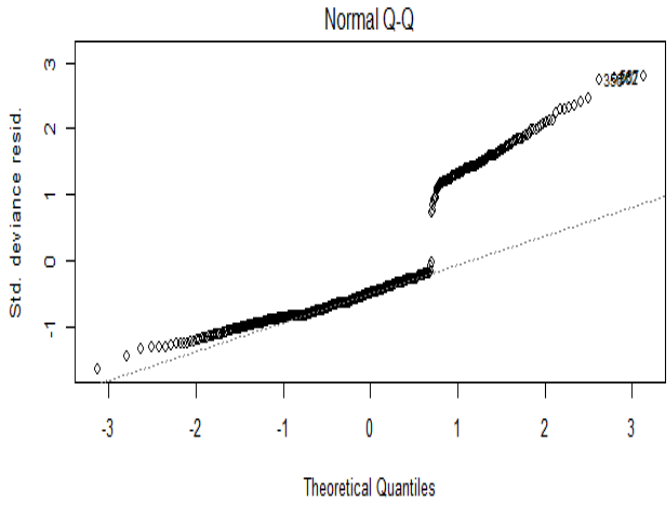
After doing the log transformation, we can see that skewness has reduced considerably.

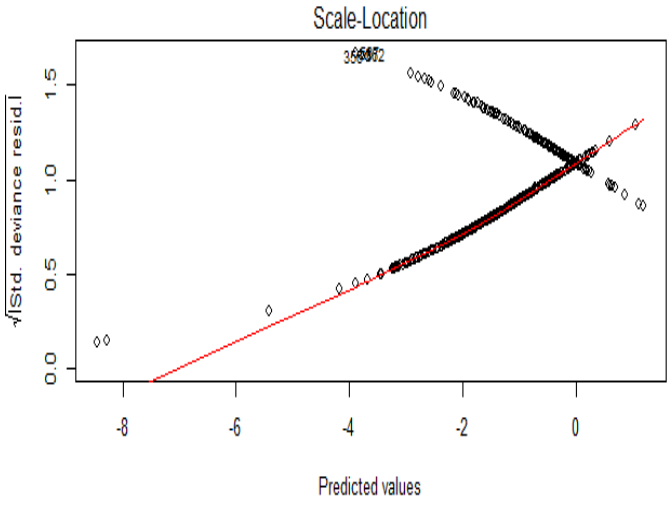
```
log.model2 <- glm(Made.Donation.in.March.2007 ~ Months.since.Last.Donation +
                  log(Number.of.Donations) +
                  log(Months.since.First.Donation) , data = train.data,family = "binomial")
```

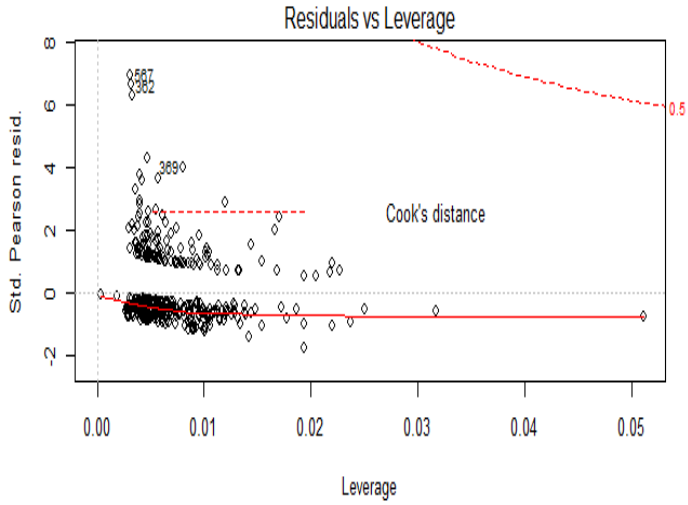
```
> log.model2

Call:  glm(formula = Made.Donation.in.March.2007 ~ Months.since.Last.Donation +
    log(Number.of.Donations) + log(Months.since.First.Donation),
    family = "binomial", data = train.data)

Coefficients:
                      (Intercept)        Months.since.Last.Donation          log(Number.of.Donations)
                          -0.35804                          -0.07634                           1.09659
log(Months.since.First.Donation)
                          -0.56392

Degrees of Freedom: 571 Total (i.e. Null);  568 Residual
Null Deviance:      632.1
Residual Deviance: 549.7        AIC: 557.7
```

Plot for the Model2:









5.1.1 Assessment of the logestic model

i) AIC (Akaike information criterion)

AIC offers an estimate of the relative information lost when a given model is used to represent the process that generated the data. In doing so, it deals with the trade-off between the goodness of fit of the model and the complexity of the model. So, lesser the AIC value better is the model.

