Open IIT
Data Analytics

Team Name: **XBoosters**

Contents

Annexure Content

Main Content

Annexure

A) Introduction to Customer Lifetime Value (CLV)

1. Theory

Customer lifetime value helps you predict future revenue and measure long-term business success. CLV tells you how much profit your company can expect from a typical client throughout the relationship. More to the point, CLV helps you estimate how much you should invest to retain a customer.

For example, if you've bought a $40 Christmas tree from the same grower for the last ten years, your CLV has been worth $400 to them. But as you can imagine, in bigger companies, CLV gets more complicated to calculate.

Q-*Why Is CLV Important to Your Business?*
Ultimately, you don't need to get bogged down in complex calculations – you just need to be mindful of the value that a customer provides over their lifetime relationship with you. By understanding the customer experience and measuring feedback at all key touchpoints, you can start to understand the critical drivers of CLV.

CLV is a great metric to track and optimize, but one thing to keep a close eye on too is the cost of that customer to your business. If the cost of serving an existing customer becomes too high, you may be making a loss despite their seemingly high CLV.
So there's a balancing act to negotiate here.

2. Calculation

CLV (customer lifetime value) calculation process consists of four steps:

1. forecasting of remaining customer lifetime (most often in years)
2. forecasting of future revenues (most often year-by-year), based on estimation about future products purchased and price paid
3. estimation of costs for delivering those products
4. calculation of the net present value of these future amounts

Factors Affecting the model:

Discount rate, the cost of capital used to discount future revenue from a customer. The current interest rate is sometimes used as a simple (but incorrect) proxy for a discount rate.

Retention cost, the amount of money a company has to spend in a given period to retain an existing customer. Retention costs include customer support, billing, promotional incentives, etc.

Gross Contribution(GC) per customer, in simple words it is the *Monthly premium* paid by the customer multiplied by 12.

Retention Rate, the probability of a customer to retain the insurance after the expiry date of the insurance.

So, we finally arrive at the following mathematical expression:

$$\mathrm{CLV} = \mathrm{GC} \cdot \sum_{i=1}^{n} \frac{r^{i}}{(1+d)^{i}} - \mathrm{M} \cdot \sum_{i=1}^{n} \frac{r^{i-1}}{(1+d)^{i-0.5}}$$

GC = yearly gross contribution per customer
M = Retention cost per customer per year
r = retention rate
N = number of years
d = discount rate

Assumptions:-

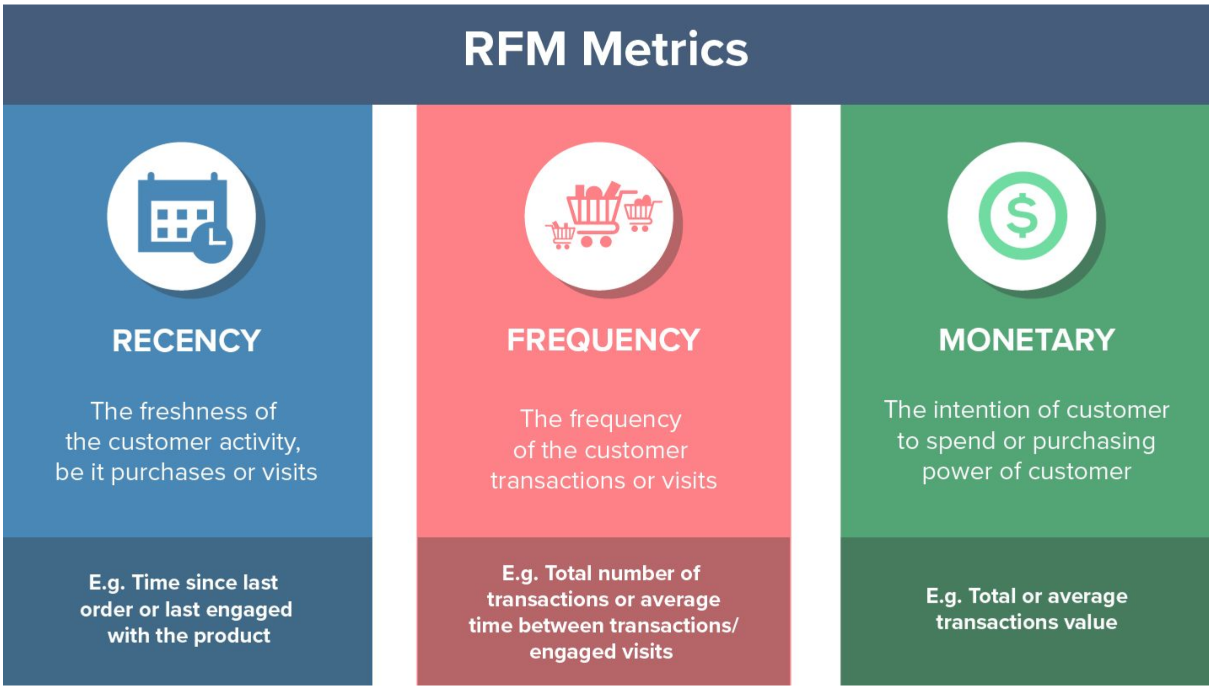
- *this formula assumes the retention activities are paid for each mid-year and they only affect those who were retained in the previous year.*

It is often helpful to estimate customer lifetime value with a simple model to make initial assessments of customer segments and targeting. If GC is found to be relatively fixed across periods, CLV can be expressed as a simpler model assuming an infinite economic life.

$$\mathrm{CLV} = \mathrm{GC} \cdot \left(\frac{r}{1 + d - r} \right)$$

B) Introduction to RFM Analysis (Recency, Frequency, Monetary)

RFM (Recency, Frequency, Monetary) is a very Simple Technique that we can apply it very easy and get the super useful analysis for our *Customer Segmentation*



RFM Metrics

RECENCY

The freshness of the customer activity, be it purchases or visits

E.g. Time since last order or last engaged with the product

FREQUENCY

The frequency of the customer transactions or visits

E.g. Total number of transactions or average time between transactions/ engaged visits

MONETARY

The intention of customer to spend or purchasing power of customer

E.g. Total or average transactions value

Recency (R) as days since last purchase: How many days ago was their last purchase? Deduct most recent purchase date from today to calculate the recency value. 1 day ago? 14 days ago? 500 days ago?

Frequency (F) as total number of transactions: How many times has the customer purchased from our store? For example, if someone placed 10 orders over a period of time, their frequency is 10.

Monetary (M) as total money spent: How many $$ (or whatever is your currency of calculation) has this customer spent? Simply total up the money from all transactions to get the M value.

Q- What is customer Segmentation?

It is a practice of dividing customer base into groups of individuals that are similar in their specific ways. Here, we will divide them on the basis of their *RFM score.*

Q- How to calculate RFM score?

First of all, we will calculate the respective recency, frequency and monetary score for a particular score. Adding them, we get the RFM score.

RFM score = Recency score + Frequency score + monetary score

Higher the RFM score shows that particular customer is more profitable for the firm.

RFM analysis readily answers these questions for your business!

