Lead Score Prediction

Logistic Regression

Data Overview

- There are 9240 rows and 37 columns (36 columns and 1 target).
- There are more categorical columns than numerical columns.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9240 entries, 0 to 9239
Data columns (total 37 columns):
 #   Column                                         Non-Null Count  Dtype
---  ------                                         --------------  -----
 0   Prospect ID                                    9240 non-null   object
 1   Lead Number                                    9240 non-null   int64
 2   Lead Origin                                    9240 non-null   object
 3   Lead Source                                    9204 non-null   object
 4   Do Not Email                                   9240 non-null   object
 5   Do Not Call                                    9240 non-null   object
 6   Converted                                      9240 non-null   int64
 7   TotalVisits                                    9103 non-null   float64
 8   Total Time Spent on Website                    9240 non-null   int64
 9   Page Views Per Visit                           9103 non-null   float64
 10  Last Activity                                  9137 non-null   object
 11  Country                                        6779 non-null   object
 12  Specialization                                 7802 non-null   object
 13  How did you hear about X Education             7033 non-null   object
 14  What is your current occupation                6550 non-null   object
 15  What matters most to you in choosing a course  6531 non-null   object
 16  Search                                         9240 non-null   object
 17  Magazine                                       9240 non-null   object
 18  Newspaper Article                              9240 non-null   object
 19  X Education Forums                             9240 non-null   object
 20  Newspaper                                      9240 non-null   object
 21  Digital Advertisement                          9240 non-null   object
 22  Through Recommendations                        9240 non-null   object
 23  Receive More Updates About Our Courses         9240 non-null   object
 24  Tags                                           5887 non-null   object
 25  Lead Quality                                   4473 non-null   object
 26  Update me on Supply Chain Content              9240 non-null   object
 27  Get updates on DM Content                      9240 non-null   object
 28  Lead Profile                                   6531 non-null   object
 29  City                                           7820 non-null   object
 30  Asymmetrique Activity Index                    5022 non-null   object
 31  Asymmetrique Profile Index                     5022 non-null   object
 32  Asymmetrique Activity Score                    5022 non-null   float64
 33  Asymmetrique Profile Score                     5022 non-null   float64
 34  I agree to pay the amount through cheque       9240 non-null   object
 35  A free copy of Mastering The Interview         9240 non-null   object
 36  Last Notable Activity                          9240 non-null   object
dtypes: float64(4), int64(3), object(30)
memory usage: 2.6+ MB
```

Dropping the skewed columns

- Skewed columns are something which has around 90% of its value to be one specific value/category and the remaining 10% will be other values. This must be dropped as there is no use in having them as a feature while training the model.
- Following are the columns of skewed values.

```
cols_to_drop = ['Do Not Email', 'Do Not Call', 'What matters most to you in choosing a course', 'Search',
                'Magazine', 'Newspaper Article', 'X Education Forums', 'Newspaper', 'Digital Advertisement',
                'Through Recommendations', 'Receive More Updates About Our Courses', 'Update me on Supply Chain Cont
                'Get updates on DM Content', 'I agree to pay the amount through cheque']
```

Replace “Select” with “Nan”

- “Select” is the default option in any drop down whenever you fill a form online and if no option is chosen then “select” is the options recorded.

Checking the missing value percentage

Checking the Missing value %

```
round(input_data.isnull().mean()*100,2)
```

```
Prospect ID                                       0.00
Lead Number                                       0.00
Lead Origin                                       0.00
Lead Source                                       0.39
Converted                                         0.00
TotalVisits                                       1.48
Total Time Spent on Website                       0.00
Page Views Per Visit                              1.48
Last Activity                                     1.11
Country                                          26.63
Specialization                                   36.58
How did you hear about X Education               78.46
What is your current occupation                  29.11
Tags                                             36.29
Lead Quality                                     51.59
Lead Profile                                     74.19
City                                             39.71
Asymmetrique Activity Index                      45.65
Asymmetrique Profile Index                       45.65
Asymmetrique Activity Score                      45.65
Asymmetrique Profile Score                       45.65
A free copy of Mastering The Interview            0.00
Last Notable Activity                             0.00
dtype: float64
```

Drop the columns with missing value>40%