Model 1:

```
> AIC(log.model1)
[1] 564.6095
```

Model 2:

```
> AIC(log.model2)
[1] 557.7377
```

ii) BIC (Bayesian information criterion)

The BIC is a criterion for model selection among a finite set of models; the model with the lowest BIC is preferred. It is based, in part, on the likelihood function and it is closely related to the Akaike information criterion (AIC).

Model 1:

```
> model1.BIC <- log.model1$deviance +
+   2*length(log.model1$coefficients)*log(length(log.model1$fitted.values) )
> model1.BIC  #607.4584
[1] 607.3639
```

Model 2:

```
> model2.BIC <- log.model2$deviance +
+   2*length(log.model2$coefficients)*log(length(log.model2$fitted.values) )
> model2.BIC  #600.627
[1] 600.5308
```

iii) Chi-squared distribution

We can also use the residual deviance to test whether the null hypothesis is true (i.e. Logistic regression model provides an adequate fit for the data). This is possible because the deviance is given by the chi-squared value at a certain

degree of freedom. To test for significance, we can find out associated p-values using the below formula in R.

Model 1:

```
> modelChi <- log.model1$null.deviance - log.model1$deviance
> chidf <- log.model1$df.null - log.model1$df.residual
> chisq.prob <- 1-pchisq(modelChi, chidf)
> chisq.prob
[1] 3.330669e-16
```

Model 2:

```
> modelChi2 <- log.model2$null.deviance - log.model2$deviance
> chidf2 <- log.model2$df.null - log.model2$df.residual
> chisq.prob2 <- 1-pchisq(modelChi2, chidf2)
> chisq.prob2
[1] 0
```

Using the above values of residual deviance and DF, we get a p-value of approximately zero for both the models showing that there is a significant lack of evidence to support the null hypothesis.

iv) Confusion matrix

A confusion matrix is a technique for summarizing the performance of a classification algorithm. Calculating a confusion matrix can give you a better idea of what your classification model is getting right and what types of errors it is making. Here the cut off value is taken as 0.5. If the predicted value is above 0.5 then it is taken as 1(the person donates blood) and below 0.5 is taken as 0(the person does not donate blood).

Model 1:

```
train.predict1 <- predict(log.model1,train.data , type = "response")
p1 <- ifelse(train.predict1 > 0.5,1,0)
tab1 <- table(predicted = p1 , actual = train.data$Made.Donation.in.March.2007)
tab1
confusionMatrix(tab1)  #76.22%
```

```
Confusion Matrix and Statistics

          actual
predicted   0   1
        0 421 123
        1  13  15

               Accuracy : 0.7622
                 95% CI : (0.7252, 0.7966)
    No Information Rate : 0.7587
    P-Value [Acc > NIR] : 0.445

                  Kappa : 0.1081
 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.9700
            Specificity : 0.1087
         Pos Pred Value : 0.7739
         Neg Pred Value : 0.5357
             Prevalence : 0.7587
         Detection Rate : 0.7360
   Detection Prevalence : 0.9510
      Balanced Accuracy : 0.5394

       'Positive' Class : 0
```

Acuracy of this model is **76.22%**

Model 2:

```
train.predict2 <- predict(log.model2,train.data , type = "response")
p2 <- ifelse(train.predict2 > 0.5,1,0)
tab2 <- table(predicted = p2 , actual = train.data$Made.Donation.in.March.2007)
tab2
confusionMatrix(tab2)  #77.08%
```

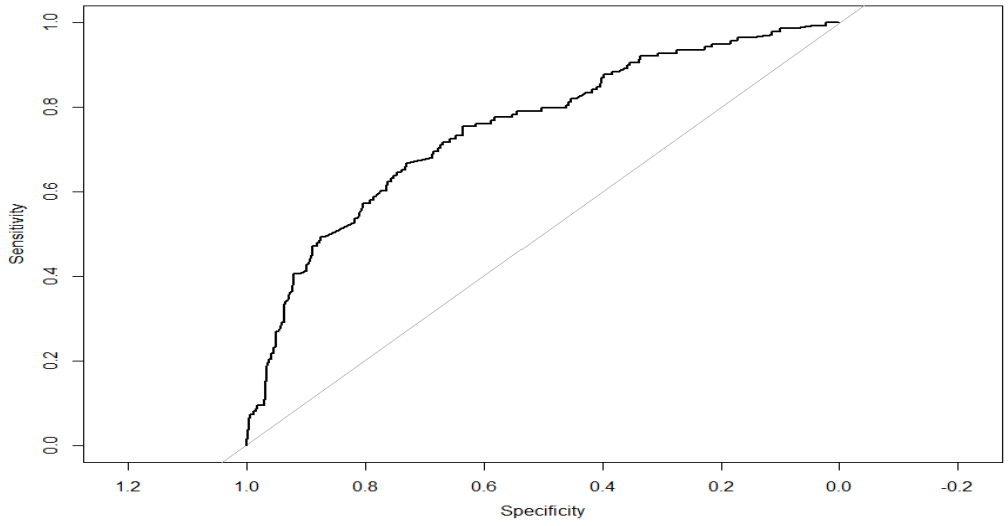
```
> confusionMatrix(tab2)  #77.08%
Confusion Matrix and Statistics

          actual
predicted   0   1
        0 422 116
        1  16  22

               Accuracy : 0.7708
                 95% CI : (0.7343, 0.8046)
    No Information Rate : 0.7604
    P-Value [Acc > NIR] : 0.2978

                  Kappa : 0.1635
 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.9635
            Specificity : 0.1594
         Pos Pred Value : 0.7844
         Neg Pred Value : 0.5789
             Prevalence : 0.7604
         Detection Rate : 0.7326
   Detection Prevalence : 0.9340
      Balanced Accuracy : 0.5614

       'Positive' Class : 0
```