- Who are my best customers?
- Which customers are at the verge of withdrawal?
- Who has the potential to be converted in more profitable customers?
- Who are lost customers that you don't need to pay much attention to?
- Which customers you must retain?
- Who are your loyal customers?
- Which group of customers is most likely to respond to your current campaign?

Hold on! We will give our final verdict in a while!

Problem Statement

Predict Customer Life-time Value for an Auto Insurance Company

Objective:

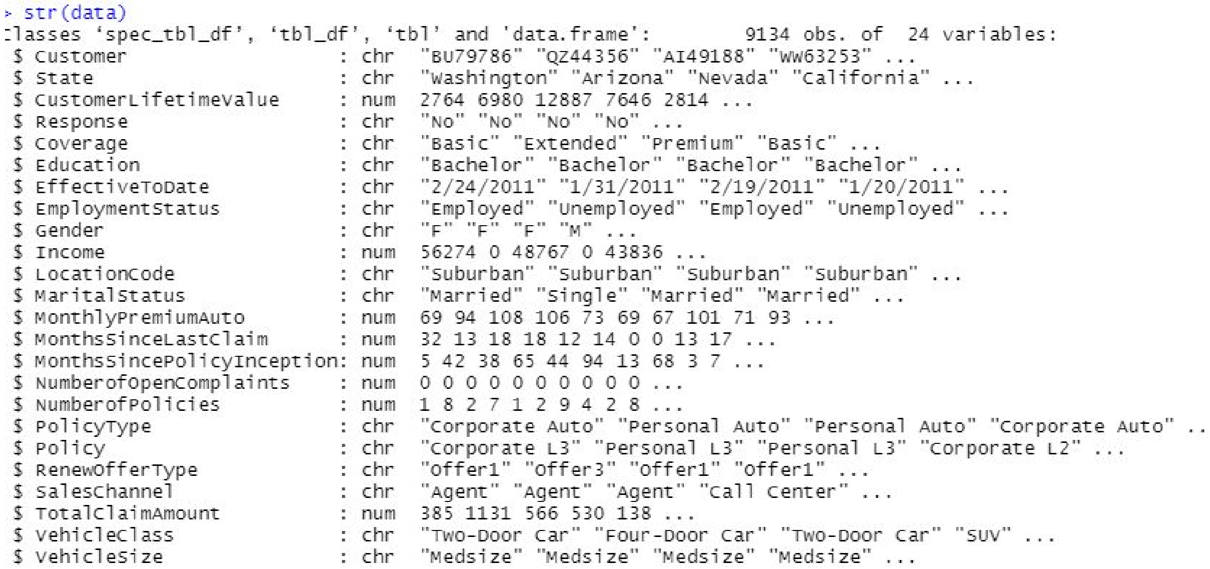
1) For an Auto Insurance company, predict the customer lifetime value (CLV). CLV is the total revenue the client will derive from their entire relationship with a customer. Because we don't know how long each customer relationship will be, we make a good estimate and state CLV as a periodic value — that is, we usually say "this customer's 12-month (or 24-month, etc) CLV is $x".

2) The client also wants to know the types of customers that would generally give us more revenue.

Introduction and exploratory data analysis

Understanding the data set:

Structure of data is shown below



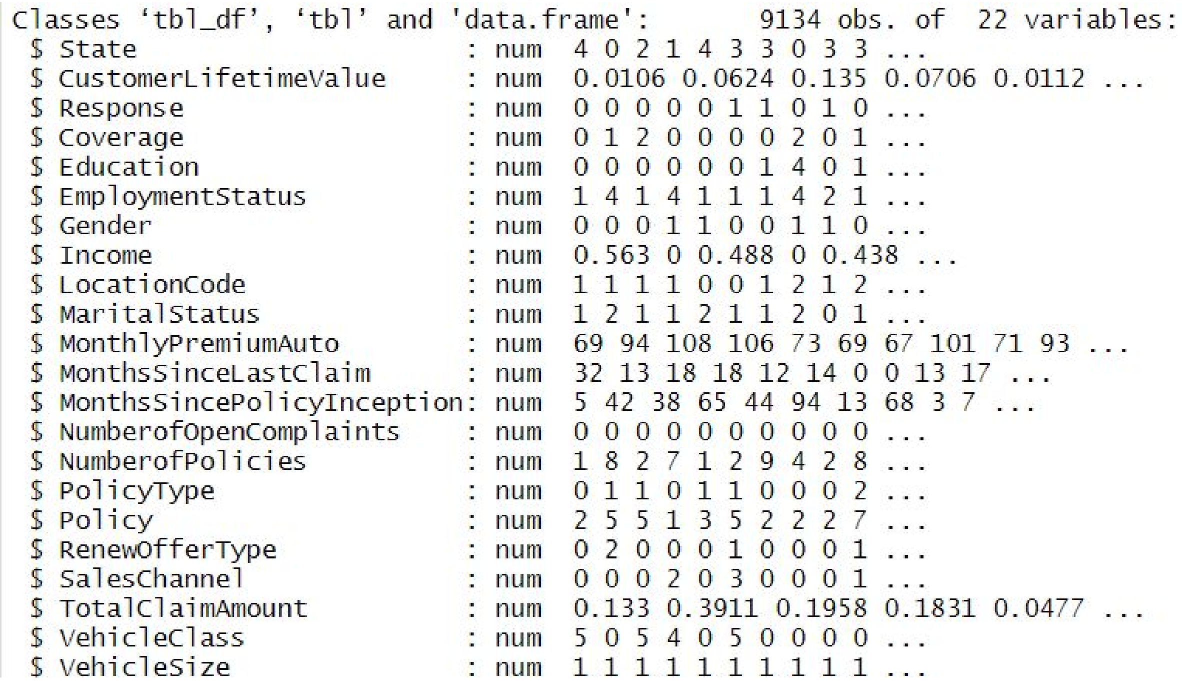
```
> str(data)
:lasses ‘spec_tbl_df’, ‘tbl_df’, ‘tbl’ and 'data.frame':       9134 obs. of  24 variables:
 $ Customer                 : chr  "BU79786" "QZ44356" "AI49188" "WW63253" ...
 $ State                    : chr  "Washington" "Arizona" "Nevada" "California" ...
 $ CustomerLifetimeValue    : num  2764 6980 12887 7646 2814 ...
 $ Response                 : chr  "No" "No" "No" "No" ...
 $ Coverage                 : chr  "Basic" "Extended" "Premium" "Basic" ...
 $ Education                : chr  "Bachelor" "Bachelor" "Bachelor" "Bachelor" ...
 $ EffectiveToDate          : chr  "2/24/2011" "1/31/2011" "2/19/2011" "1/20/2011" ...
 $ EmploymentStatus         : chr  "Employed" "Unemployed" "Employed" "Unemployed" ...
 $ Gender                   : chr  "F" "F" "F" "M" ...
 $ Income                   : num  56274 0 48767 0 43836 ...
 $ LocationCode             : chr  "Suburban" "Suburban" "Suburban" "Suburban" ...
 $ MaritalStatus            : chr  "Married" "Single" "Married" "Married" ...
 $ MonthlyPremiumAuto       : num  69 94 108 106 73 69 67 101 71 93 ...
 $ MonthsSinceLastClaim     : num  32 13 18 18 12 14 0 0 13 17 ...
 $ MonthsSincePolicyInception: num  5 42 38 65 44 94 13 68 3 7 ...
 $ NumberofOpenComplaints   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ NumberofPolicies         : num  1 8 2 7 1 2 9 4 2 8 ...
 $ PolicyType               : chr  "Corporate Auto" "Personal Auto" "Personal Auto" "Corporate Auto" ..
 $ Policy                   : chr  "Corporate L3" "Personal L3" "Personal L3" "Corporate L2" ...
 $ RenewOfferType           : chr  "Offer1" "Offer3" "Offer1" "Offer1" ...
 $ SalesChannel             : chr  "Agent" "Agent" "Agent" "Call Center" ...
 $ TotalClaimAmount         : num  385 1131 566 530 138 ...
 $ VehicleClass             : chr  "Two-Door Car" "Four-Door Car" "Two-Door Car" "SUV" ...
 $ VehicleSize              : chr  "Medsize" "Medsize" "Medsize" "Medsize" ...
```