```
round(input_data.isnull().mean()*100,2)
```

```
Prospect ID                                  0.00
Lead Number                                  0.00
Lead Origin                                  0.00
Lead Source                                  0.39
Converted                                    0.00
TotalVisits                                  1.48
Total Time Spent on Website                  0.00
Page Views Per Visit                         1.48
Last Activity                                1.11
Country                                     26.63
Specialization                              36.58
What is your current occupation             29.11
Tags                                        36.29
City                                        39.71
A free copy of Mastering The Interview       0.00
Last Notable Activity                        0.00
dtype: float64
```

Impute missing values

- when the distribution is skewed, we can use MODE (most repeated value)
- when the distribution is almost evenly distributed, we can use "others" as a new category to impute missing values.

```
input_data['Lead Source'].fillna('others',inplace=True)
input_data['Specialization'].fillna('others',inplace=True)
input_data['Tags'].fillna('others',inplace=True)
```

```
input_data['Country'].fillna(input_data['Country'].mode()[0],inplace=True)
input_data['What is your current occupation'].fillna(input_data['What is your current occupation'].mode()[0],inplace
input_data['City'].fillna(input_data['City'].mode()[0],inplace=True)
```

Dropping the duplicate records & dropping other missing value records

- Dropping Nan records

```
input_data.dropna(inplace=True)
```

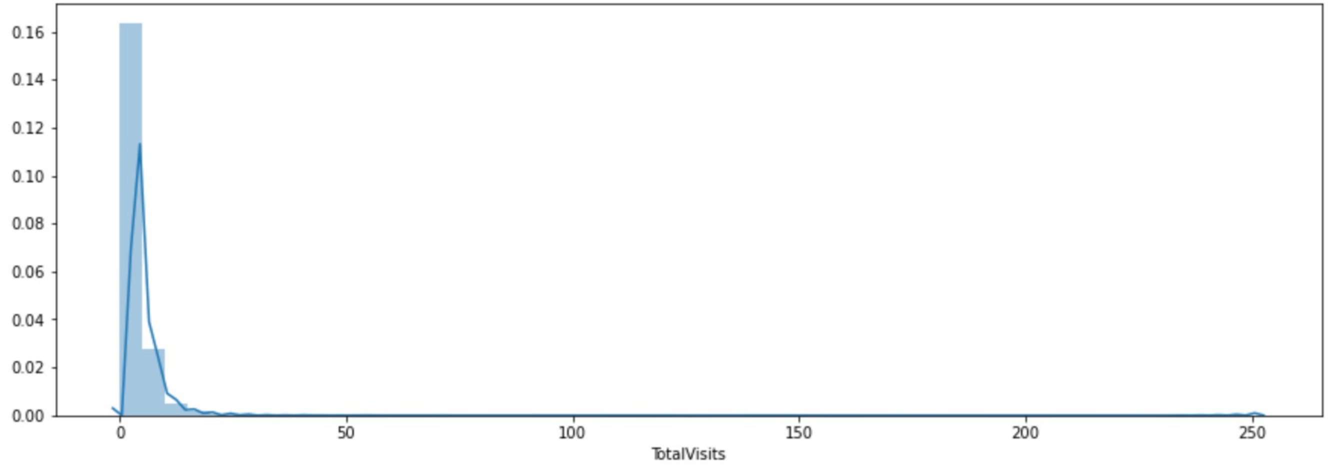
- Dropping Duplicate records