Accuracy of this model is **77.08%**

v) ROC curve

The area under the ROC curve can give us insight into the predictive ability of the model. If it is equal to 0.5, the model can be thought of as predicting at random (an ROC curve with slope = 1). Values close to 1 indicate that the model has good predictive ability. A similar measure is Somers' Dxy rank correlation between predicted probabilities and observed outcomes.

Model 1:

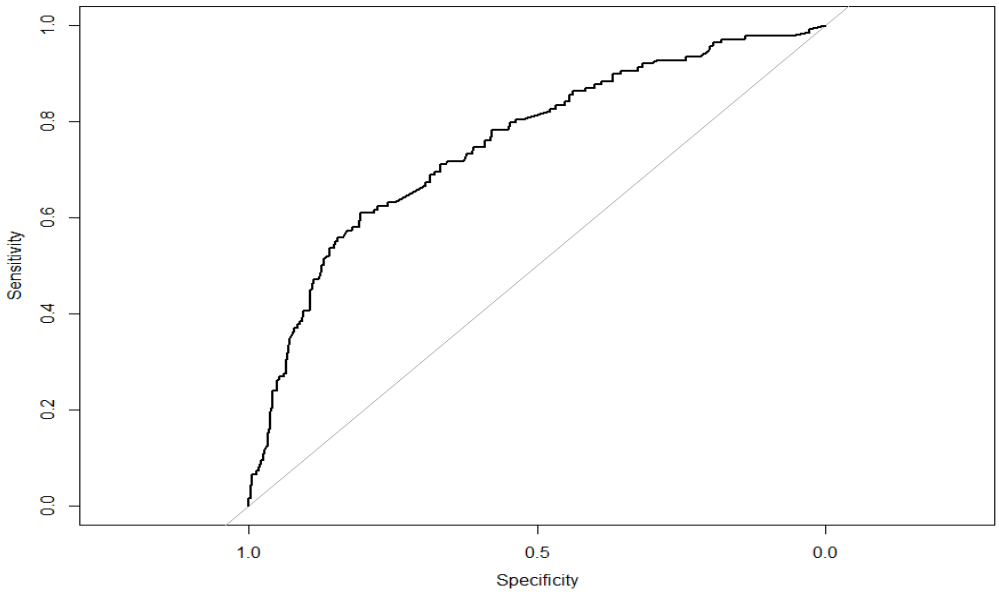


where c is the area under the ROC curve. When Dxy = 0, the model is making random predictions. When Dxy = 1, the model discriminates perfectly. We can get this Dxy and c value by using the somers2() function in the Hmisc library in R.

```
> somers2(fitted(log.model1),train.data$Made.Donation.in.March.2007) #0.7547978 ROC value
          C         Dxy           n     Missing
  0.7467274   0.4934549 572.0000000   0.0000000
```

This shows that the model has 74.6% of area under the ROC curve.

Model 2:




```
> somers2(fitted(log.model2),train.data$Made.Donation.in.March.2007) #0.7547978 ROC value
          C         Dxy           n     Missing
  0.7525713   0.5051426 572.0000000   0.0000000
```

This shows that the model 2 has 75.2% of are under the ROC curve.

vi) VIF Test

VIF is used to detect multicollinearity of the predictors in the model by calculating the variance inflation factors. It is a measure of how much the variance of the estimated regression coefficient is inflated by the existence of correlation among the predictor variables in the model. A VIF of 1 means that there is no correlation among the predictor and the remaining predictor variables, and hence the variance is not inflated at all. The general rule of thumb is that VIFs exceeding 4 warrant further investigations, while VIFs exceeding 10 are signs of serious multicollinearity requiring correction. In R, we can use VIF function to analyze the value of VIF.

Model 1:

```
> vif(log.model1)
 Months.since.Last.Donation         Number.of.Donations Months.since.First.Donation
                   1.084934                    2.121902                    2.203637
```

The outcome shows that all predictors have no significant multicollinearity in the model (all vifs are less than 4) and the predictor Month.Since.Last.Donation is better used as a factor than other predictors in the model.

