# Final Report of Internship Program 2021

*On*

# *“Predicting blood donations”*

## MEDTOUREASY, NEW DELHI

30th January 2021

# ACKNOWLEDGMENTS

The internship opportunity that I had with MedTourEasy was a great change for learning and understanding the intricacies of the subject of Data Visualizations in Data Analysis; and also, for personal as well as professional development. I am very obliged for having a chance to interact with so many professionals who guided me throughout the internship project and made it a great learning curve for me.

Firstly, I express my deepest gratitude and special thanks to the Training Head of MedTourEasy, Mr. Ankit Hasija who gave me an opportunity to carry out my internship at their esteemed organization. Also, I express my thanks to him for making me understand the details of the Data Scientist profile and training me in the same so that I can carry out the project properly and with maximum client satisfaction and also for sparing his valuable time in spite of his busy schedule.

I would also like to thank the team of MedTourEasy and my colleagues who made the working environment productive and very conducive.

# TABLE OF CONTENTS

# ABSTRACT

Blood transfusion saves lives - from replacing lost blood during major surgery or a serious injury to treating various illnesses and blood disorders. Ensuring that there's enough blood in supply whenever needed is a serious challenge for the health professionals. According to WebMD, "about 5 million Americans need a blood transfusion every year". 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.

## 1.1 About the Company

MedTourEasy, a global healthcare company, provides you the informational resources needed to evaluate your global options. It helps you find the right healthcare solution based on specific health needs, affordable care while meeting the quality standards that you expect to have in healthcare. MedTourEasy improves access to healthcare for people everywhere. It is an easy to use platform and service that helps patients to get medical second opinions and to schedule affordable, high-quality medical treatment abroad.

## 1.2 About the Project

Blood transfusion saves lives - from replacing lost blood during major surgery or a serious injury to treating various illnesses and blood disorders. Ensuring that there's enough blood in supply whenever needed is a serious challenge for the health professionals. According to WebMD, "about 5 million Americans need a blood transfusion every year". 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. The data is stored in datasets/transfusion.data and it is structured according to the RFMTC marketing model (a variation of RFM)
The above section has been represented in the ipython notebook.