Cleaning the data set:

1. For creating different statistical models, we need to remove some variables:
   Since variable "Customer" has unique values it will have no contribution in making the model. Also since we don't know the initial date, effective to date becomes irrelevant. So we remove both of those.

2. We encode all the variables who are of “chr” type to a numeric type. Example: Variable Gender has only two values “M” and “F” suggesting male and female. We change it to a numeric type by assigning M to 0 and F and 1.
3. We normalize the CustomerLifetimeValue, Income and TotalClaimAmount by using:
   X = (X - min(X))/(max(X)-min(X))
   So that we get their values between 0 and 1.

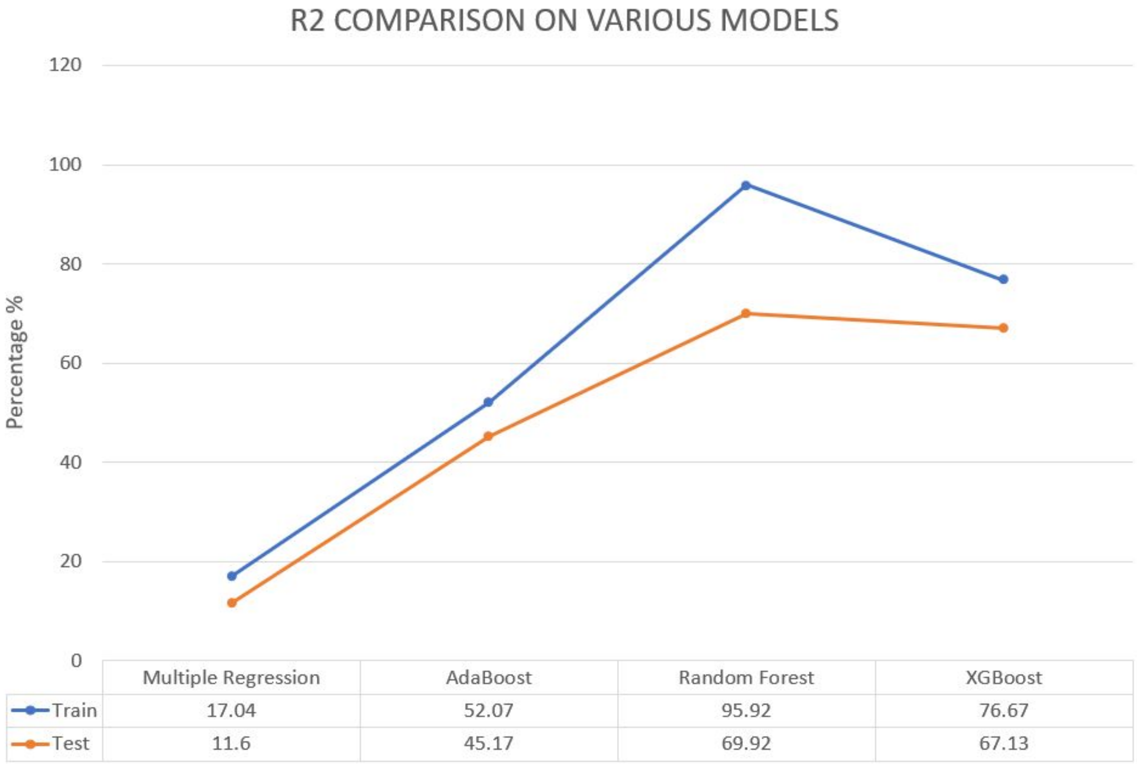
Structure of the data after the first time cleaning is as shown below:



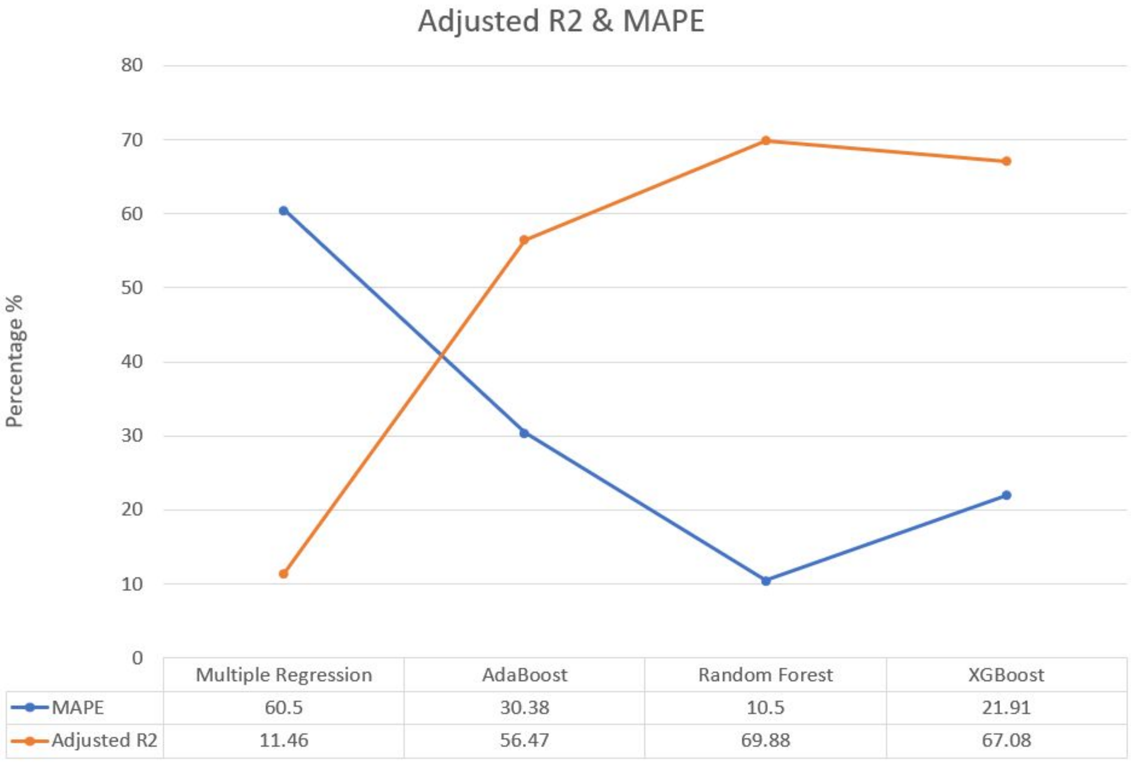
```
Classes ‘tbl_df’, ‘tbl’ and 'data.frame':	9134 obs. of  22 variables:
 $ State                     : num  4 0 2 1 4 3 3 0 3 3 ...
 $ CustomerLifetimeValue     : num  0.0106 0.0624 0.135 0.0706 0.0112 ...
 $ Response                  : num  0 0 0 0 0 1 1 0 1 0 ...
 $ Coverage                  : num  0 1 2 0 0 0 0 2 0 1 ...
 $ Education                 : num  0 0 0 0 0 0 1 4 0 1 ...
 $ EmploymentStatus          : num  1 4 1 4 1 1 1 4 2 1 ...
 $ Gender                    : num  0 0 0 1 1 0 0 1 1 0 ...
 $ Income                    : num  0.563 0 0.488 0 0.438 ...
 $ LocationCode              : num  1 1 1 1 0 0 1 2 1 2 ...
 $ MaritalStatus             : num  1 2 1 1 2 1 1 2 0 1 ...
 $ MonthlyPremiumAuto        : num  69 94 108 106 73 69 67 101 71 93 ...
 $ MonthsSinceLastClaim      : num  32 13 18 18 12 14 0 0 13 17 ...
 $ MonthsSincePolicyInception: num  5 42 38 65 44 94 13 68 3 7 ...
 $ NumberofOpenComplaints    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ NumberofPolicies          : num  1 8 2 7 1 2 9 4 2 8 ...
 $ PolicyType                : num  0 1 1 0 1 1 0 0 0 2 ...
 $ Policy                    : num  2 5 5 1 3 5 2 2 2 7 ...
 $ RenewOfferType            : num  0 2 0 0 0 1 0 0 0 1 ...
 $ SalesChannel              : num  0 0 0 2 0 3 0 0 0 1 ...
 $ TotalClaimAmount          : num  0.133 0.3911 0.1958 0.1831 0.0477 ...
 $ VehicleClass              : num  5 0 5 4 0 5 0 0 0 0 ...
 $ VehicleSize               : num  1 1 1 1 1 1 1 1 1 1 ...
```

Basic Models

1. **Multiple Linear Regression**
2. **AdaBoost Regression**
3. **RandomForest Regression**
4. **XGBoost Regression**



	Multiple Regression	AdaBoost	Random Forest	XGBoost
Train	17.04	52.07	95.92	76.67
Test	11.6	45.17	69.92	67.13



	Multiple Regression	AdaBoost	Random Forest	XGBoost
MAPE	60.5	30.38	10.5	21.91
Adjusted R2	11.46	56.47	69.88	67.08

Reasons for rejection of a model:

1. **Multiple Linear Regressor**: The r-square value, as well as the adjusted r-square value, is too low. Hence, we can infer that a linear model cannot be fitted in this case.

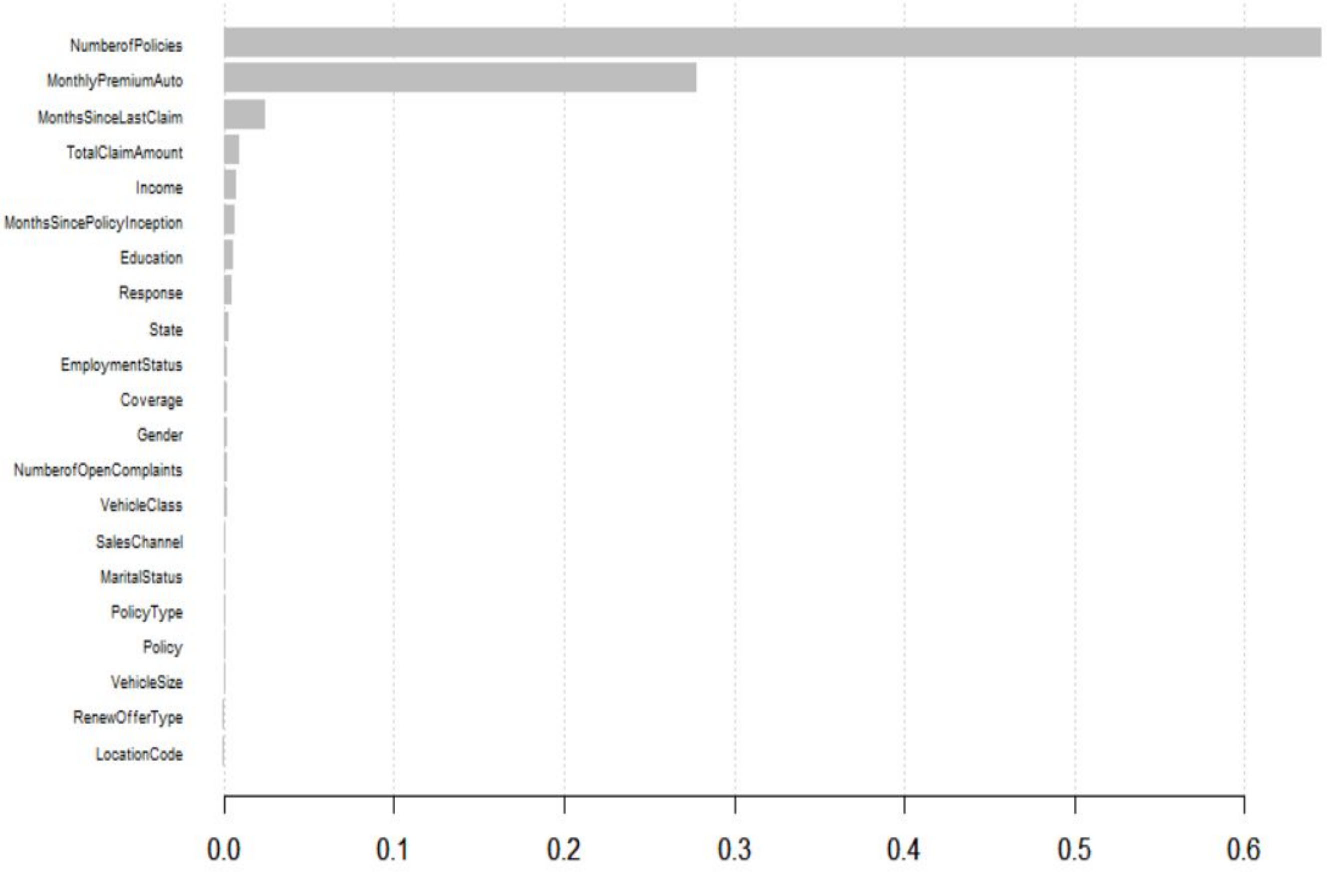
2. **AdaBoost Regressor**: Reason same as above.