Model 2:

```
> vif(log.model2)
      Months.since.Last.Donation         log(Number.of.Donations)
                        1.262059                         3.153503
log(Months.since.First.Donation)
                        3.472833
```

As the above result shows, the multicollinearity in model2 is obviously serious than that in model1 although all the values are still less than 4. This means our model may have a big improvement space and we should analyze more on reducing data-based multicollinearity and structural multicollinearity.

vii) DurbinWatsonTest

Durbin-Watson test is used to test the hypothesis that there is no lag one autocorrelation in the residuals. If there is no autocorrelation, the Durbin-Watson distribution is symmetric around 2. A small p-value indicates there is significant autocorrelation remaining in the residuals. In R, we can use durbinWatsonTest function to evaluate the value.

Model 1:

```
> durbinWatsonTest(log.model1,alternative="two.sided")
 lag Autocorrelation D-W Statistic p-value
   1       0.1490203      1.701885       0
 Alternative hypothesis: rho != 0
```

The p-value in the above outcome shows 0 which means there are some remaining autocorrelation in the residuals and the model can be further improved.

Model 2:

```
> durbinWatsonTest(log.model2,alternative="two.sided")
 lag Autocorrelation D-W Statistic p-value
   1       0.1300576      1.738922       0
 Alternative hypothesis: rho != 0
```

The p-value of model2 is 0 which is the same as that in model1. This again to better modify our models.

From assessment of the logistic model from various test we can see that the model2 i.e. the one done after taking the log transformation gives better accuracy then the other normal logistic model.

5.2 Techique 2 – Decision Tree Classifier

A decision tree is a schematic, tree-shaped diagram used to determine a course of action or show a statistical probability. Each branch of the decision tree represents a possible decision, occurrence or reaction. The tree is structured to show how and why one choice may lead to the next, with the use of the branches indicating each option is mutually exclusive.

A decision tree typically starts with a single node, which branches into possible outcomes. Each of those outcomes leads to additional nodes, which branch off into other possibilities. This gives it a tree like shape.

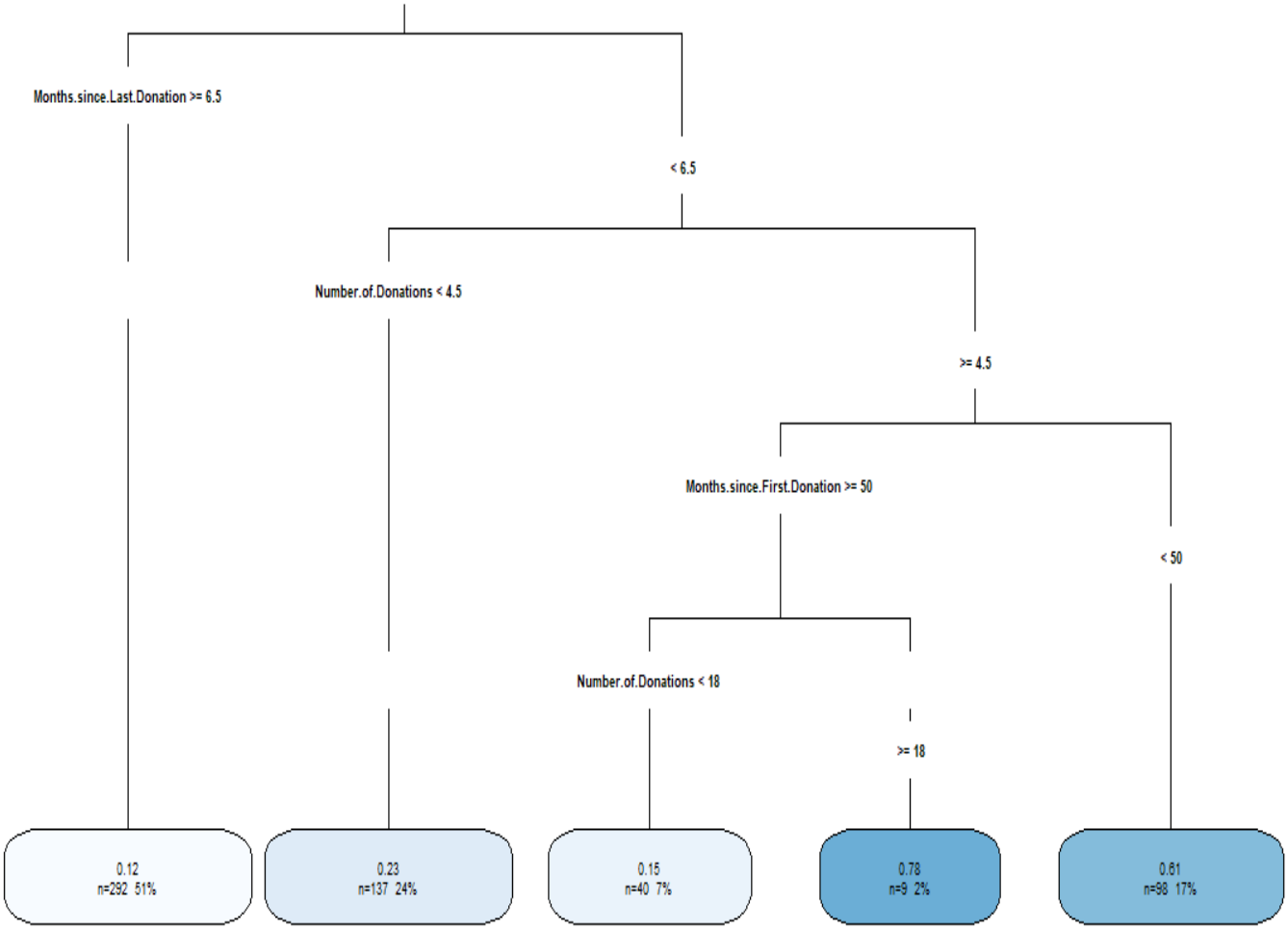
Model:

```
> summary(tree_model1)
Call:
rpart(formula = Made.Donation.in.March.2007 ~ Months.since.Last.Donation +
    Number.of.Donations + Months.since.First.Donation, data = train.data)
  n= 572

          CP nsplit rel error    xerror       xstd
1 0.08408940      0 1.0000000 1.0028209 0.05071942
2 0.04948883      1 0.9159106 0.9438476 0.04824808
3 0.03755240      2 0.8664218 0.9105243 0.05045180
4 0.02765321      3 0.8288694 0.8723504 0.05287049
5 0.01000000      4 0.8012162 0.8398132 0.05386241

Variable importance
Months.since.First.Donation         Number.of.Donations  Months.since.Last.Donation
                         35                          34                          30

Node number 1: 572 observations,    complexity param=0.0840894
  mean=0.2412587, MSE=0.183053
  left son=2 (288 obs) right son=3 (284 obs)
  Primary splits:
      Months.since.Last.Donation  < 6.5  to the right, improve=0.084089400, (0 missing)
      Number.of.Donations         < 4.5  to the left,  improve=0.053846410, (0 missing)
      Months.since.First.Donation < 51.5 to the right, improve=0.007955105, (0 missing)
  Surrogate splits:
      Months.since.First.Donation < 10.5 to the right, agree=0.629, adj=0.254, (0 split)
      Number.of.Donations         < 4.5  to the left,  agree=0.570, adj=0.134, (0 split)
```

