LEAD SCORE ASSIGNMENT

Problem Statement

An education company named X Education sells online courses to industry professionals

X Education gets a lot of leads, its lead conversion rate is very poor. For example, if, say, they acquire 100 leads in a day, only about 30 of them are converted. To make this process more efficient, the company wishes to identify the most potential leads, also known as ‘Hot Leads’. If they successfully identify this set of leads, the lead conversion rate should go up as the sales team will now be focusing more on communicating with the potential leads rather than making calls to everyone. A typical lead conversion process can be represented using the following funnel

Problem Statement

Social Media Marketing
Initial Pool of Leads
Lead Nurturing
Hot leads
£ ₹
€ $
Converted leads

As you can see, there are a lot of leads generated in the initial stage (top) but only a few of them come out as paying customers from the bottom. In the middle stage, you need to nurture the potential leads well (i.e. educating the leads about the product, constantly communicating etc.) in order to get a higher lead conversion.

X Education has appointed you to help them select the most promising leads, i.e. the leads that are most likely to convert into paying customers. The CEO, in particular, has given a ballpark of the target lead conversion rate to be around 80%.

Goals of the case study

- Build a logistic regression model to assign a lead score between 0 and 100 to each of the leads which can be used by the company to target potential leads. A higher score would mean that the lead is hot, i.e. is most likely to convert whereas a lower score would mean that the lead is cold and will mostly not get converted.
- There are some more problems presented by the company which your model should be able to adjust to if the company's requirement changes in the future so you will need to handle these as well. These problems are provided in a separate doc file. Please fill it based on the logistic regression model you got in the first step. Also, make sure you include this in your final PPT where you'll make recommendations.

APPROACH

- Data cleaning and preparation
    - Combining three dataframes
    - Handling categorical variables
        - Mapping categorical variables to integers
        - Dummy variable creation
    - Handling missing values
- Test-train split and scaling

APPROACH

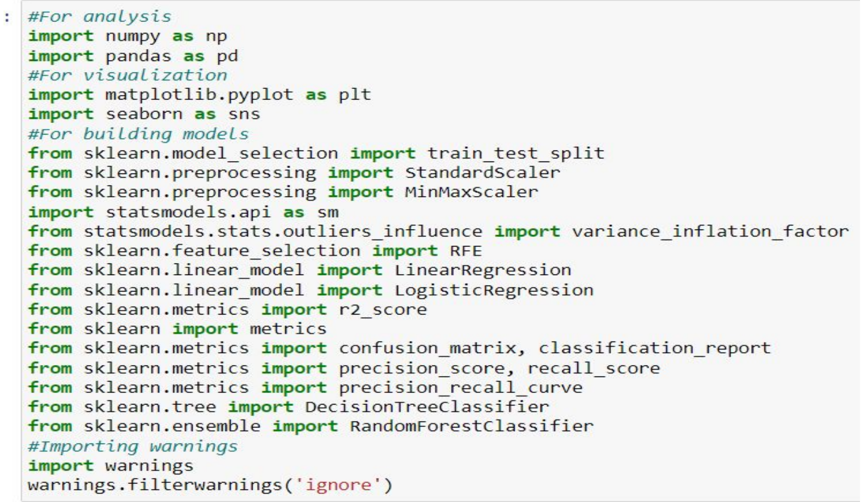
- Model Building
  - Feature elimination based on correlations
  - Feature selection using RFE (Coarse Tuning)
  - Manual feature elimination (using p-values and VIFs)
- Model Evaluation
  - Accuracy
  - Sensitivity and Specificity
  - Optimal cut-off using ROC curve
  - Precision and Recall
- Predictions on the test set