3. **RandomForest Regressor**: As the r-square varying on my Train(Development) and Test(Validation) dataset a lot this is mainly due to Overfitting.

Reason for selecting XGBoost as our final model:

1. XGBoost was giving stable r-square value over the train and test dataset.

Checking the Variable Importance



It is clear from the above graph that
Variables such as SalesChannel, Marital Status, PolicyType, Policy, VehicleSize, RenewOfferType and Location Code has a nearly negligible effect on predicting CLV. So, we remove them from our model.

FINAL MODEL - XGBOOST

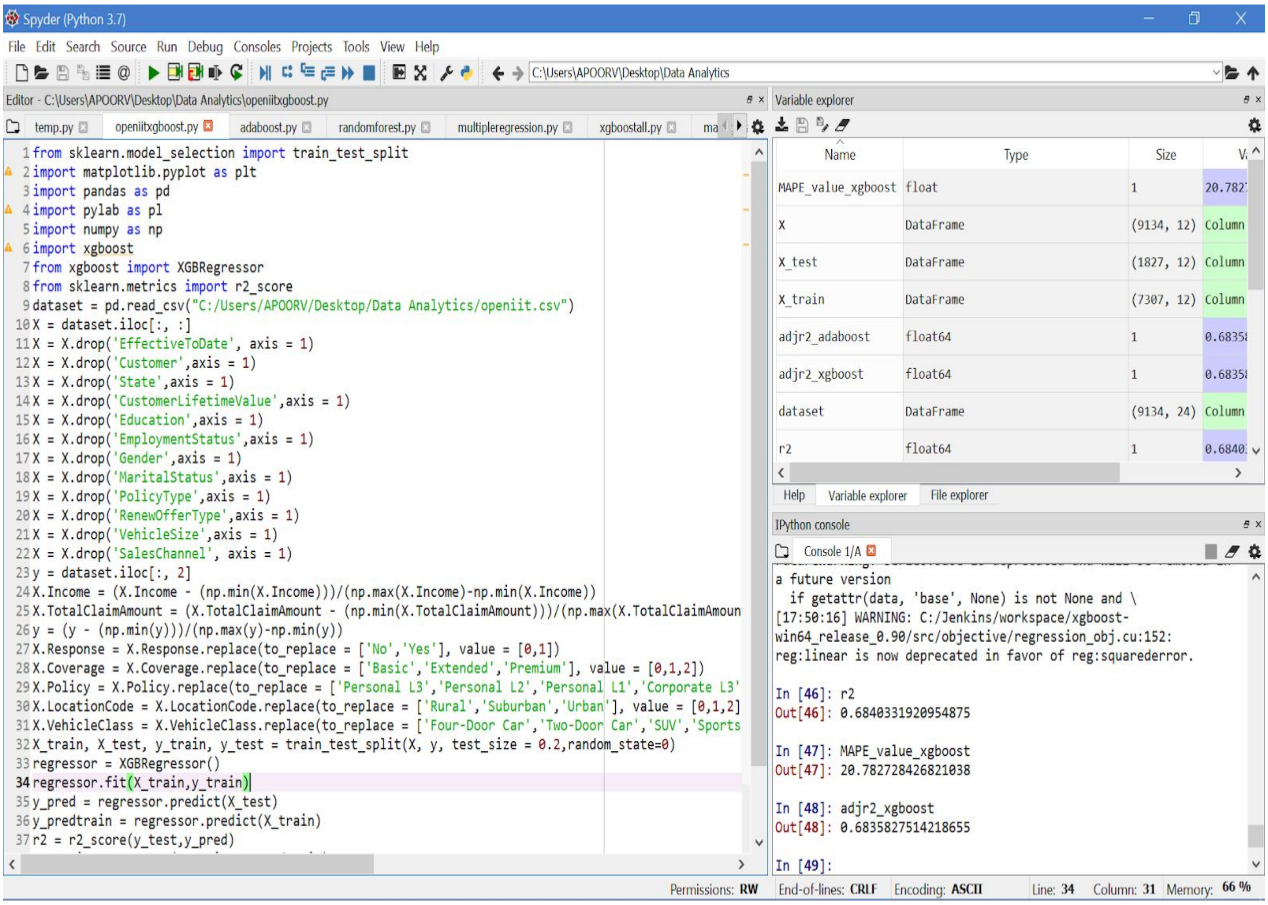
Values Computed:

R-square = 0.684

Adj. R-square = 0.6835

MAPE = 20.78

Code Snippet (Spyder):



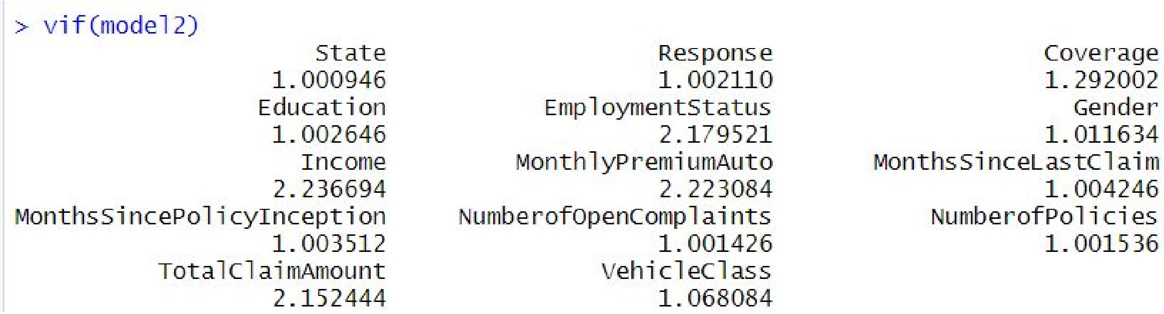
```python
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import pandas as pd
import pylab as pl
import numpy as np
import xgboost
from xgboost import XGBRegressor
from sklearn.metrics import r2_score
dataset = pd.read_csv("C:/Users/APOORV/Desktop/Data Analytics/openiit.csv")
X = dataset.iloc[:, :]
X = X.drop('EffectiveToDate', axis = 1)
X = X.drop('Customer',axis = 1)
X = X.drop('State',axis = 1)
X = X.drop('CustomerLifetimeValue',axis = 1)
X = X.drop('Education',axis = 1)
X = X.drop('EmploymentStatus',axis = 1)
X = X.drop('Gender',axis = 1)
X = X.drop('MaritalStatus',axis = 1)
X = X.drop('PolicyType',axis = 1)
X = X.drop('RenewOfferType',axis = 1)
X = X.drop('VehicleSize',axis = 1)
X = X.drop('SalesChannel', axis = 1)
y = dataset.iloc[:, 2]
X.Income = (X.Income - (np.min(X.Income)))/(np.max(X.Income)-np.min(X.Income))
X.TotalClaimAmount = (X.TotalClaimAmount - (np.min(X.TotalClaimAmount)))/(np.max(X.TotalClaimAmoun
y = (y - (np.min(y)))/(np.max(y)-np.min(y))
X.Response = X.Response.replace(to_replace = ['No','Yes'], value = [0,1])
X.Coverage = X.Coverage.replace(to_replace = ['Basic','Extended','Premium'], value = [0,1,2])
X.Policy = X.Policy.replace(to_replace = ['Personal L3','Personal L2','Personal L1','Corporate L3'
X.LocationCode = X.LocationCode.replace(to_replace = ['Rural','Suburban','Urban'], value = [0,1,2]
X.VehicleClass = X.VehicleClass.replace(to_replace = ['Four-Door Car','Two-Door Car','SUV','Sports
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,random_state=0)
regressor = XGBRegressor()
regressor.fit(X_train,y_train)
y_pred = regressor.predict(X_test)
y_predtrain = regressor.predict(X_train)
r2 = r2_score(y_test,y_pred)
```