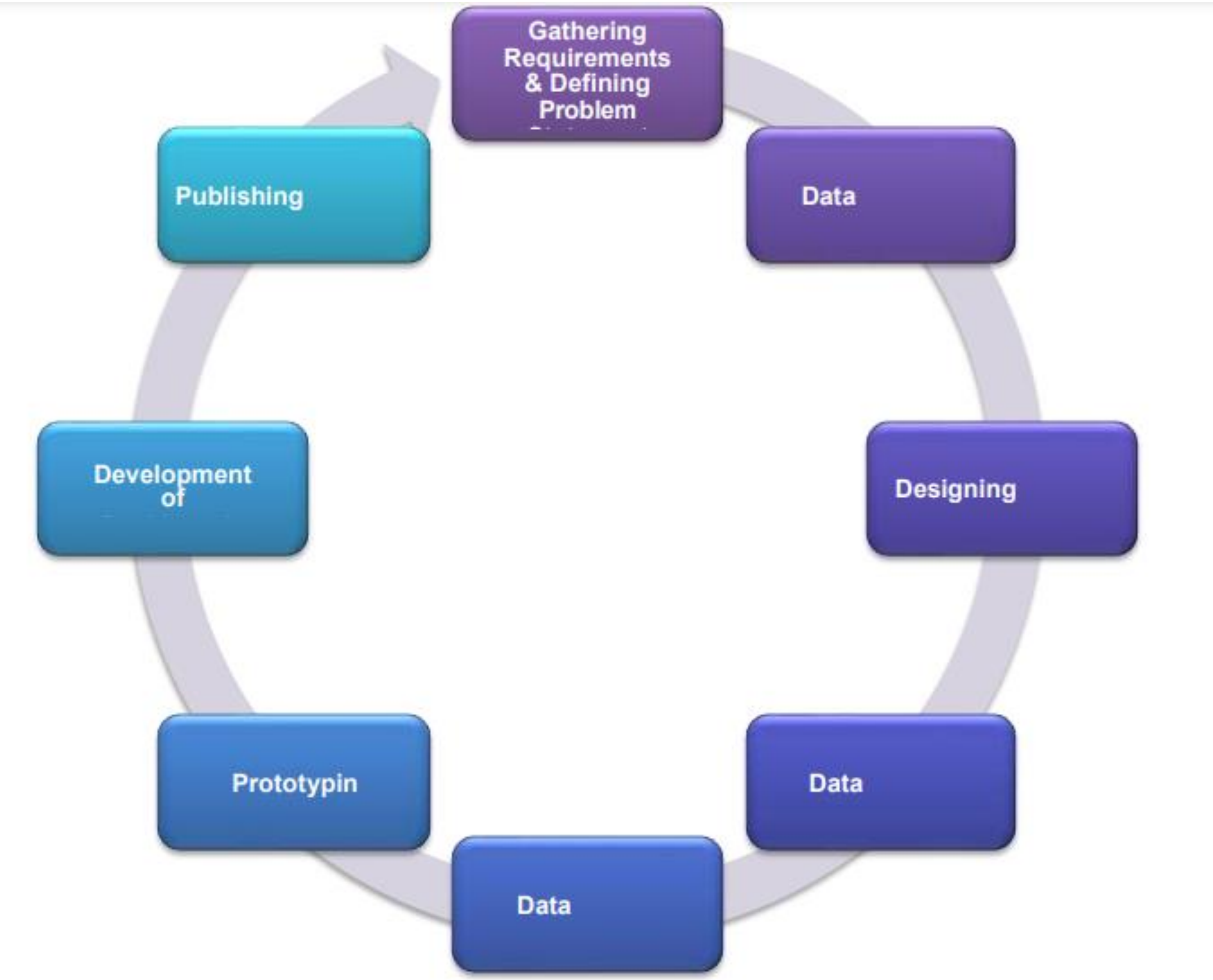
## 1.3 Objectives

This project focuses drawing model which predicts whether donor will be able to give blood next time vehicle come