Importing Libraries

1. Firstly we have insert the files in the python jupyter notebook.

2. After inserting the files we have imported the libraries for the analysis.

3. Then we have started data cleaning from the files



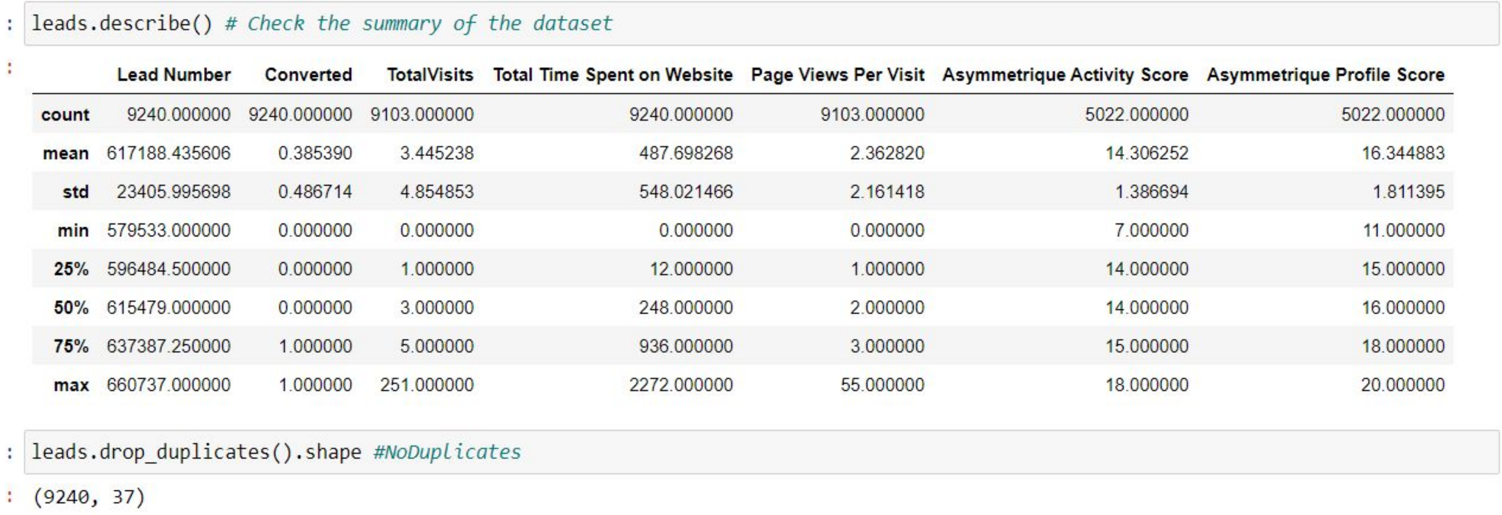
```
#For analysis
import numpy as np
import pandas as pd
#For visualization
import matplotlib.pyplot as plt
import seaborn as sns
#For building models
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor
from sklearn.feature_selection import RFE
from sklearn.linear_model import LinearRegression
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import r2_score
from sklearn import metrics
from sklearn.metrics import confusion_matrix, classification_report
from sklearn.metrics import precision_score, recall_score
from sklearn.metrics import precision_recall_curve
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
#Importing warnings
import warnings
warnings.filterwarnings('ignore')
```

Step 1:- Reading and Understanding the Data

-> During this step we will be doing

- Loading the Dataset
- Reading and Understanding the Data

Step 1:-Reading and Understanding the Data



```
leads.describe() # Check the summary of the dataset
```

	Lead Number	Converted	TotalVisits	Total Time Spent on Website	Page Views Per Visit	Asymmetrique Activity Score	Asymmetrique Profile Score
count	9240.000000	9240.000000	9103.000000	9240.000000	9103.000000	5022.000000	5022.000000
mean	617188.435606	0.385390	3.445238	487.698268	2.362820	14.306252	16.344883
std	23405.995698	0.486714	4.854853	548.021466	2.161418	1.386694	1.811395
min	579533.000000	0.000000	0.000000	0.000000	0.000000	7.000000	11.000000
25%	596484.500000	0.000000	1.000000	12.000000	1.000000	14.000000	15.000000
50%	615479.000000	0.000000	3.000000	248.000000	2.000000	14.000000	16.000000
75%	637387.250000	1.000000	5.000000	936.000000	3.000000	15.000000	18.000000
max	660737.000000	1.000000	251.000000	2272.000000	55.000000	18.000000	20.000000

```
leads.drop_duplicates().shape #NoDuplicates
```

```
(9240, 37)
```