```
input_data.drop_duplicates(inplace=True)
```

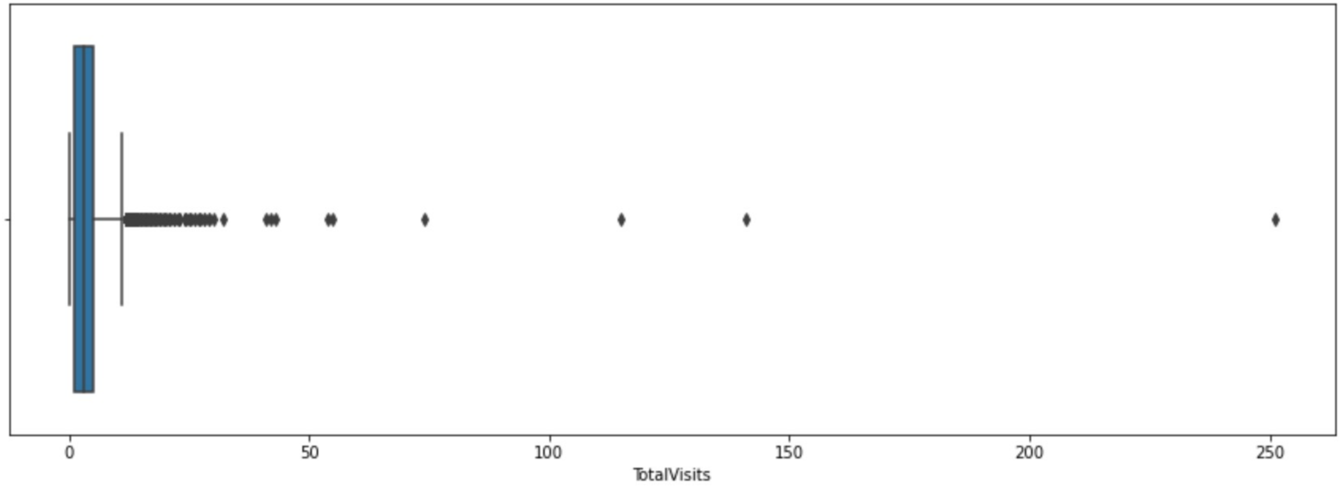
Checking the distribution and outliers

- This is for the numerical columns
    - TotalVisits
    - Total time spent on websites
    - Page views per visit

Total Visits

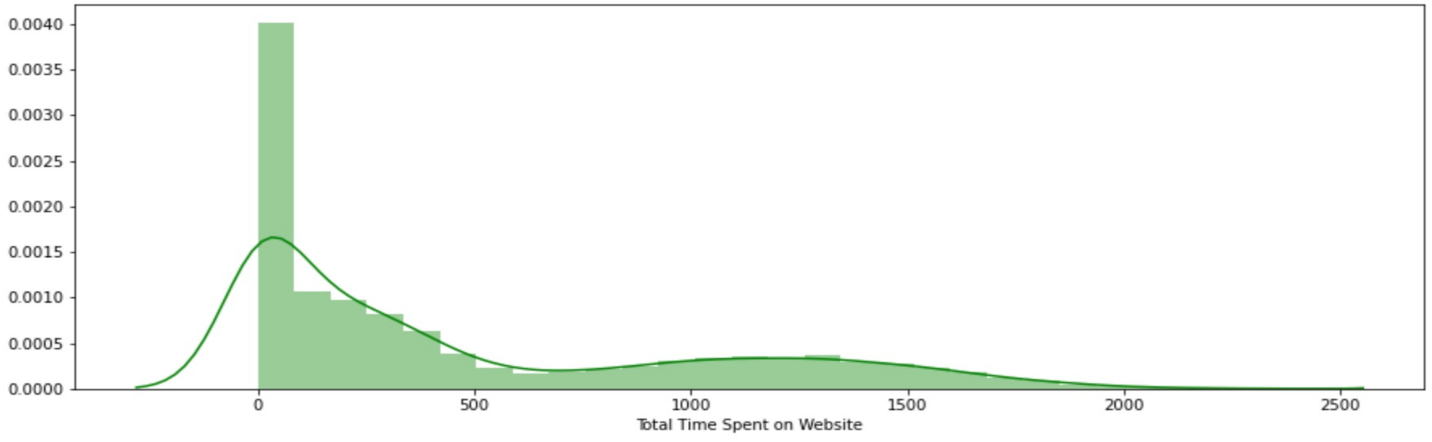


Skewed data between 0 - 20

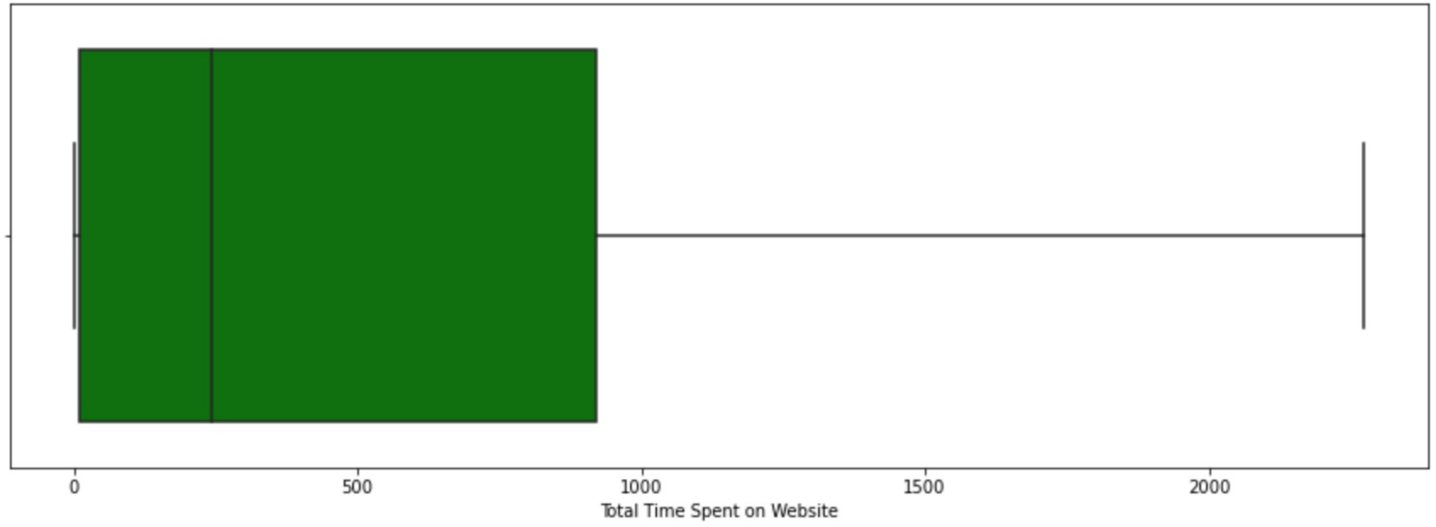


Obviously, there are outliers

Total time spent on website

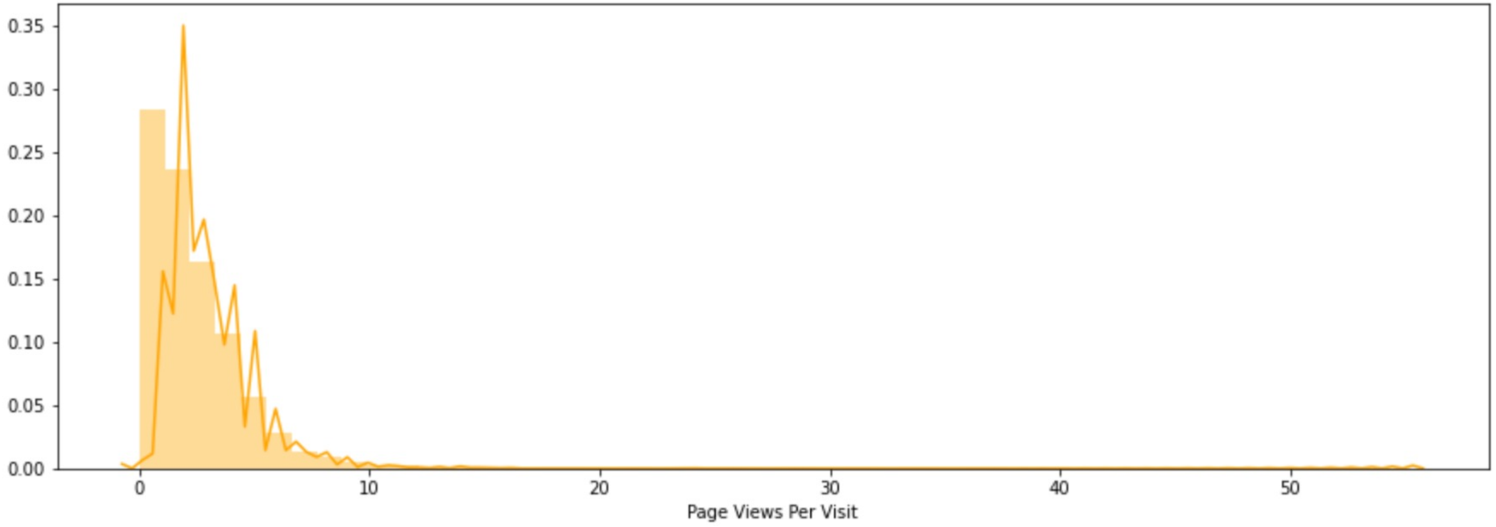


Skewed data again but if it is adjusted using Log it will be converted to normal.