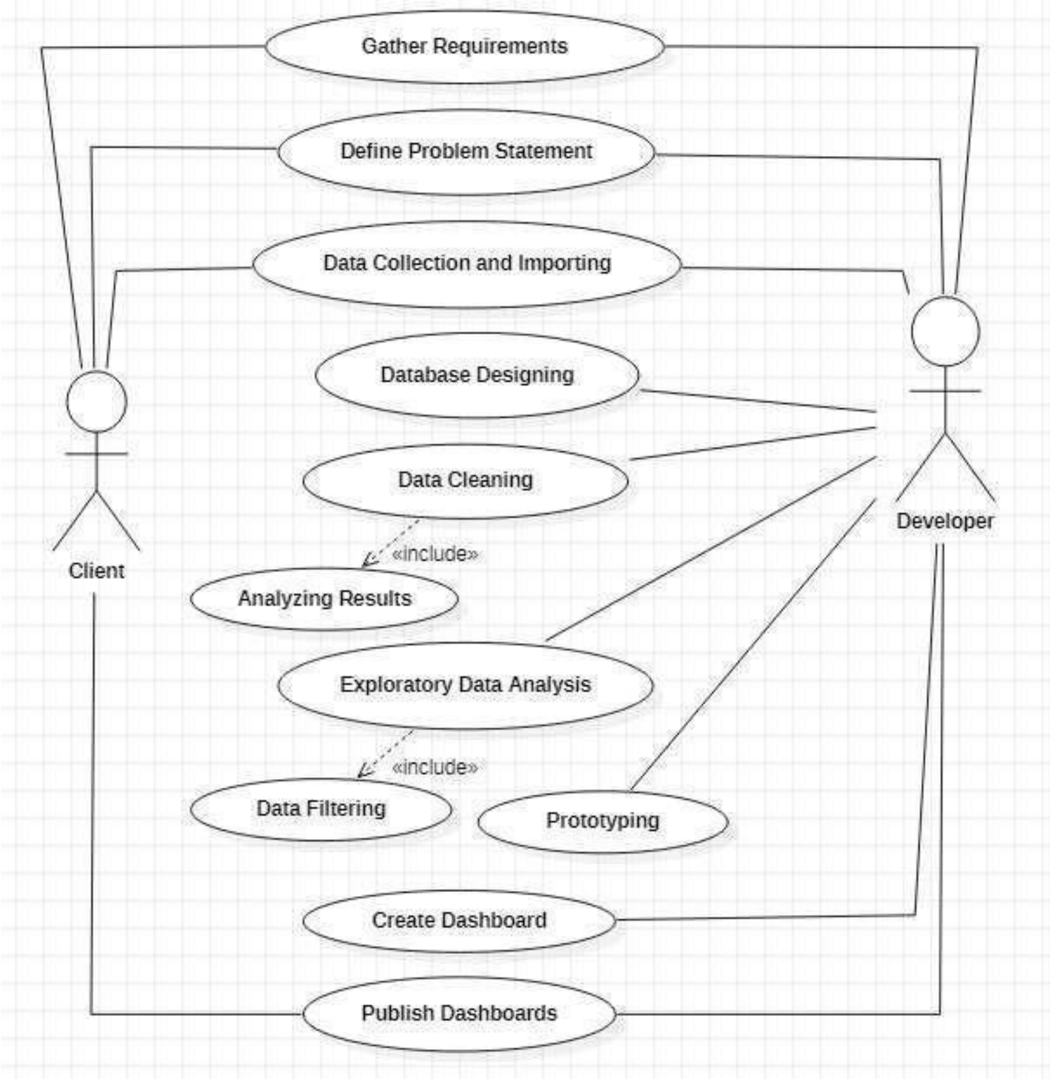
# I. METHODOLOGY

## 2.1 Flow of the Project

The project followed the following steps to accomplish the desired objectives and deliverables. Each step has been explained in detail in the following section.



## 2.2 Use Case Diagram



Above figure shows the use case of the project. There are two main actors in the same: The Client and Developer. The developer will first gather requirements and define the problem statement then collecting the required data and importing it. Then the developer will design databases so as to identify various constraints and relations in the data. Next step is to clean the data to remove irregular values, blank values etc. Next, exploratory data analysis is conducted to filter the data according to the requirements of the project. Then a prototype of the dashboards is created using PowerBI to get a clear view of the visualizations to be developed. Finally, a dashboard is developed and analyzed to publish the results to the client.

## 2.3 Language and Platform Used

### 2.3.1 Language: Python

Python is an interpreted, high-level and general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.. The important features of Python are:

Free and Open Source

Object-Oriented Language

GUI Programming Support

High-Level Language

Extensible feature

Python is Portable language

Python is Integrated language

□

### 2.3.2 IDE: Jupyter Notebook

The IPython Notebook is now known as the Jupyter Notebook. It is an interactive computational environment, in which you can combine code execution, rich text, mathematics, plots and rich media. Major features are:

Interactive shells (terminal and Qt-based).

A browser-based notebook interface with support for code, text, mathematical expressions, inline plots and other media.

Support for interactive data visualization and use of GUI toolkits.

Flexible, embeddable interpreters to load into one's own projects.

Tools for parallel computing.

# II. IMPLEMENTATION

## 1. Gathering Requirements and Defining Problem Statement

This is the first step wherein the requirements are collected from the clients to understand the deliverables and goals to be achieved after which a problem statement is defined which has to be adhered to while development of the project.

## 2. Inspecting transfusion.data file

Blood transfusion saves lives - from replacing lost blood during major surgery or a serious injury to treating various illnesses and blood disorders. Ensuring that there's enough blood in supply whenever needed is a serious challenge for the health professionals. According to WebMD, "about 5 million Americans need a blood transfusion every year". 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. The data is stored in datasets/transfusion.data and it is structured according to the RFMTC marketing model (a variation of RFM). We'll explore what that means later in this notebook. First, let's inspect the data.

## 3. Loading the blood donations data

We now know that we are working with a typical CSV file (i.e., the delimiter is ,, etc.). We proceed to loading the data into memory.

## 4. Inspecting transfusion DataFrame

Let's briefly return to our discussion of RFM model. RFM stands for Recency, Frequency and Monetary Value and it is commonly used in marketing for identifying your best customers. In our case, our customers are blood donors.

RFMTC is a variation of the RFM model. Below is a description of what each column means in our dataset:

1. R (Recency - months since the last donation)
2. F (Frequency - total number of donation)
3. M (Monetary - total blood donated in c.c.)
4. T (Time - months since the first donation)
5. a binary variable representing whether he/she donated blood in March 2007 (1 stands for donating blood; 0 stands for not donating blood)
6. It looks like every column in our DataFrame has the numeric type, which is exactly what we want when building a machine learning model. Let's verify our hypothesis.

## 5. Creating target column

We are aiming to predict the value in whether he/she donated blood in the March 2007 column. Let's rename this to target so that it's more convenient to work with.