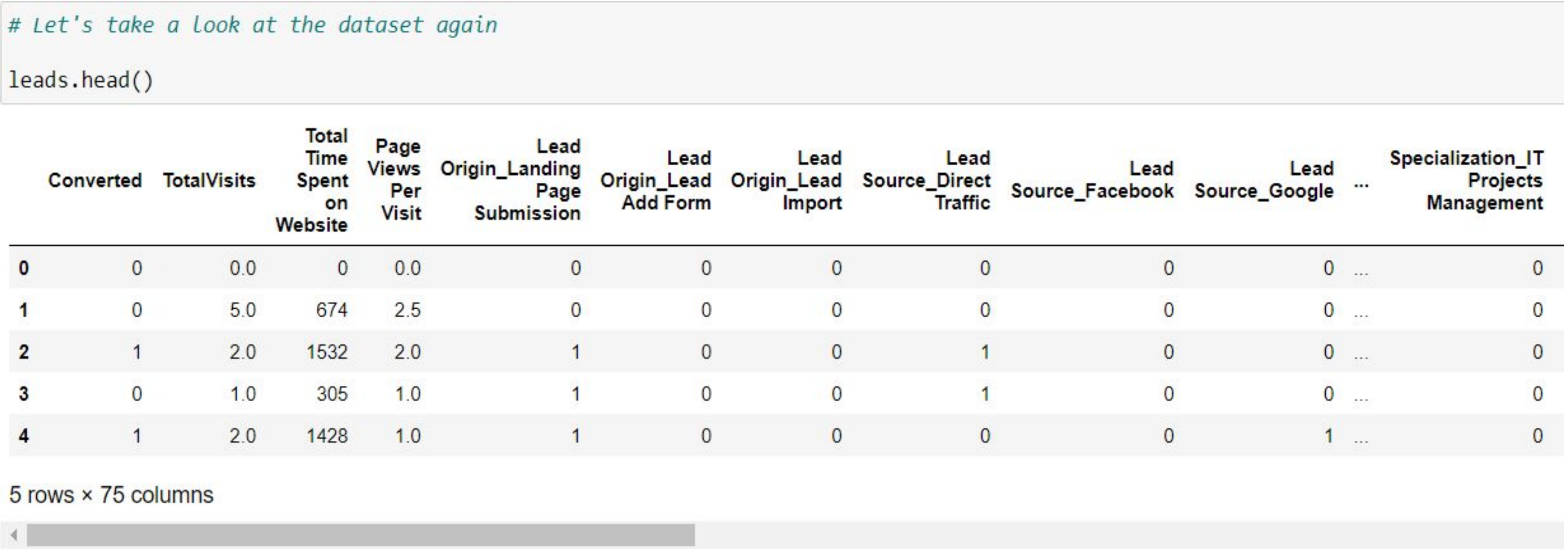
Looks like there are quite a few categorical variables present in this dataset for which we will need to create dummy variables. Also, there are a lot of null values present as well, so we will need to treat them accordingly.

Step 2: Data Cleaning and Preparation

-> In this step we will be handling

- Null values
- Missing values
- Outliers and
- Dropping the unnecessary values
- Creating dummies

Step 2: Data Cleaning and Preparation



```python
# Let's take a look at the dataset again

leads.head()
```

	Converted	TotalVisits	Total Time Spent on Website	Page Views Per Visit	Lead Origin_Landing Page Submission	Lead Origin_Lead Add Form	Lead Origin_Lead Import	Lead Source_Direct Traffic	Lead Source_Facebook	Lead Source_Google	...	Specialization_IT Projects Management
0	0	0.0	0	0.0	0	0	0	0	0	0	...	0
1	0	5.0	674	2.5	0	0	0	0	0	0	...	0
2	1	2.0	1532	2.0	1	0	0	1	0	0	...	0
3	0	1.0	305	1.0	1	0	0	1	0	0	...	0
4	1	2.0	1428	1.0	1	0	0	0	0	1	...	0

5 rows × 75 columns

After handling everything we will be getting the above data set with 5 rows and 75 colums

