LEAD SCORING CASE STUDY

PRESENTED BY

SUSHANT SARSWAT (DS C-38)

DEEPAK NAIR (DS C-38)

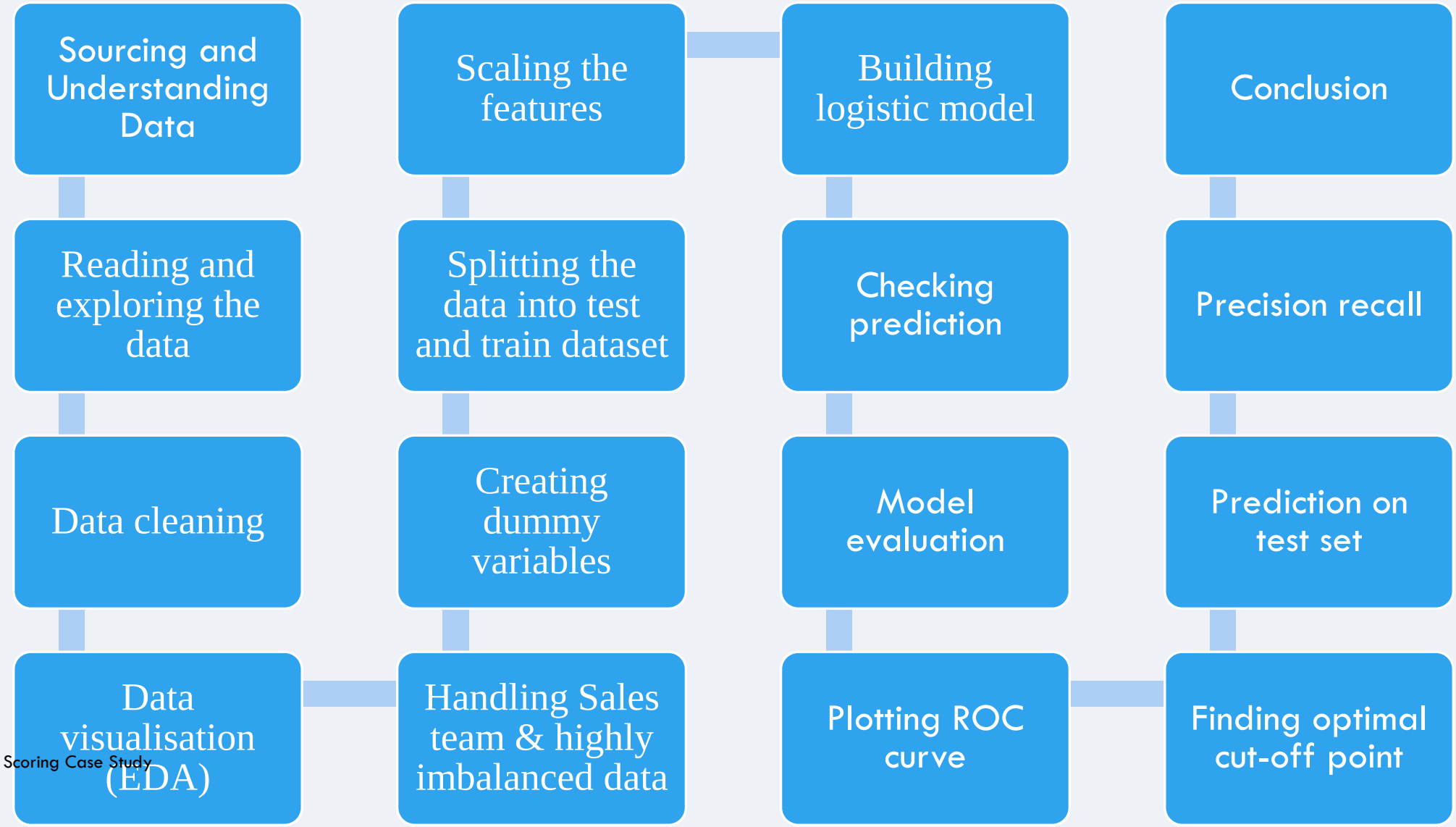
PROBLEM STATEMENT

- An education company X Education sells online courses to industry professionals.
- The company wishes to make the process more efficient by making use of a ML model in order to assign a lead score to each of the leads such that the customers with higher lead score have a higher conversion chance and the customers with lower lead score have a lower conversion chance.
- 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 most likely to convert whereas a lower score would mean that the lead will mostly not get converted.
- Our model should be able to adjust to if the company's requirement changes in the future.