## 6. Checking target incidence

We want to predict whether or not the same donor will give blood the next time the vehicle comes to campus. The model for this is a binary classifier, meaning that there are only 2 possible outcomes:

1. 0 - the donor will not give blood
2. 1 - the donor will give blood
3. Target incidence is defined as the number of cases of each individual target value in a dataset. That is, how many 0s in the target column compared to how many 1s? Target incidence gives us an idea of how balanced (or imbalanced) is our dataset

## 7. Splitting transfusion into train and test datasets

We'll now use train_test_split() method to split transfusion DataFrame.

Target incidence informed us that in our dataset 0s appear 76% of the time. We want to keep the same structure in train and test datasets, i.e., both datasets must have 0 target incidence of 76%. This is very easy to do using the train_test_split() method from the scikit learn library - all we need to do is specify the stratify parameter. In our case, we'll stratify on the target column.customization.

## 8. Selecting model using TPOT

TPOT is a Python Automated Machine Learning tool that optimizes machine learning pipelines using genetic programming.TPOT will automatically explore hundreds of possible pipelines to find the best one for our dataset. Note, the outcome of this search will be a scikit-learn pipeline, meaning it will include any pre-processing steps as well as the model.We are using TPOT to help us zero in on one model that we can then explore and optimize further.

## 9. Checking the variance

TPOT picked LogisticRegression as the best model for our dataset with no pre-processing steps, giving us the AUC score of 0.7850. This is a great starting point. Let's see if we can make it better. One of the assumptions for linear regression models is that the data and the features we are giving it are related in a linear fashion, or can be measured with a linear distance metric. If a feature in our dataset has a high variance that's an order of magnitude or more greater than the other features, this could impact the model's ability to learn from other features in the dataset. Correcting for high variance is called normalization. It is one of the possible transformations you do before training a model. Let's check the variance to see if such transformation is needed.

## 10. Log normalization

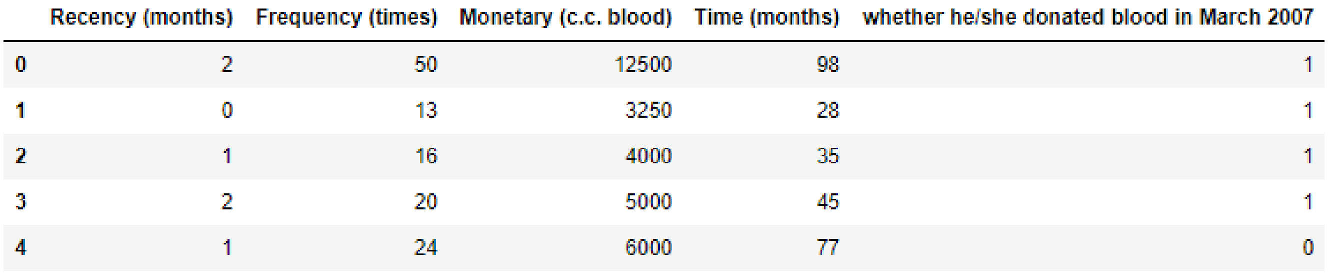
Monetary (c.c. blood)'s variance is very high in comparison to any other column in the dataset. This means that, unless accounted for, this feature may get more weight by the model (i.e., be seen as more important) than any other feature. One way to correct for high variance is to use log normalization.

## 11. Training the linear regression model

The variance looks much better now. Notice that now Time (months) has the largest variance, but it's not the orders of magnitude higher than the rest of the variables, so we'll leave it as is. We are now ready to train the linear regression model.