Test-Train split

- Putting feature variable to X

```
X = leads.drop(['Converted'], axis=1)
```

- Putting target variable to y

```
y = leads['Converted']
```

- Splitting the data into train and test

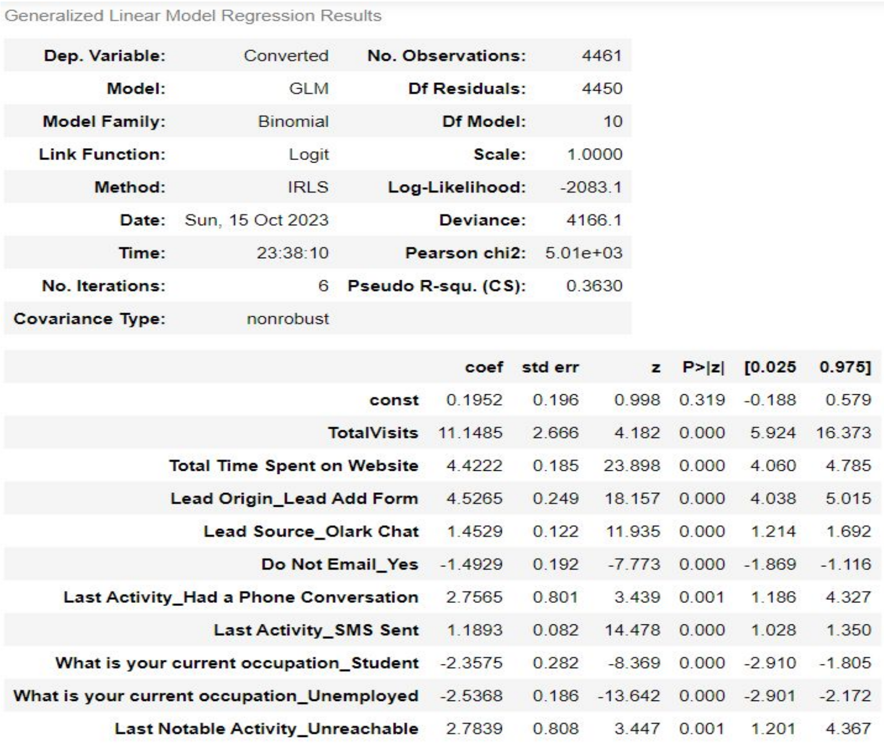
```
X_train, X_test, y_train, y_test = train_test_split(X, y, train_size=0.7,
test_size=0.3, random_state=100)
```

Step 3: Model Building

-> In this step we will use

- RFE
- Logistic Regression
- Making different Models, with considering P values, VIF and Z values

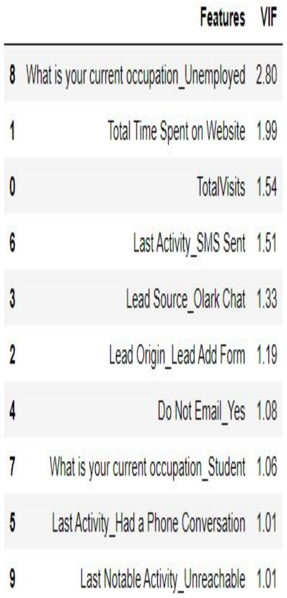
Step 3: Model Building



Generalized Linear Model Regression Results

Dep. Variable:	Converted	**No. Observations:**	4461
Model:	GLM	**Df Residuals:**	4450
Model Family:	Binomial	**Df Model:**	10
Link Function:	Logit	**Scale:**	1.0000
Method:	IRLS	**Log-Likelihood:**	-2083.1
Date:	Sun, 15 Oct 2023	**Deviance:**	4166.1
Time:	23:38:10	**Pearson chi2:**	5.01e+03
No. Iterations:	6	**Pseudo R-squ. (CS):**	0.3630
Covariance Type:	nonrobust		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
const	0.1952	0.196	0.998	0.319	-0.188	0.579
TotalVisits	11.1485	2.666	4.182	0.000	5.924	16.373
Total Time Spent on Website	4.4222	0.185	23.898	0.000	4.060	4.785
Lead Origin_Lead Add Form	4.5265	0.249	18.157	0.000	4.038	5.015
Lead Source_Olark Chat	1.4529	0.122	11.935	0.000	1.214	1.692
Do Not Email_Yes	-1.4929	0.192	-7.773	0.000	-1.869	-1.116
Last Activity_Had a Phone Conversation	2.7565	0.801	3.439	0.001	1.186	4.327
Last Activity_SMS Sent	1.1893	0.082	14.478	0.000	1.028	1.350
What is your current occupation_Student	-2.3575	0.282	-8.369	0.000	-2.910	-1.805
What is your current occupation_Unemployed	-2.5368	0.186	-13.642	0.000	-2.901	-2.172
Last Notable Activity_Unreachable	2.7839	0.808	3.447	0.001	1.201	4.367