APPROACH FOR ANALYSIS



SOURCING AND UNDERSTANDING DATA

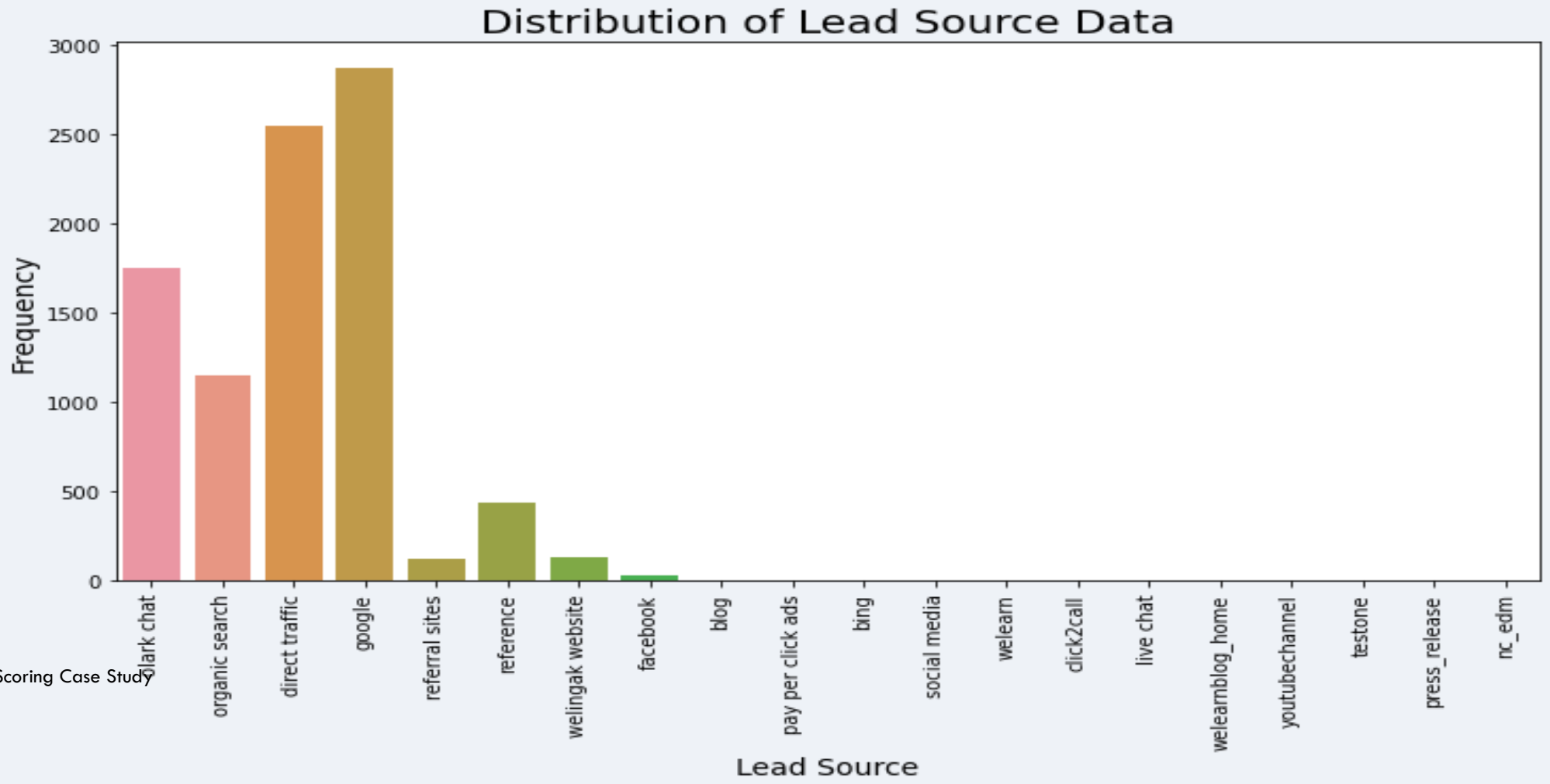
- Importing the given dataset
- Converting the dataset into a dataframe
- Understanding the data dictionary

READING AND EXPLORING THE DATA

- Shape of dataframe: 9240 rows and 37 columns
- Checking for missing values using info()
- Confirming outliers with describe()
- Checking for duplicates using duplicated().sum()