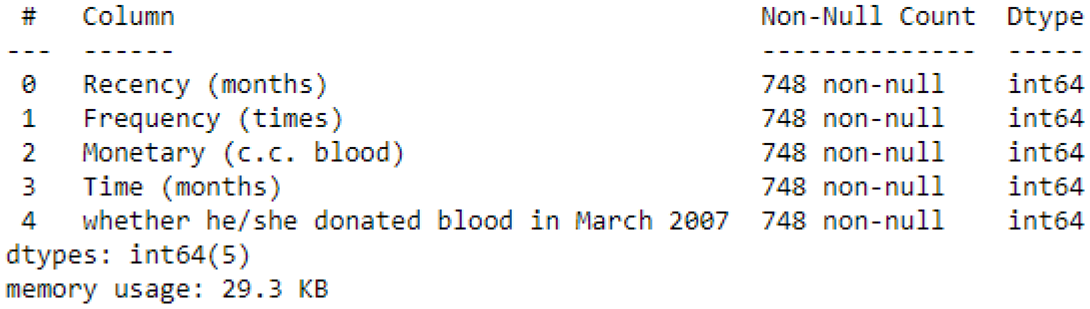
# III. SAMPLE SCREENSHOTS

### Dataset:



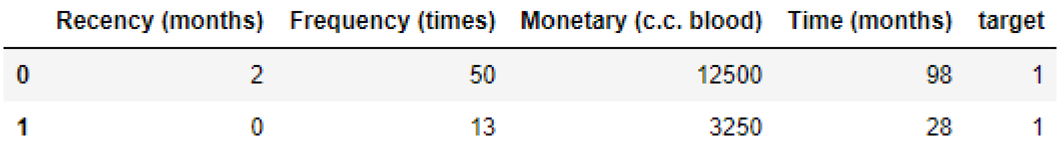
| | Recency (months) | Frequency (times) | Monetary (c.c. blood) | Time (months) | whether he/she donated blood in March 2007 |
|---|---|---|---|---|---|
| 0 | 2 | 50 | 12500 | 98 | 1 |
| 1 | 0 | 13 | 3250 | 28 | 1 |
| 2 | 1 | 16 | 4000 | 35 | 1 |
| 3 | 2 | 20 | 5000 | 45 | 1 |
| 4 | 1 | 24 | 6000 | 77 | 0 |

### Dataset Info:



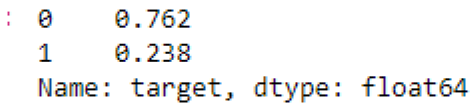
```
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Recency (months)                            748 non-null    int64
 1   Frequency (times)                           748 non-null    int64
 2   Monetary (c.c. blood)                       748 non-null    int64
 3   Time (months)                               748 non-null    int64
 4   whether he/she donated blood in March 2007  748 non-null    int64
dtypes: int64(5)
memory usage: 29.3 KB
```

## Adding Target column



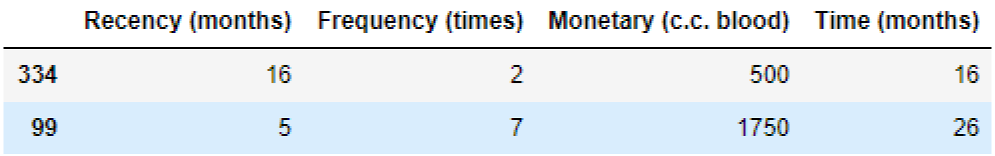
| | Recency (months) | Frequency (times) | Monetary (c.c. blood) | Time (months) | target |
|---|---|---|---|---|---|
| **0** | 2 | 50 | 12500 | 98 | 1 |
| **1** | 0 | 13 | 3250 | 28 | 1 |

## Target incidence



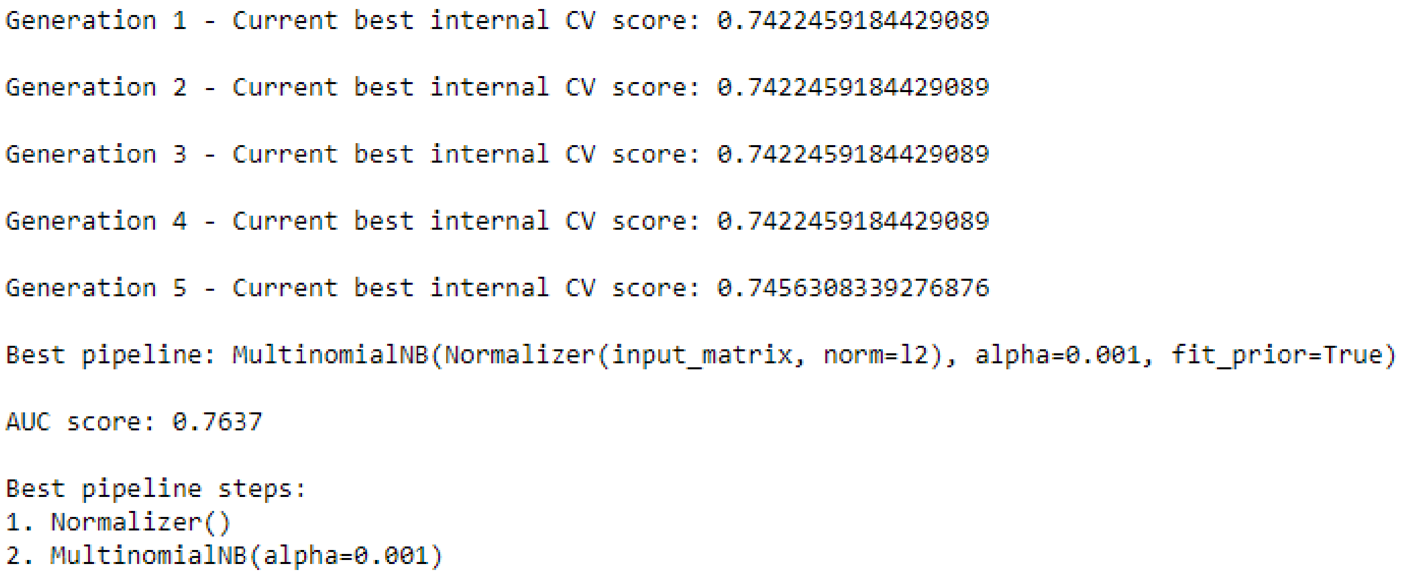
```
0    0.762
1    0.238
Name: target, dtype: float64
```