	Features	VIF
8	What is your current occupation_Unemployed	2.80
1	Total Time Spent on Website	1.99
0	TotalVisits	1.54
6	Last Activity_SMS Sent	1.51
3	Lead Source_Olark Chat	1.33
2	Lead Origin_Lead Add Form	1.19
4	Do Not Email_Yes	1.08
7	What is your current occupation_Student	1.06
5	Last Activity_Had a Phone Conversation	1.01
9	Last Notable Activity_Unreachable	1.01

Since the P values of all variables is close 0 and VIF values are low for all the variables, Model 6 is our final model. We have 10 variables in our final model.

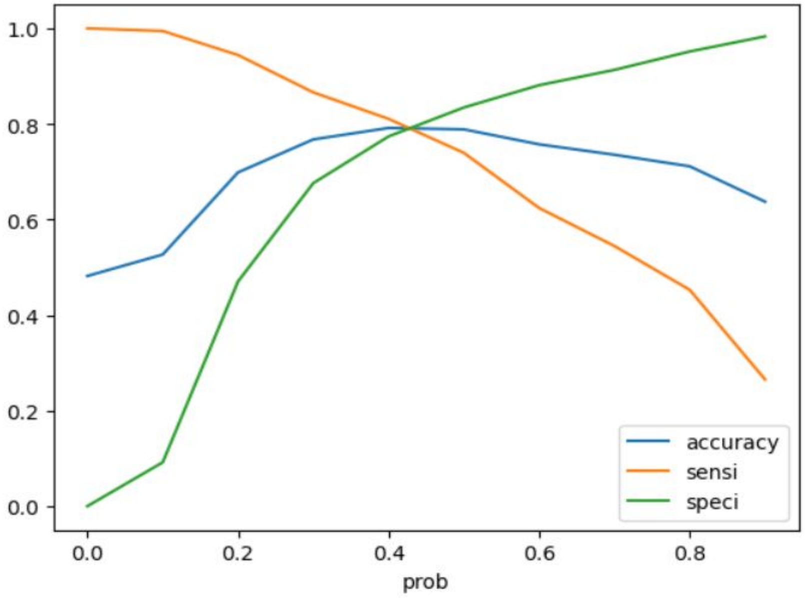
Step 4: Model Evaluation

-> This step involves

- Reshaping
- Create confusion matrix
- Accuracy
- Sensitivity
- Specificity
- False Positive Rate
- Positive Predictive Value
- Negative predictive value

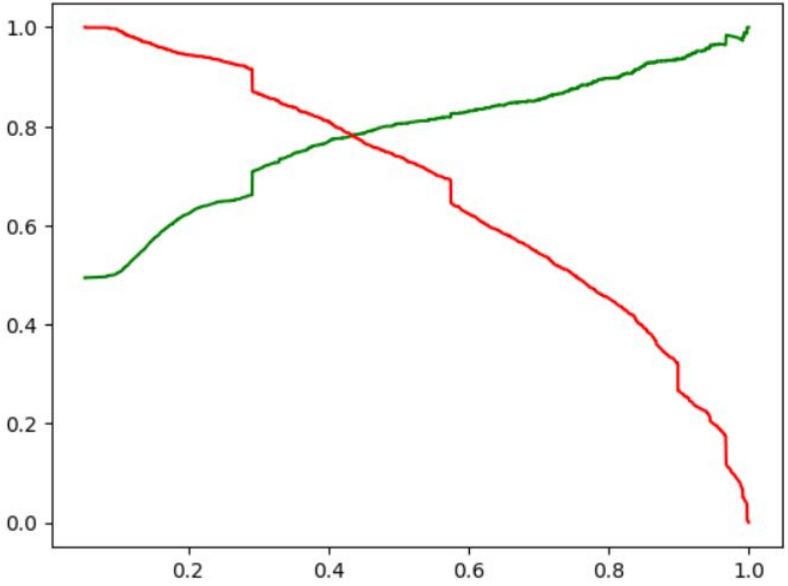
Step 4: Model Evaluation

Finding Optimal Cutoff Point



From the curve above, 0.4 is the optimum point to take it as a cutoff probability.