Tree diagram:

A diagram used in strategic decision making, valuation or probability calculations. The diagram starts at a single node, with branches emanating to additional nodes, which represent mutually exclusive decisions or events. In the diagram below, the analysis will begin at the first blank node. A decision or event will then lead to node A or B. From these secondary nodes, additional decisions or events will occur leading to the third level of nodes, until a conclusion is reached.

We can draw tree diagram through the rpart.plot function which is available in the rpart.plot library.



5.2.1 Assessment of the decision tree model

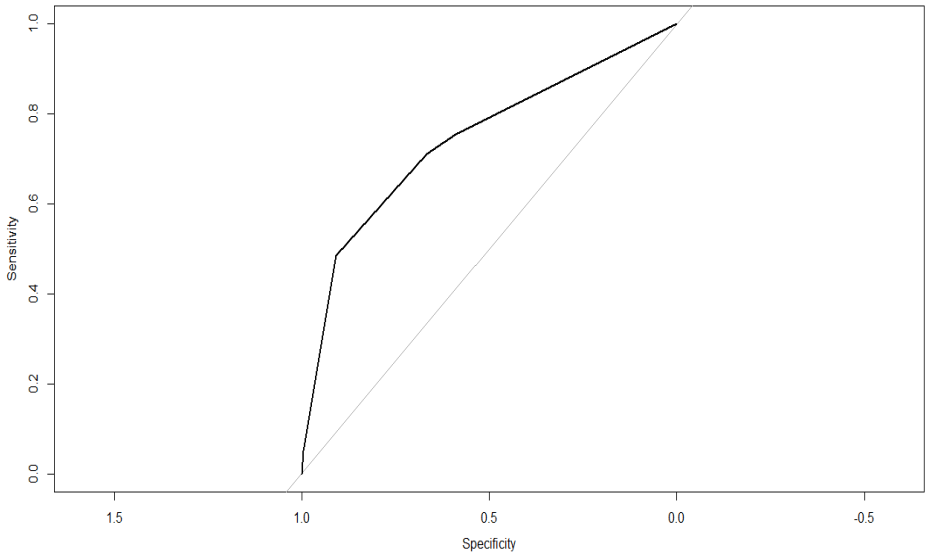
i) Confusion matrix

```
#Computing the accuracy by creating the confusion matrix:
ctree_predict <- predict(tree_model1,train.data)
p3 <- ifelse(ctree_predict > 0.5,1,0)
tab3 <- table(predicted = p3 , actual = train.data$Made.Donation.in.March.2007)
tab3
confusionMatrix(tab3)
```

```
> confusionMatrix(tab3)
Confusion Matrix and Statistics

         actual
predicted   0   1
        0 398  71
        1  40  67

               Accuracy : 0.8073
                 95% CI : (0.7727, 0.8387)
    No Information Rate : 0.7604
    P-Value [Acc > NIR] : 0.004136

                  Kappa : 0.427
 Mcnemar's Test P-Value : 0.004407

            Sensitivity : 0.9087
            Specificity : 0.4855
         Pos Pred Value : 0.8486
         Neg Pred Value : 0.6262
             Prevalence : 0.7604
         Detection Rate : 0.6910
   Detection Prevalence : 0.8142
      Balanced Accuracy : 0.6971

       'Positive' Class : 0
```