Name	Type	Size	V
MAPE_value_xgboost	float	1	20.782
X	DataFrame	(9134, 12)	Column
X_test	DataFrame	(1827, 12)	Column
X_train	DataFrame	(7307, 12)	Column
adjr2_adaboost	float64	1	0.6835
adjr2_xgboost	float64	1	0.6835
dataset	DataFrame	(9134, 24)	Column
r2	float64	1	0.6840

```
a future version
  if getattr(data, 'base', None) is not None and \
[17:50:16] WARNING: C:/Jenkins/workspace/xgboost-
win64_release_0.90/src/objective/regression_obj.cu:152:
reg:linear is now deprecated in favor of reg:squarederror.

In [46]: r2
Out[46]: 0.6840331920954875

In [47]: MAPE_value_xgboost
Out[47]: 20.782728426821038

In [48]: adjr2_xgboost
Out[48]: 0.6835827514218655

In [49]:
```

Statistical tests and Assumption

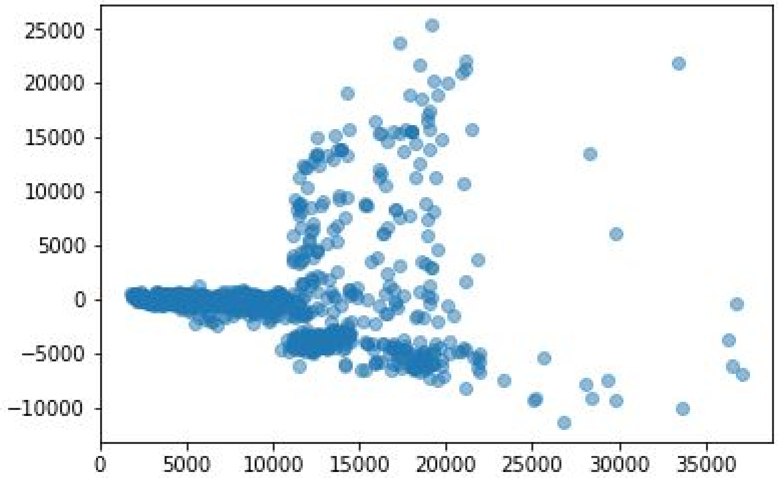
1. **Multicollinearity**: This phenomenon exist when the independent variables are correlated. With correlated variables it becomes a tough task to figure out the true relationship of predictors with response variable. We can use VIF factor to determine multicollinearity. VIF values are shown below-



```
> vif(model2)
                     State                  Response                 Coverage
                  1.000946                  1.002110                 1.292002
                 Education          EmploymentStatus                   Gender
                  1.002646                  2.179521                 1.011634
                    Income        MonthlyPremiumAuto       MonthsSinceLastClaim
                  2.236694                  2.223084                 1.004246
MonthsSincePolicyInception     NumberofOpenComplaints          NumberofPolicies
                  1.003512                  1.001426                 1.001536
          TotalClaimAmount              VehicleClass
                  2.152444                  1.068084
```

Since VIF value for all variables is <=5 there is no multicollinearity.

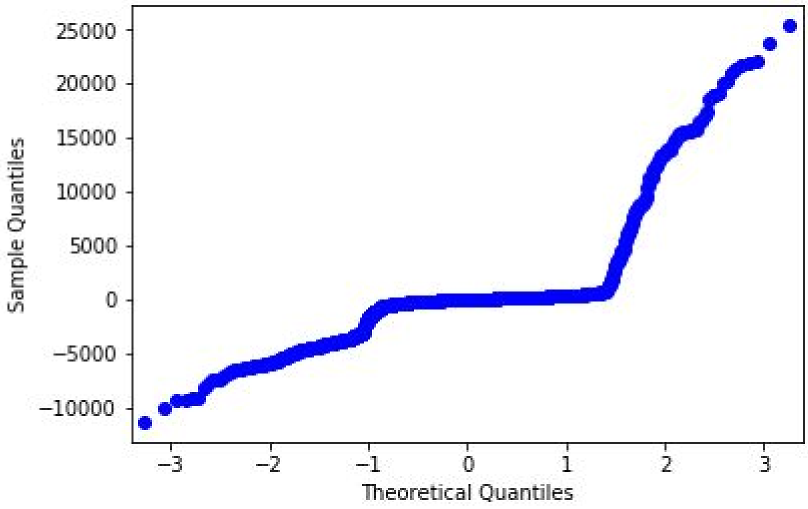
2. **Homoskedasticity**: The error terms must have constant variance. This phenomenon is known as homoskedasticity. We can check t by residual-fitted plot.



Since, it is not of funnel shape hence homoskedasticity is evident

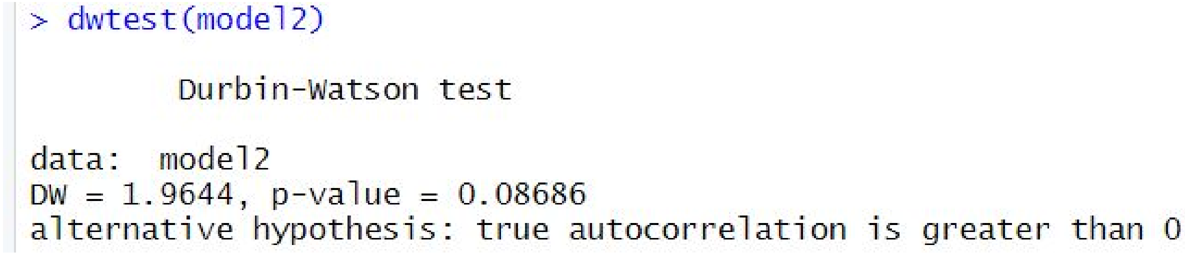
3. **Normality:** If the error terms are non- normally distributed, confidence intervals may become too wide or narrow. Once confidence interval becomes unstable, it leads to difficulty in estimating coefficients based on minimization of least squares.

We can check normality by qq plot as shown below:



We get almost a straight line, hence error terms or residuals are normally distributed.