plotting a trade-off curve between precision and recall

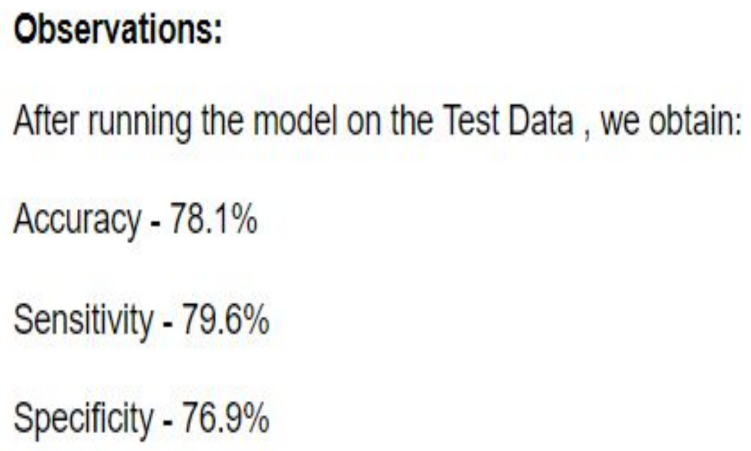


Step 5: Making Predictions on the Test Set

-> In this step we will be doing

- Scale the test set as well using just 'transform'
- Make predictions on the test set and store it in the variable 'y_test_pred'
- Make predictions on the test set using 0.4 as the cutoff

Step 5: Making Predictions on the Test Set



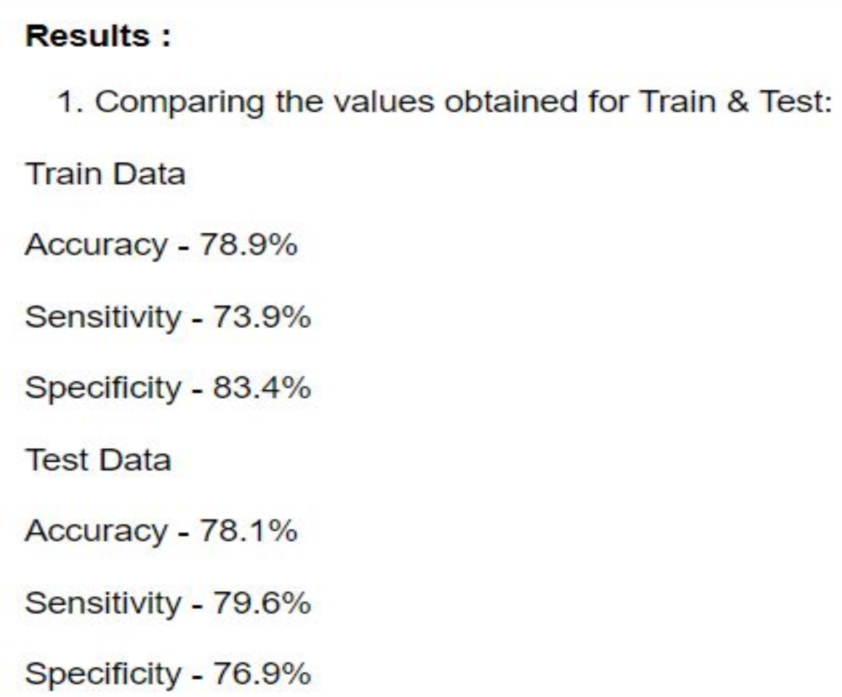
Observations:

After running the model on the Test Data , we obtain:

Accuracy - 78.1%

Sensitivity - 79.6%

Specificity - 76.9%



Results :

1. Comparing the values obtained for Train & Test:

Train Data

Accuracy - 78.9%

Sensitivity - 73.9%

Specificity - 83.4%

Test Data

Accuracy - 78.1%

Sensitivity - 79.6%

Specificity - 76.9%

Summary

- Thus we have achieved our goal of getting a ballpark of the target lead conversion rate to be around 77% . The Model seems to predict the Conversion Rate very well and we should be able to give the CEO confidence in making good calls based on this model to get a higher lead conversion rate of 77%.

- Finding out the leads which should be contacted:

  The customers which should be contacted are the customers whose "Lead Score" is equal to or greater than 77. They can be termed as 'Hot Leads'.

THANK YOU