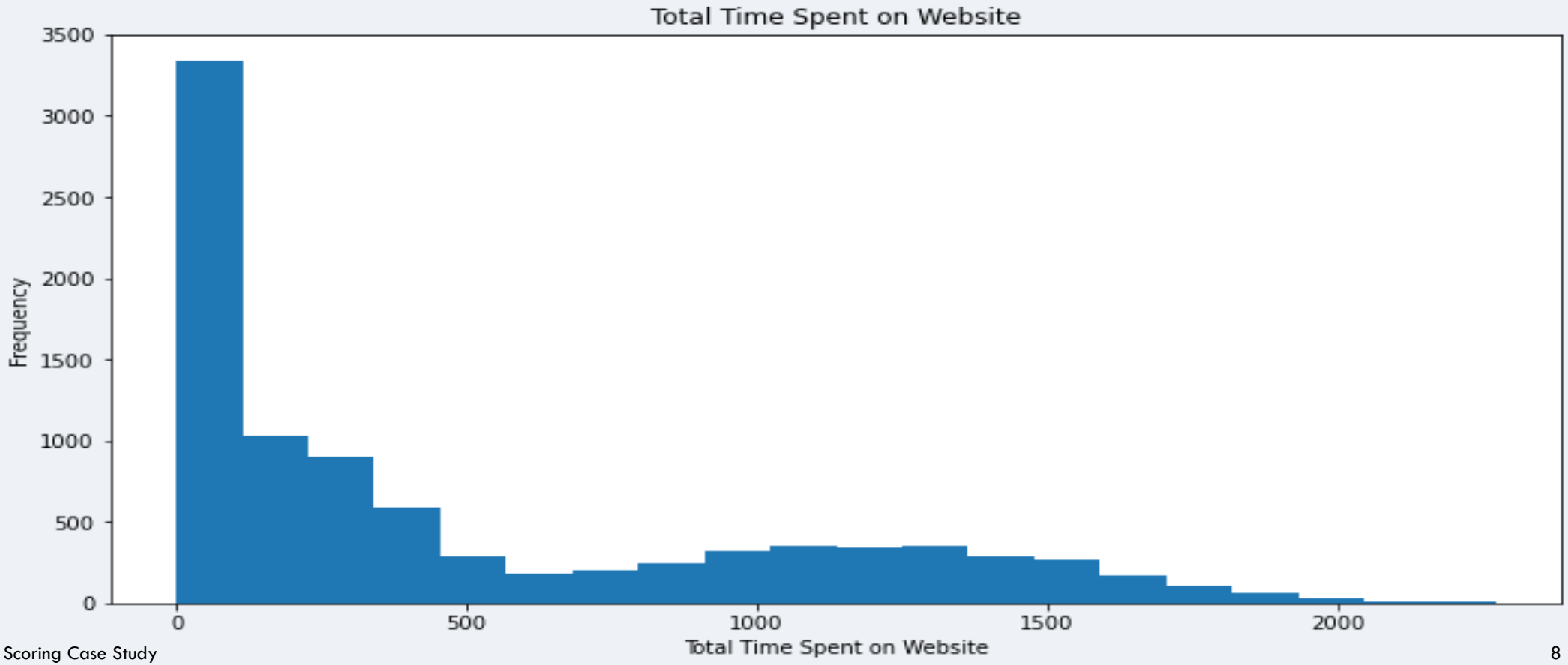
DATA VISUALISATION (EDA)