We can see accuarcy of the decision tree model is 80.7%.

ii) ROC curve:



6 Discussion and Recommendation

We have applied the above mentioned 3 classification techniques to the data and then tested the models. Later, these models were used for test data and the obtained output in each of the cases is stored in a .csv file. This output file is like the submission format mentioned in the project competition website.

Each of those output data thus obtained is uploaded to the competition website. In this competition, the evaluating parameter is the log-loss value which is given by the formula as below:

$$\text{Log loss} = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right]$$

The metric used for this competition is logarithmic loss. y^y^ is the probability that y=1y=1. Logarithmic loss provides a steep penalty for predictions that are both confident and wrong. The goal is to minimize the log loss.

Out of the models used by us for the prediction, we see that the **Decision Tree classifier model** performs better of all and resulted in better rank on the competition website when the output .csv file is uploaded.

Model	Accuracy
Linear regression - model 1	76.22%
Linear regression - model 2	77.08%
Decision Tree	**80.73%**

```
#predict the test data:
Made.Donation.in.March.2007 <- predict(ctree_model1, test.data)
final_data <- cbind(test.data$X,Made.Donation.in.March.2007)
write.csv(final_data,file="Blood.Donation.csv",row.names = FALSE)
```

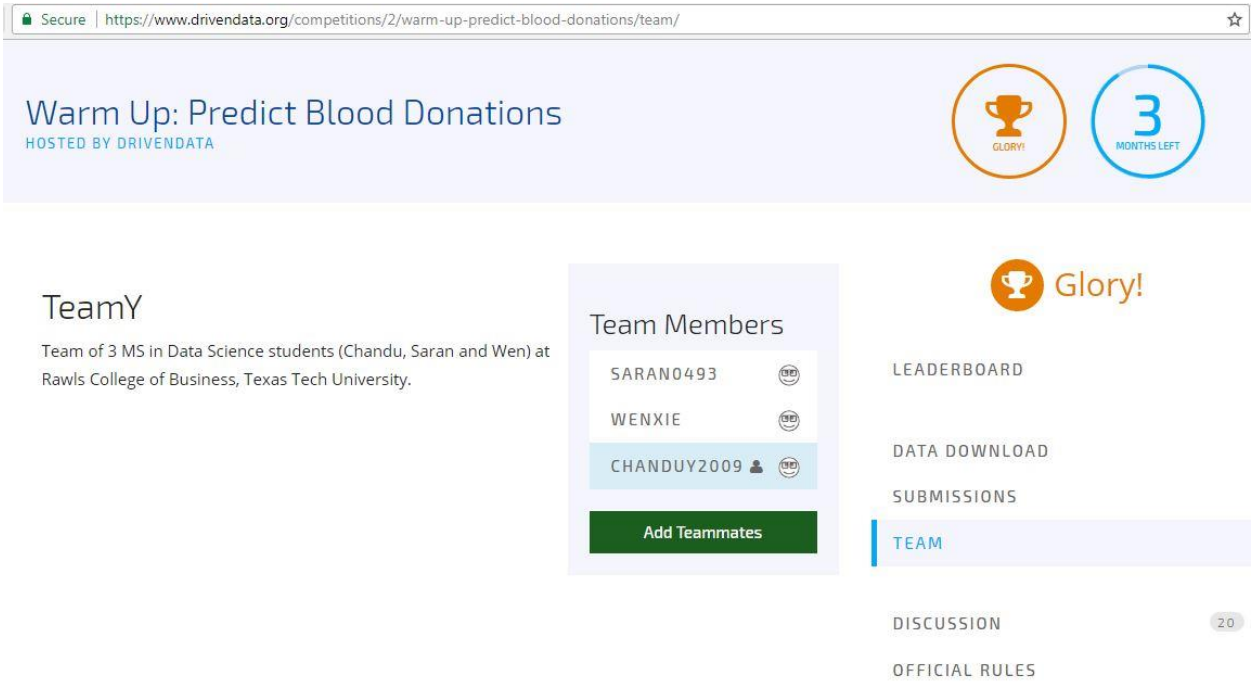
Sample Test prediction data:

Donor ID	Made.Donation.in.March.2007
659	0.15
276	0.118055556
263	0.226277372
303	0.118055556
83	0.612244898
500	0.612244898
530	0.226277372
244	0.118055556
249	0.118055556
728	0.118055556
129	0.118055556
534	0.118055556
317	0.15
401	0.226277372
696	0.226277372
192	0.118055556
176	0.118055556
571	0.118055556
139	0.118055556
423	0.226277372
563	0.612244898
56	0.612244898

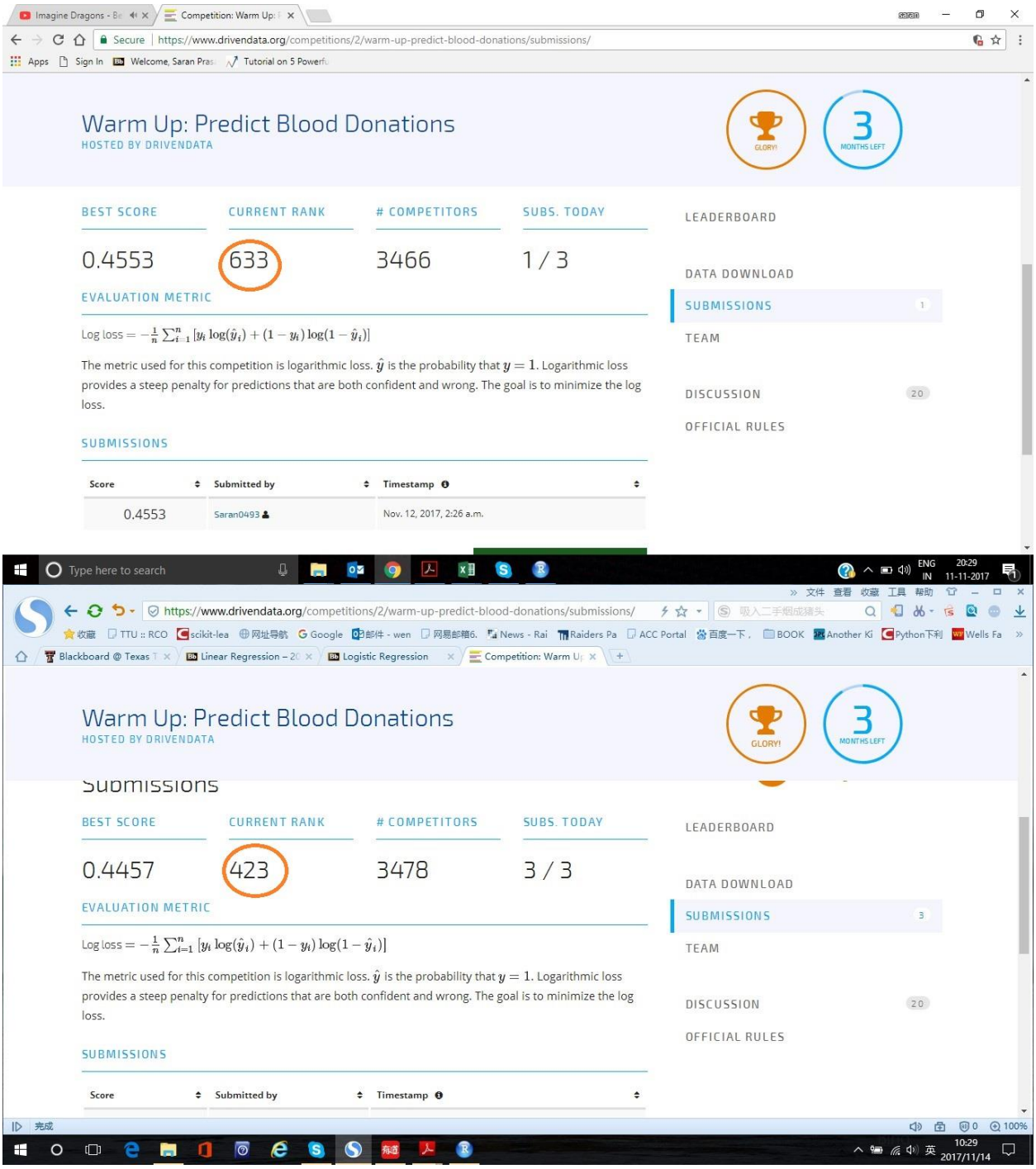
Thus, based on our observations and predictions for the data, we recommend the Decision tree classifier model to this data to better predict if a person will donate blood or not.