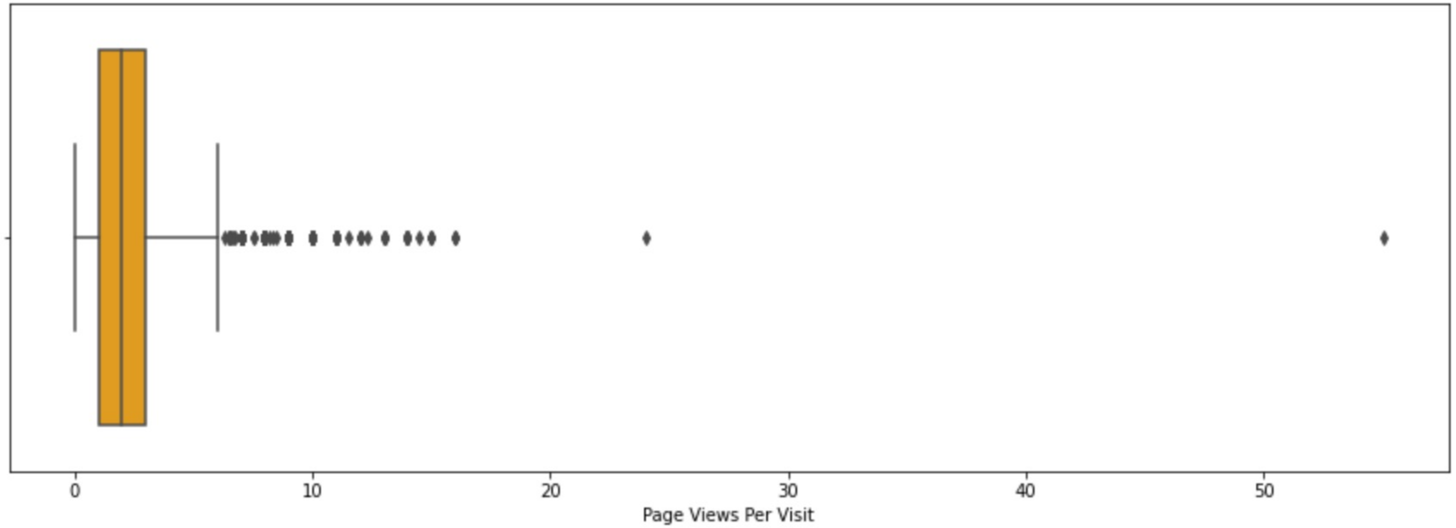


No outlier as such when we observe

Page Views per visit



Skewed data between 0 - 20

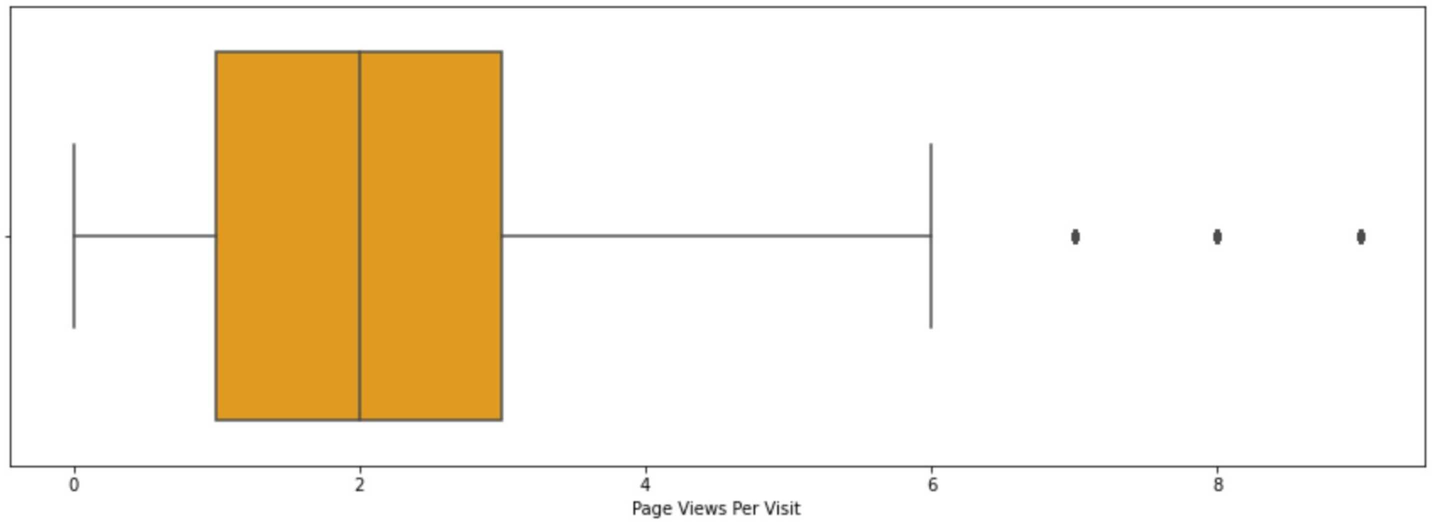


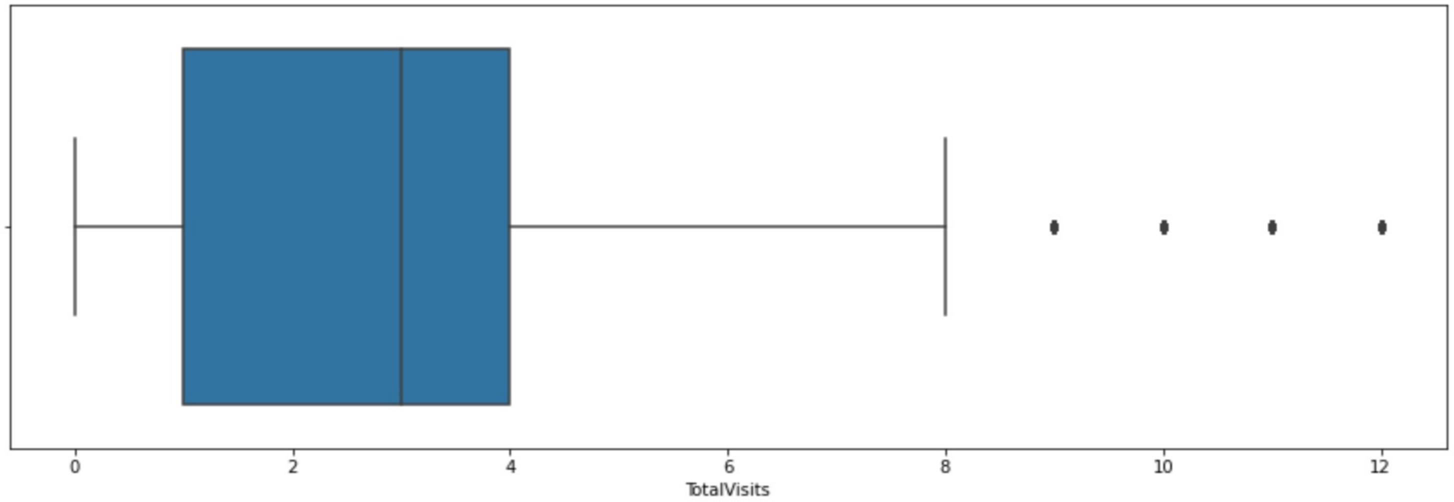
Obviously, there are outliers

Outlier Treatment

- **Outlier treatments are subjective**
- Usual method :
  - should be greater than Q1 - (1.5 * IQR)
  - should be lesser than Q3 + (1.5 * IQR)
- Used method below :
  - should be greater than percentile(10%) - (1.5 * IQR)
  - should be lesser than percentile(80%) + (1.5 * IQR)

After treatment





Drop Junk Values

- As per the explanation
  - - Closed by Horizzon
  - - Lost to EINS
- the above 2 features means that the leads are lost or closed by the competitors, but the data says that these leads are converted (1), So Dropping these features because of the discrepancies (This is under the column Tag).

Encoding & Scaling the data

- One hot encoding is used to convert the categorical value into numerical value.
- As most of our data is binary, we are choosing MINMAXSCALER to have our distribution between 0-1 which will be easier to get the coefficients faster.

Feature Selection

- Stats model
  - Here we use the p-value in order to choose the best features for the model
- RFE
  - Recursive feature elimination helps to choose a specific number of features using the model results
- VIF
  - Helps to identify the if there is any multicollinearity between the selected features

Stats model

- Scaled data is then fed into the model and p-value is checked to choose the right features.