- Univariate analysis - Categorical variables



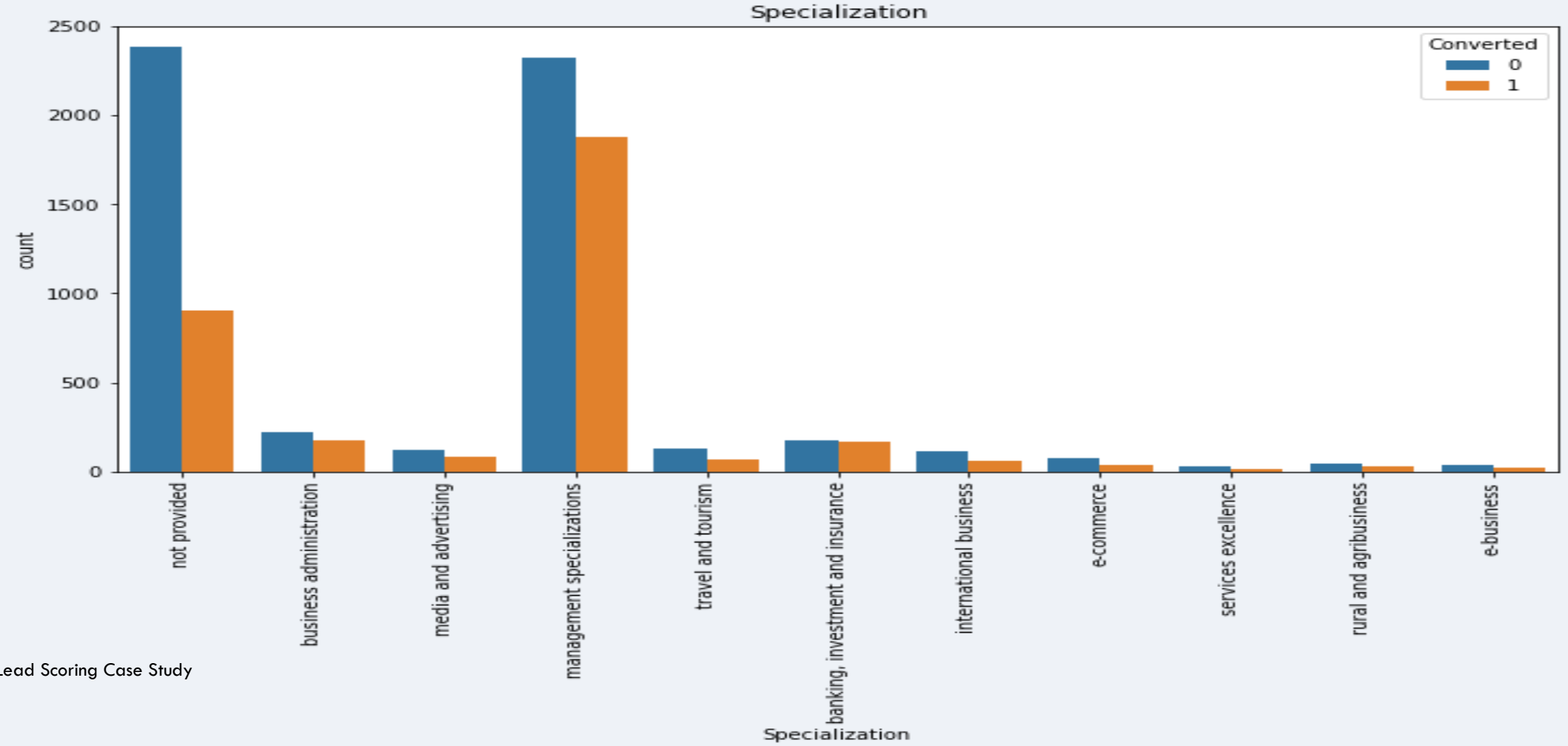
DATA VISUALISATION (EDA)

- Univariate analysis - Numerical variables



DATA VISUALISATION (EDA)

- Analysis of categorical variables with respect to dependent variable



DATA VISUALISATION (EDA)