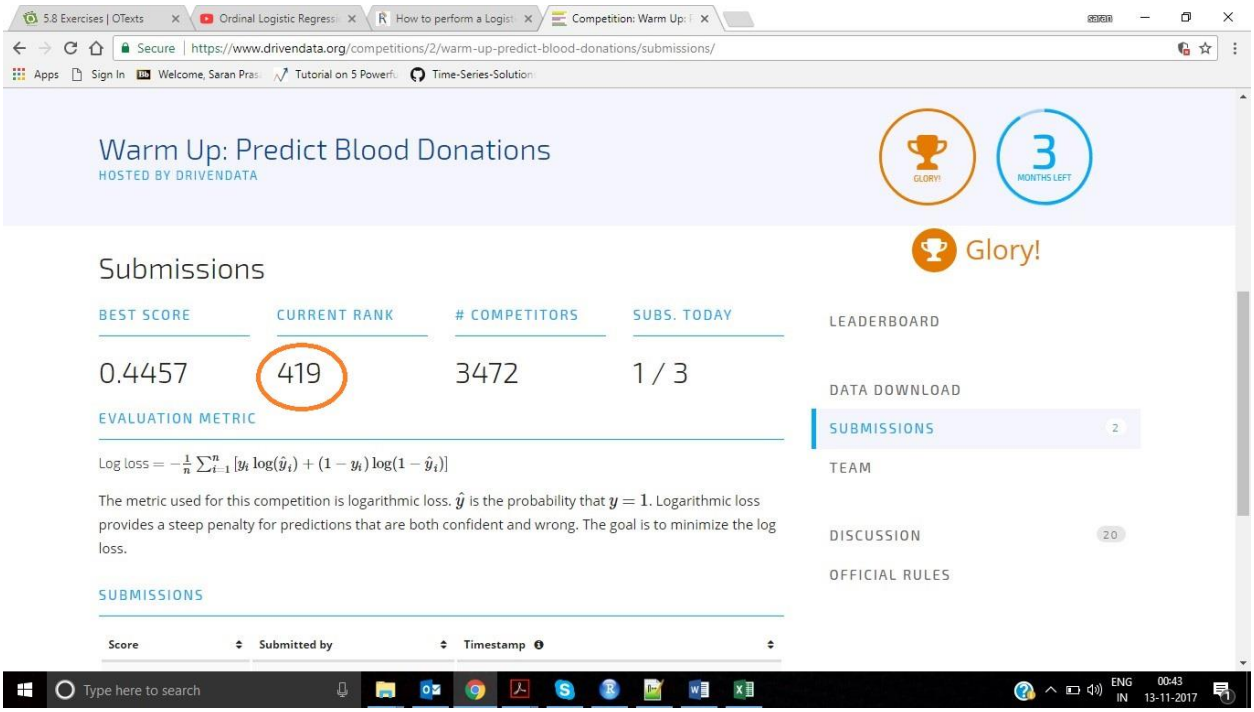
7 Submission to Competition Website

As suggested, we have registered as **TeamY** in this project competition on drivendata.org:



We have submitted the results to the competition website and below are the different ranks that we obtained for each of our models:


Warm Up: Predict Blood Donations
HOSTED BY DRIVENDATA
GLORY!
3
MONTHS LEFT
BEST SCORE
CURRENT RANK
COMPETITORS
SUBS. TODAY
0.4553
633
3466
1 / 3
EVALUATION METRIC
Log loss $= -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)]$
The metric used for this competition is logarithmic loss. $\hat{y}$ is the probability that $y = 1$. Logarithmic loss provides a steep penalty for predictions that are both confident and wrong. The goal is to minimize the log loss.
SUBMISSIONS
Score
Submitted by
Timestamp
0.4553
Saran0493
Nov. 12, 2017, 2:26 a.m.
LEADERBOARD
DATA DOWNLOAD
SUBMISSIONS
TEAM
DISCUSSION
OFFICIAL RULES
Warm Up: Predict Blood Donations
HOSTED BY DRIVENDATA
GLORY!
3
MONTHS LEFT
BEST SCORE
CURRENT RANK
COMPETITORS
SUBS. TODAY
0.4457
423
3478
3 / 3
EVALUATION METRIC
Log loss $= -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)]$
The metric used for this competition is logarithmic loss. $\hat{y}$ is the probability that $y = 1$. Logarithmic loss provides a steep penalty for predictions that are both confident and wrong. The goal is to minimize the log loss.
SUBMISSIONS
Score
Submitted by
Timestamp
LEADERBOARD
DATA DOWNLOAD
SUBMISSIONS
TEAM
DISCUSSION
OFFICIAL RULES



The best rank we have obtained in our submissions is 419 out of 3472 participants.