```
In [56]: import statsmodels.api as sm

In [57]: log_stats_mdl = sm.GLM(np.array(y_train).reshape(-1,1), sm.add_constant(x_train_scaled), f

In [58]: log_stats_mdl.fit().summary()
```

Dep. Variable:	y	**No. Observations:**	6700
Model:	GLM	**Df Residuals:**	6562
Model Family:	Binomial	**Df Model:**	137
Link Function:	logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-1249.5
Date:	Mon, 08 Mar 2021	**Deviance:**	2499.0
Time:	22:21:48	**Pearson chi2:**	2.18e+04
No. Iterations:	24		
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
const	40.7915	3.09e+05	0.000	1.000	-6.05e+05	6.05e+05
TotalVisits	1.3713	0.404	3.391	0.001	0.579	2.164

RFE – choosing top 20 features

- Top 20 features relevant for the model.

```
In [68]: top_20_features = list(rfe_rank.sort_values(['rank']).head(20)['cols'])
         top_20_features

Out[68]: ['Lead Source_Referral Sites',
          'Last Notable Activity_Olark Chat Conversation',
          'Tags_Busy',
          'Last Activity_Email Bounced',
          'Tags_Interested in Next batch',
          'Tags_Lateral student',
          'Tags_Ringing',
          'Tags_Will revert after reading the email',
          'Tags_in touch with EINS',
          'Lead Source_Welingak Website',
          'Tags_invalid number',
          'Lead Source_Organic Search',
          'Last Activity_SMS Sent',
          'Tags_others',
          'Tags_switched off',
          'Lead Source_Google',
          'Tags_wrong number given',
          'Total Time Spent on Website',
          'Lead Source_Direct Traffic',
          'Last Notable Activity_Modified']
```

VIF

- VIF scores are lesser than 5 which means there is no correlation between the features

	cols	vif_score
17	Total Time Spent on Website	2.479571
15	Lead Source_Google	2.343825
18	Lead Source_Direct Traffic	2.078224
13	Tags_others	1.887296
7	Tags_Will revert after reading the email	1.843569
12	Last Activity_SMS Sent	1.716140
19	Last Notable Activity_Modified	1.512612
11	Lead Source_Organic Search	1.511207
6	Tags_Ringing	1.448102
3	Last Activity_Email Bounced	1.117242
14	Tags_switched off	1.113838
2	Tags_Busy	1.094434
9	Lead Source_Welingak Website	1.055093
0	Lead Source_Referral Sites	1.046636
1	Last Notable Activity_Olark Chat Conversation	1.045236
10	Tags_invalid number	1.032151
16	Tags_wrong number given	1.021056
4	Tags_Interested in Next batch	1.005696
8	Tags_in touch with EINS	1.004770
5	Tags_Lateral student	1.003717

Model Building

- We use a sklearn logistic regression model (which by default has 0.5 as the threshold for the prediction)