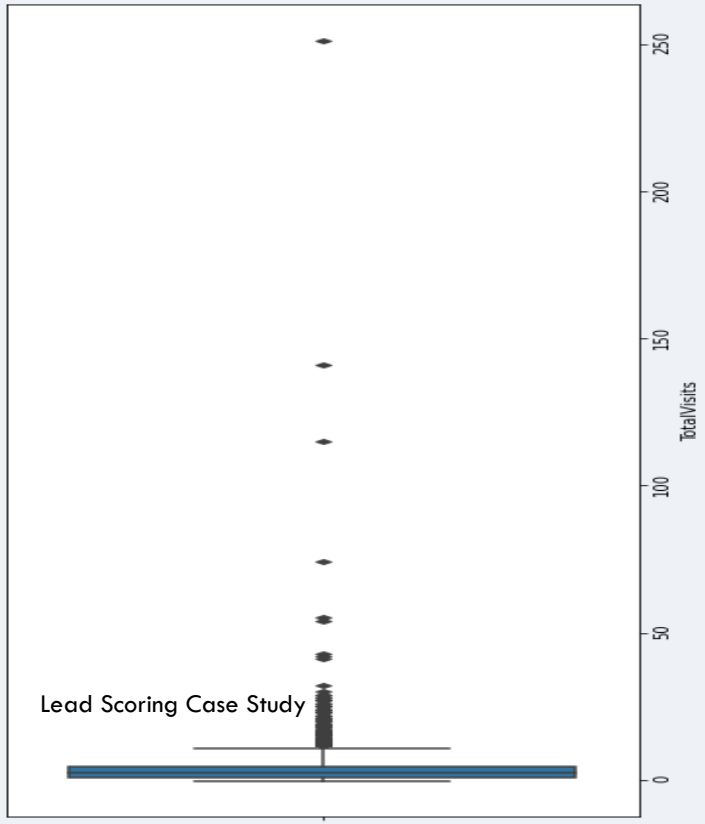
Correlation heatmap

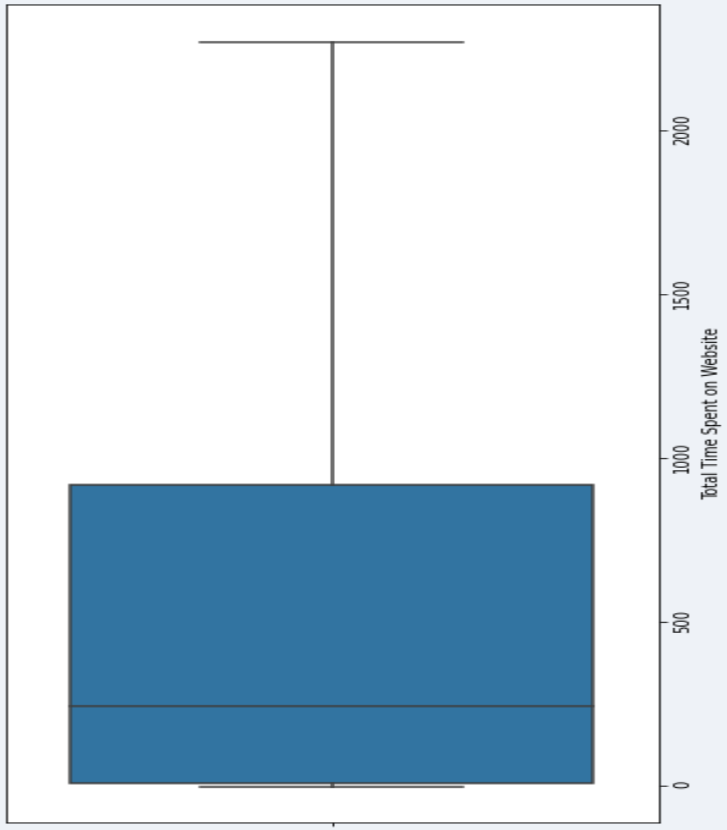
- No significant collinearity is observed between the variables

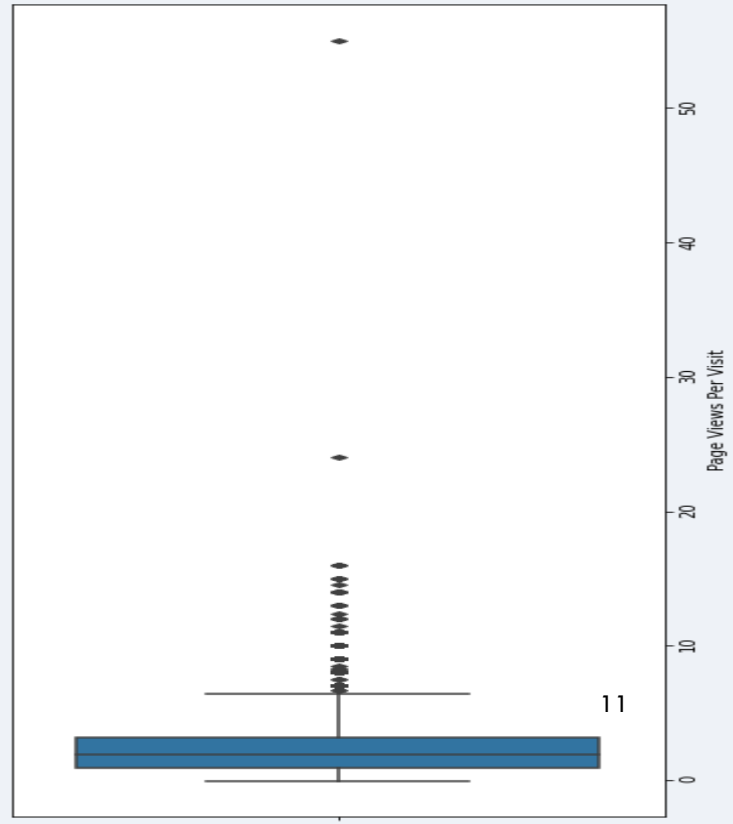
DATA VISUALISATION (EDA)

Outlier checks

- Outlier checks on total visits, total time spent on website and page views per visit:

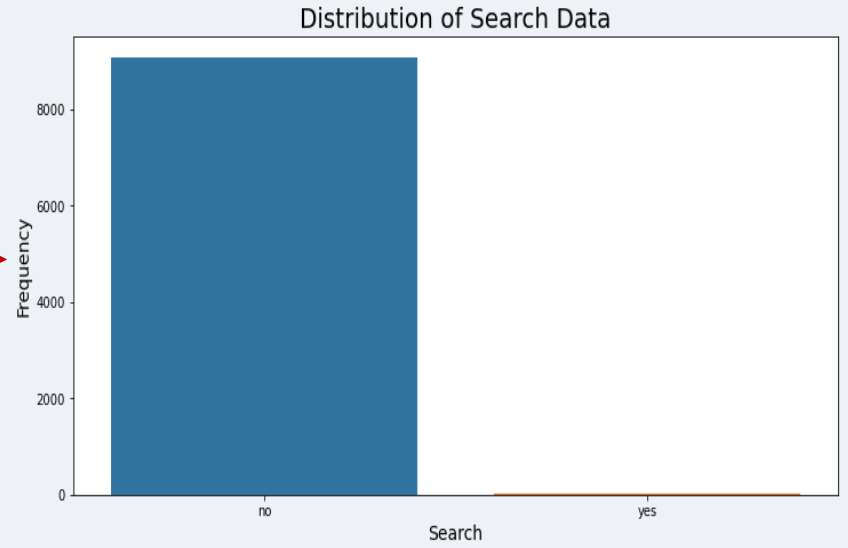






HANDLING SALES TEAM & HIGHLY IMBALANCED DATA

- Dropping sales team generated columns:
  - Tags
  - Last activity
  - Last notable activity
- Dropping columns which have highly imbalanced data:
  - Do not call
  - Country
  - What matters most to you in choosing a course
  - Search
  - Newspaper article
  - X education forums
  - Newspaper
  - Digital advertisement
  - Through recommendations



CREATING DUMMY VARIABLES

- Create dummy variables using the 'get_dummies' for categorical columns:
  - Lead origin
  - Lead source
  - Do not email
  - Specialization
  - What is your current occupation
  - City
  - A free copy of mastering the interview

SPLITTING THE DATA

- Separating the dependent target variable column
- Splitting data into 70% for train and 30% for test
- Random state is assigned as 100 (random_state=100)

SCALING THE FEATURES

- Scaling three numeric features for efficient processing and better comprehension using MinMaxscaler()
  - Totalvisits
  - Total time spent on website
  - Page views per visit

```
In [116]: # Verifying scaling
          X_test
```

Out[116]:

	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_blog	Lead Source_click2call	Lead Source_direct traffic	Lead Source_facebook	...	What occupation_
3271	0.235294	0.069102	0.444444	0	0	0	0	0	0	0	...	
1490	0.294118	0.665933	0.555556	1	0	0	0	0	1	0	...	
7936	0.117647	0.032570	0.222222	0	0	0	0	0	0	0	...	
4216	0.000000	0.000000	0.000000	0	1	0	0	0	0	0	...	
3830	0.470588	0.072183	0.888889	1	0	0	0	0	0	0	...	
...	...	...	...	...	...	...	...	...	...	...	...	
850	0.176471	0.364877	0.166667	1	0	0	0	0	0	0	...	
2879	0.117647	0.259243	0.222222	1	0	0	0	0	0	0	...	
6501	0.470588	0.587588	0.888889	1	0	0	0	0	1	0	...	
7155	0.176471	0.226673	0.333333	1	0	0	0	0	1	0	...	
376	0.235294	0.163732	0.444444	1	0	0	0	0	0	0	...	

BUILDING LOGISTIC MODEL

- Use RFE for feature selection
- Running RFE for 15 variables as output
- Building model by removing variables where VIF > 5 and p value > 0.05
- Fitting the model
- Checking Statistics