4. **<u>Autocorrelation</u>:** Autocorrelation occurs when the residuals are not independent from each other. In other words when the value of y(x+1) is not independent from the value of y(x). To check it we use Durbin Watson test, if we get the durbin watson value is between 0 and 2, there is positive correlation, if it is 2 no correlation and greater than 2 negative correlation.



```
> dwtest(model2)

	Durbin-Watson test

data:  model2
DW = 1.9644, p-value = 0.08686
alternative hypothesis: true autocorrelation is greater than 0
```

DW value is close to 2, hence almost no correlation.

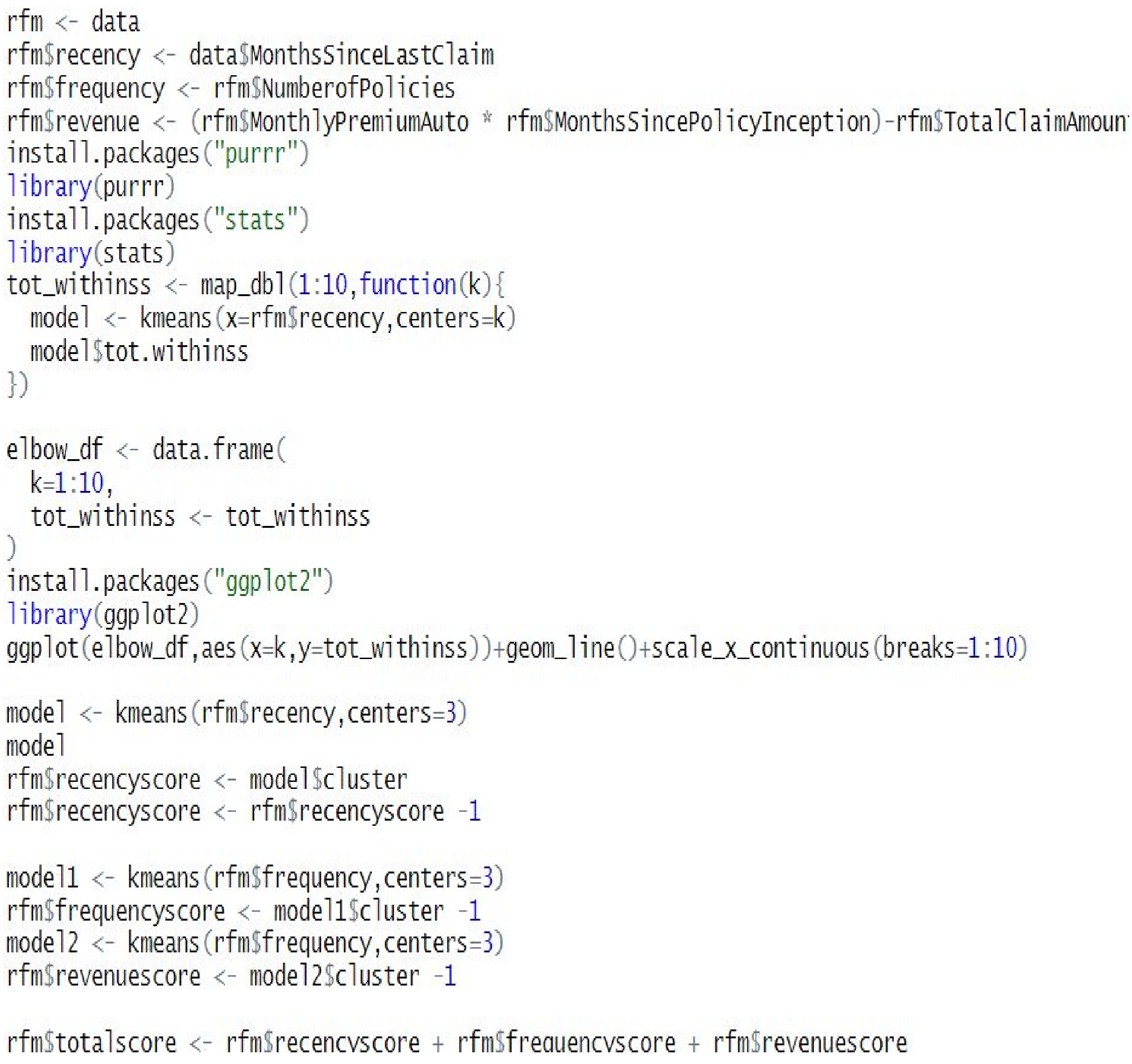
RFM Analysis

Variables used for RFM Analysis:

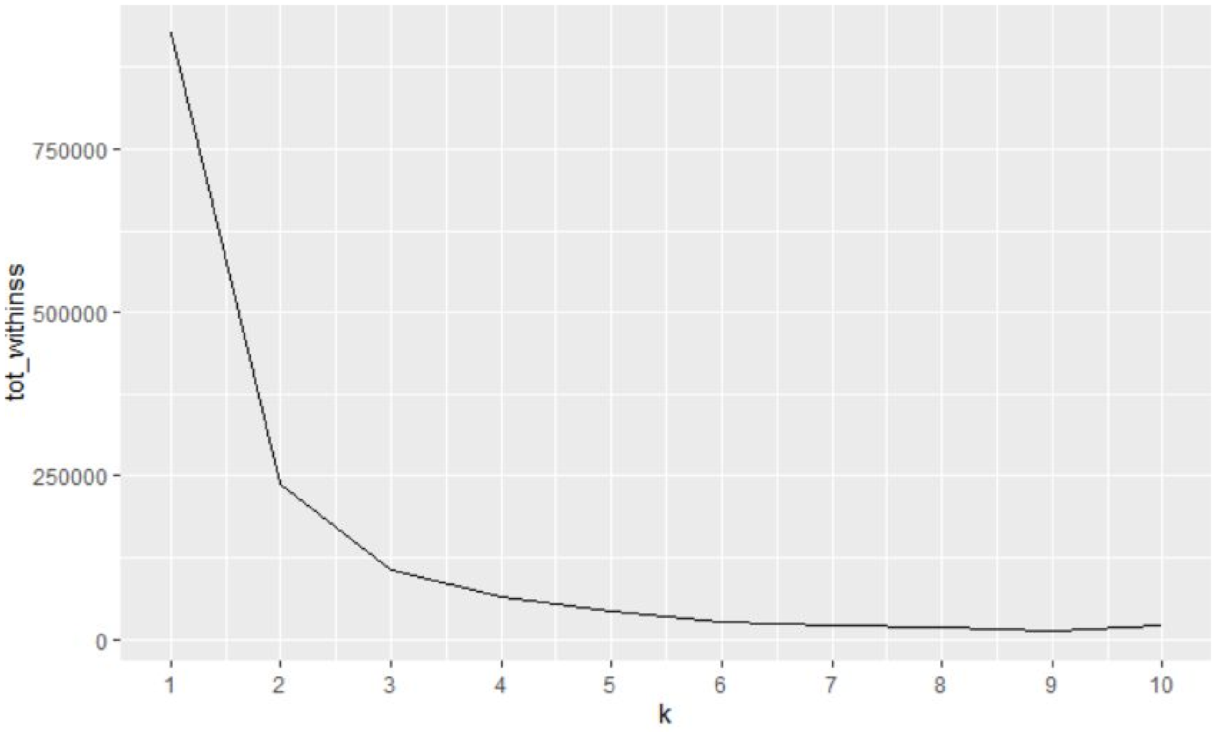
1. Recency = Months Since Last Claim

2. Frequency = Number of Policies
3. Monterey = Revenue =
   (Monthly Premium Auto * Month Since Last Inception) - Total Claim Amount