```
In [72]: log_mdl.fit(x_train_scaled[top_20_features], y_train)

Out[72]: LogisticRegression()

In [73]: y_pred = log_mdl.predict(x_test_scaled[top_20_features])
         y_pred_train = log_mdl.predict(x_train_scaled[top_20_features])
```

Model Results

- Scores are above 80% as expected.

Test Recall Score

```
In [75]: recall_score(y_test, y_pred)
Out[75]: 0.8646003262642741
```

Test Precision Score

```
In [76]: precision_score(y_test,y_pred)
Out[76]: 0.9314586994727593
```

Train Recall Score

```
In [77]: recall_score(y_train, y_pred_train)
Out[77]: 0.8557650153441473
```

Test Precision Score

```
In [78]: precision_score(y_train, y_pred_train)
Out[78]: 0.9255571360834519
```

Confusion matrix & AUC-ROC score

Confusion Matric and classification Report (only on test data)

```
In [79]: from sklearn.metrics import confusion_matrix, classification_report
```

```
In [80]: confusion_matrix(y_test,y_pred)
```

```
Out[80]: array([[1024,   39],
                [  83,  530]])
```

```
In [81]: print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.93      0.96      0.94      1063
           1       0.93      0.86      0.90       613

    accuracy                           0.93      1676
   macro avg       0.93      0.91      0.92      1676
weighted avg       0.93      0.93      0.93      1676
```

Auc-Roc Score (only on test data)

```
In [82]: from sklearn.metrics import roc_auc_score
         roc_auc_score(y_test, y_pred)
```

```
Out[82]: 0.9139558545714597
```

Feature importance

	Coefficients
Tags_Will revert after reading the email	6.207466
Total Time Spent on Website	3.865871
Lead Source_Welingak Website	3.129140
Tags_Busy	2.100997
Last Activity_SMS Sent	1.955803
Tags_Lateral student	1.718531
Tags_others	1.550068
Tags_in touch with EINS	1.241696
Tags_Interested in Next batch	1.078165
Last Activity_Email Bounced	-1.046988
Lead Source_Organic Search	-1.113832
Lead Source_Google	-1.122498
Tags_invalid number	-1.164388
Tags_wrong number given	-1.170474
Lead Source_Referral Sites	-1.287500
Lead Source_Direct Traffic	-1.487034

Result

The result here has lead id/number, lead prediction (binary) and the probability of getting converted. This Helps the intern or sales team to easily pick the lead and Approach them to convert the lead by helping them understand the Service and benefits they would get if they get converted. The Threshold can be modified to approach the leads better. The leads With Higher probability will be chosen as they are the hot leads to Get converted and the cold leads are chosen to understand the reason where we can improve

	lead_num	lead_pred	lead_pred_prob
837	601868	1	99.96
269	630200	1	99.95
289	641173	1	99.94
335	608835	1	99.93
603	639056	1	99.93
1284	608183	1	99.92
468	628916	1	99.92
622	601618	1	99.92
952	608709	1	99.92
1541	647404	1	99.91
168	647201	1	99.91