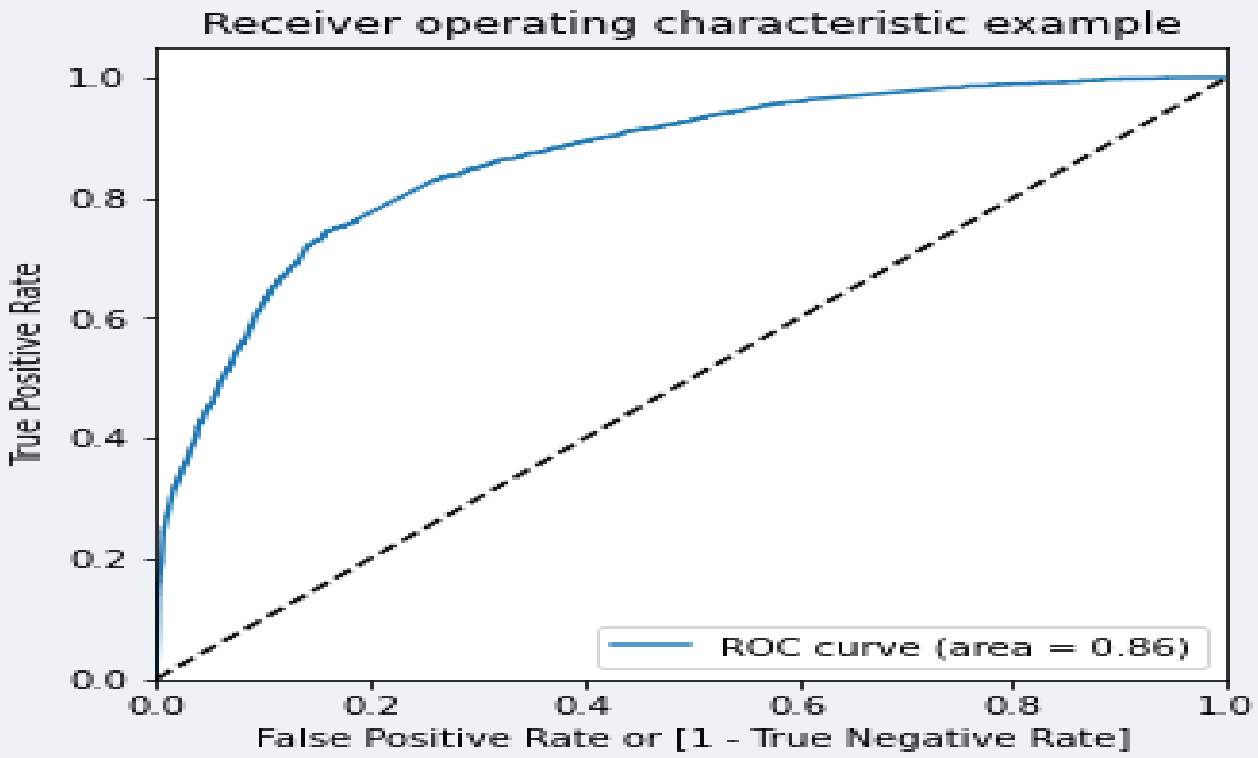
CHECKING PREDICTIONS

- Predict probabilities on Train set
- Reshaping y_train_pred
- Creating DataFrame with actual converted and predicted probabilities
- Creating new column 'Predicted' with 1 if Converted_Prob>0.5 else 0

MODEL EVALUATION

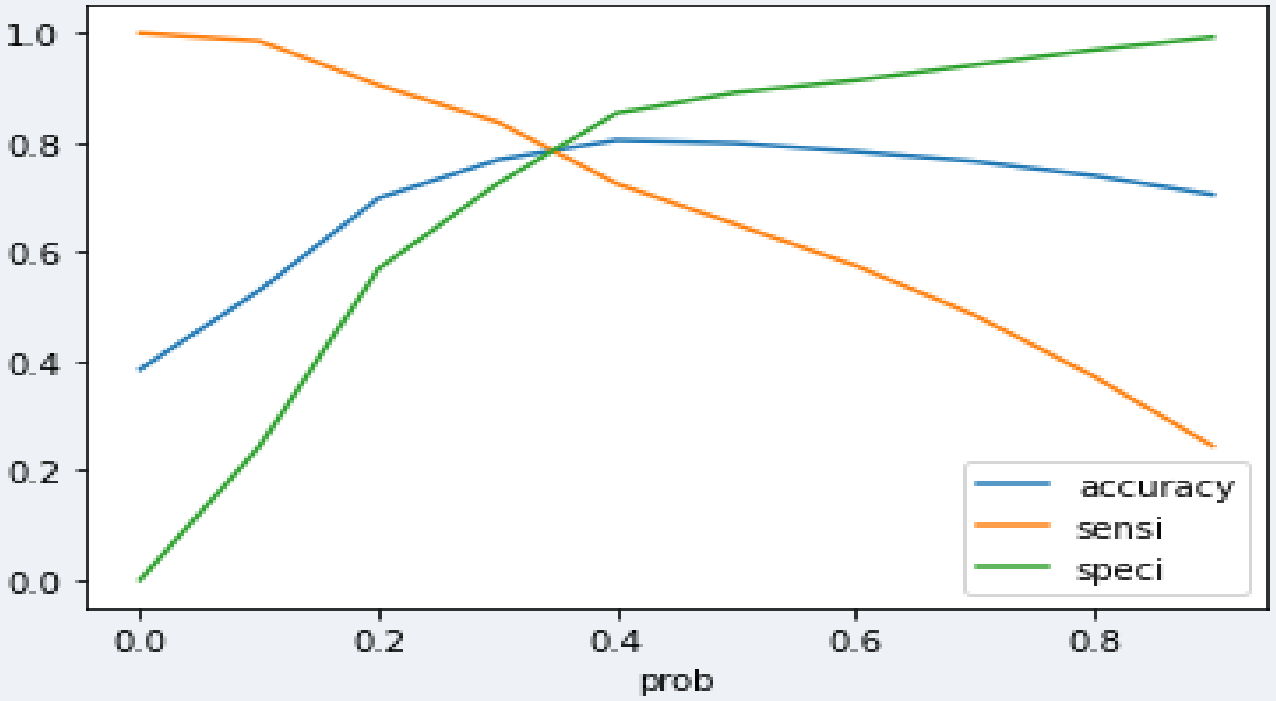
- Creating confusion matrix
- Checking overall accuracy
- Calculating Sensitivity and Specificity
- With a cutoff as 0.5, we have accuracy of 79.83%, Sensitivity as 65.00% and Specificity of the model as 89.12%