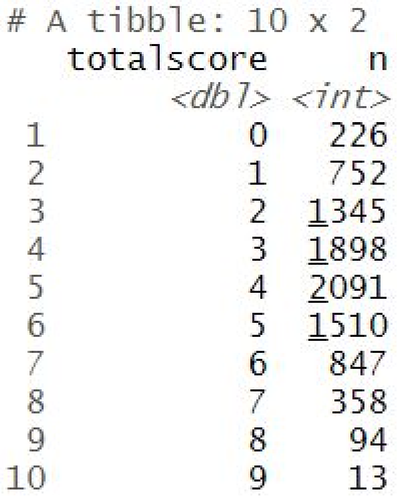
Code Snippet (RStudio):



```
rfm <- data
rfm$recency <- data$MonthsSinceLastClaim
rfm$frequency <- rfm$NumberofPolicies
rfm$revenue <- (rfm$MonthlyPremiumAuto * rfm$MonthsSincePolicyInception)-rfm$TotalClaimAmoun
install.packages("purrr")
library(purrr)
install.packages("stats")
library(stats)
tot_withinss <- map_dbl(1:10,function(k){
  model <- kmeans(x=rfm$recency,centers=k)
  model$tot.withinss
})

elbow_df <- data.frame(
  k=1:10,
  tot_withinss <- tot_withinss
)
install.packages("ggplot2")
library(ggplot2)
ggplot(elbow_df,aes(x=k,y=tot_withinss))+geom_line()+scale_x_continuous(breaks=1:10)

model <- kmeans(rfm$recency,centers=3)
model
rfm$recencyscore <- model$cluster
rfm$recencyscore <- rfm$recencyscore -1

model1 <- kmeans(rfm$frequency,centers=3)
rfm$frequencyscore <- model1$cluster -1
model2 <- kmeans(rfm$frequency,centers=3)
rfm$revenuescore <- model2$cluster -1

rfm$totalscore <- rfm$recencyscore + rfm$frequencyscore + rfm$revenuescore
```



We used k-means clustering to evaluate recency, frequency and revenue score. Finally we have the total score for each customer by adding all the scores.



```
# A tibble: 10 x 2
   totalscore     n
        <dbl> <int>
 1          0   226
 2          1   752
 3          2  1345
 4          3  1898
 5          4  2091
 6          5  1510
 7          6   847
 8          7   358
 9          8    94
10          9    13
```

Customers with high total score are our profitable customers.

So getting back to our unanswered questions:

Q - Who are firm's best customers?

A - After analysing RFM model, that customers will generate more revenue in the future who have at least 7 number of policies and have claimed in the recent months.

Q - Which customers are at the verge of withdrawal?

A - Our analysis shows that customers who have only a few number of policies mostly 1 and who have been inactive for more than roughly 27 months are likely to withdraw their membership with the insurance company.

Q - Who has the potential to be converted in more profitable customers?

A - Customers with the total score of 7 or 8 has the potential to be converted into profitable customers.

Q - Who are lost customers that you don't need to pay much attention to?

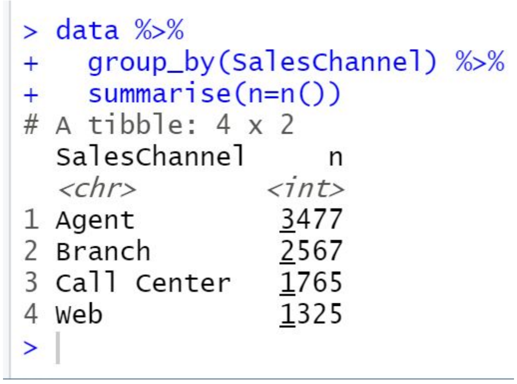
A - Customers whose Month Since Last Claim is greater than roughly 30 months and still they not generated positive revenue.

Q - Who are your loyal customers?

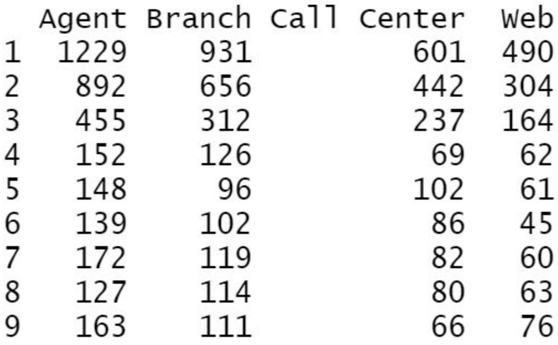
A - Customers who have been with the firm for more than 5 to 6 years and have developed an increasing positive revenue over the years. These customers have trust in the firm.

Business Recommendations

1. **Increase Number of Agents-** In an industry that's as competitive as auto insurance, smart agents are always on the lookout for new ways to get an edge.As we can see from the following data maximum sales was through Agents for any number of policies. We need to make sure that there are sufficient number of agents that can reach the target.



```
> data %>%
+   group_by(SalesChannel) %>%
+   summarise(n=n())
# A tibble: 4 x 2
  SalesChannel     n
  <chr>        <int>
1 Agent         3477
2 Branch        2567
3 Call Center   1765
4 Web           1325
> 
```



```
  Agent Branch Call Center  Web
1  1229    931         601  490
2   892    656         442  304
3   455    312         237  164
4   152    126          69   62
5   148     96         102   61
6   139    102          86   45
7   172    119          82   60
8   127    114          80   63
9   163    111          66   76
```

2. **Use consumer Surveys -** The more information you have on your clients, the better you will be able to serve them. Just soliciting surveys lets your clients know that you put some value on their experience, and the survey results themselves can identify opportunities for improvement. Most email marketing services will help you set up email surveys; consider giving out some prize or incentive for completing the questionnaire to get as many results as possible. Give your clients the opportunity to fill in their own custom comments and post the best ones on your website to build trust with new consumers.

3.**Encourage client Referrals -** Another way to use surveys is to identify your customers who are most likely to refer you. In your survey, simply ask a question such as "How likely are you to recommend your agent to a friend?" If you use survey software like SuveyMonkey that can connect responses to the email address of the respondent, you can determine exactly which of your clients are most likely to refer you, and then you can send them follow-up marketing and incentives to refer you.

Conclusion

1. Concluding that XGBoost Model is stable in order to predict the Customer Lifetime Value. We got a r-square value of 0.684 for the validation data set.

2. Types of customers who will generate more revenue are the customers having RFM total score of greater than 7. RFM analysis helped in customer segmentation and eventually it will help the insurance company in increasing customer retention and conversion rate.

Thank you!

-Team XBoosters!