## Training dataset



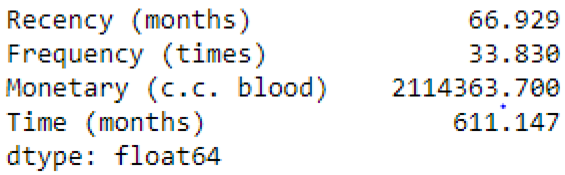
| | Recency (months) | Frequency (times) | Monetary (c.c. blood) | Time (months) |
|---|---|---|---|---|
| 334 | 16 | 2 | 500 | 16 |
| 99 | 5 | 7 | 1750 | 26 |

## Selecting model using tpot



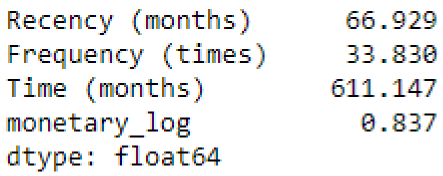
```
Generation 1 - Current best internal CV score: 0.7422459184429089

Generation 2 - Current best internal CV score: 0.7422459184429089

Generation 3 - Current best internal CV score: 0.7422459184429089

Generation 4 - Current best internal CV score: 0.7422459184429089

Generation 5 - Current best internal CV score: 0.7456308339276876

Best pipeline: MultinomialNB(Normalizer(input_matrix, norm=l2), alpha=0.001, fit_prior=True)

AUC score: 0.7637

Best pipeline steps:
1. Normalizer()
2. MultinomialNB(alpha=0.001)
```

## Variance of Training set



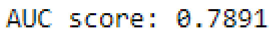
```
Recency (months)              66.929
Frequency (times)             33.830
Monetary (c.c. blood)    2114363.700
Time (months)                611.147
dtype: float64
```

## Log normalization to correct high variance



```
Recency (months)      66.929
Frequency (times)     33.830
Time (months)        611.147
monetary_log           0.837
dtype: float64
```

## Auc Score



```
AUC score: 0.7891
```

## IV. CONCLUSION AND FUTURE SCOPE

The demand for blood fluctuates throughout the year. As one prominent example, blood donations slow down during busy holiday seasons. An accurate forecast for the future supply of blood allows for an appropriate action to be taken ahead of time and therefore saving more lives.

In this notebook, we explored automatic model selection using TPOT and AUC score we got was 0.7850. This is better than simply choosing 0 all the time (the target incidence suggests that such a model would have 76% success rate). We then normalized our training data and improved the AUC score by 0.5%. In the field of machine learning, even small improvements in accuracy can be important, depending on the purpose.

Another benefit of using a logistic regression model is that it is interpretable. We can analyze how much of the variance in the response variable (target) can be explained by other variables in our dataset

## V. REFERENCES

**Data Collection:**

**https://drive.google.com/file/d/1S2o3wEAfEPha06ECh6kirwUijcCq54nY/view**

**Code Link:**

**https://github.com/harshsomaiya18/MedTourEasy/blob/main/Data%20Analyst%20MTE%20Project.ipynb**