PLOTTING ROC CURVE



Area under the ROC curve is 0.86 which is very good

FINDING OPTIMAL CUTOFF POINT



From the graph, we observe that the optimum cutoff is at 0.32

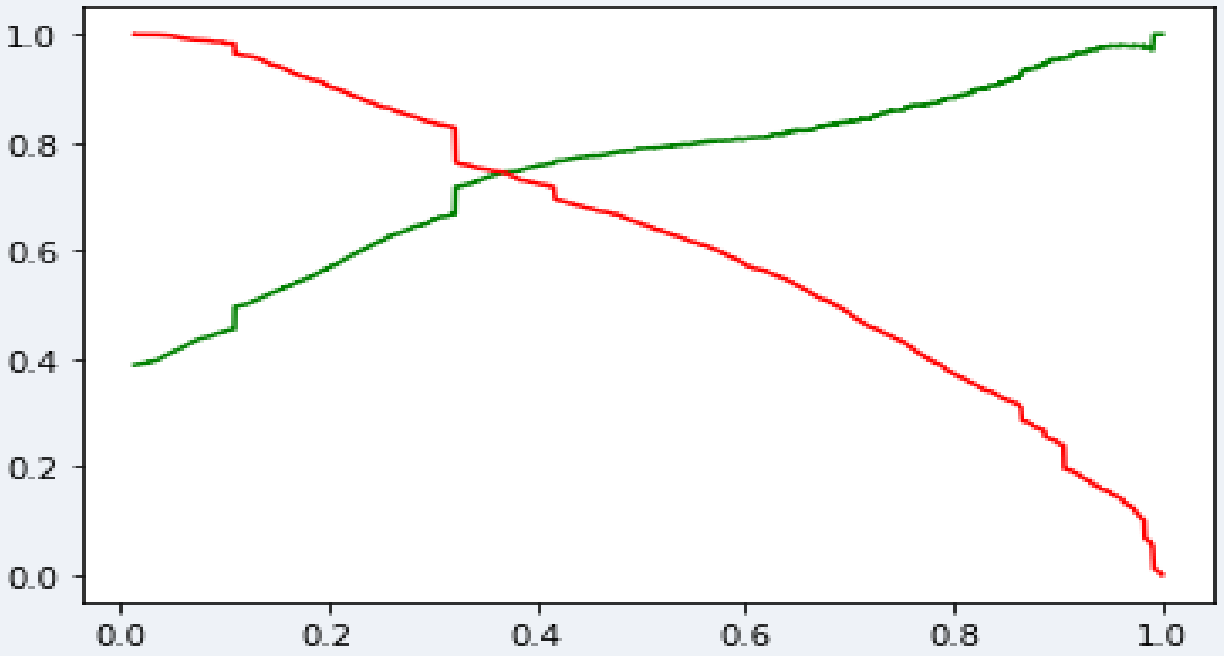
Train Data

1. Accuracy ➡ 77.47%
2. Sensitivity ➡ 82.79%
3. Specificity ➡ 74.14%

Test Data

1. Accuracy ➡ 77.08%
2. Sensitivity ➡ 81.19%
3. Specificity ➡ 74.74%

PRECISION RECALL



From the graph, the optimum cutoff is 0.38

Train Data

1. Accuracy ➡ 80.23%
2. Precision ➡ 74.74%
3. Recall ➡ 73.55%

Test Data

1. Accuracy ➡ 80.43%
2. Precision ➡ 73.75%
3. Recall ➡ 71.59%

CONCLUSION

- Top three variables in our model which contribute most towards the probability of a lead getting converted are as follows:
  - Totalvisits
  - Lead source_google
  - Total time spent on website

- Top 3 categorical/dummy variables in the model which should be focused the most on in order to increase the probability of lead conversion are as follows:
  - Lead Source_welingak website
  - Lead Source_reference
  - What is your current occupation_working professional

- Following personnel are most likely to convert:
  - Whose last activity was through SMS or Olark chat conversation.
  - Who has a management specialization
  - Who are working professionals
  - Who are visiting website repeatedly
  - Who are spending a lot of time on the website