Lead scoring case study

By
Pradeep Rajendran,
Shubhangi Dubey

Problem statement:

- X Education sells online courses to the industry professionals. The company getting the details of the leads through fill up forms provided in the website or through referrals. This will be given to the sales team to make calls, sending emails and SMS etc.
- Even though the company getting lot of leads, the lead conversion rate from the lead to paying customer is poor and the conversion rate is around 30%.

Objective:

- A dataset given with the details of leads and the conversion status from the past.
- Considering the technical and business objective build a logistic model to assign a score to each lead to find the promising leads.
- Identify the key driver variables and the strong indicators to find the hot leads.
- Model need to predict who are all hot leads (higher chance of conversion) and cold leads (low chance for conversion). So, the sales team will focus spending time on hot leads and which is increase the conversion rate.

Importing the libraries and datasets:

- Importing the libraries:
  - Numpy
  - Pandas
  - Matplotlib.pyplot
  - Seaborn
  - Statsmodels
  - Sklearn
- Reading the dataset ‘Leads.csv file’ from the local directory to python data frame.
- Checking the dataset’s shape and data imported are complete.
- Leads.csv file is having 9240 rows and 37 columns.

Data cleaning:

- **Case mismatch:**
  - Converting categorical labels into lower case to remove the duplication due to case mismatch.
- **Categorical label as ‘select’:**
  - Some columns have the categorical label as ‘select’. In the submission form, there might be some drop downs and they might have default value as select. If the user not selecting any value, those columns will have the value as ‘select’.
  - Converting this value ‘select’ to ‘Nan’ to handle them as missing values.
- **Highly skewed columns or columns with one value:**
  - **Highly skewed columns:** 'Do Not Call’, 'Search', 'X Education Forums', 'Newspaper Article', 'Newspaper’, 'Digital Advertisement', 'Through Recommendations’.
  - **Columns with one label:** 'Update me on Supply Chain Content’, 'Magazine’, 'Receive More Updates About Our Courses', 'Get updates on DM Content', 'I agree to pay the amount through cheque’
  - Dropping these columns which are highly skewed with one label or the columns which are having only one label. These will not be helpful for the analysis and building the model.

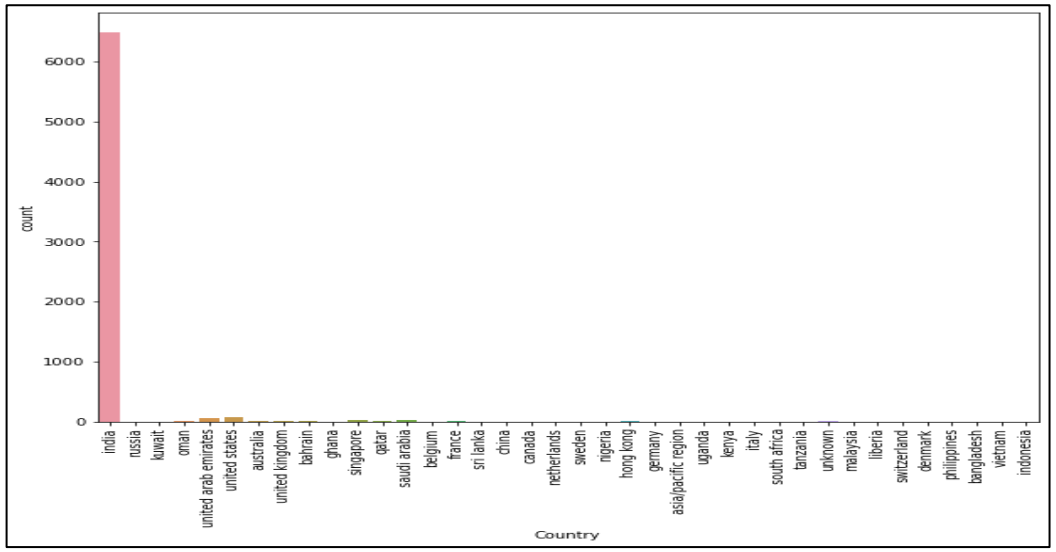
Data cleaning – Handling missing values:

- **Row wise missing values:**
  - There is no column which is having missing value more than 70%. We can keep the dataset.
- **Column wise missing values:**
  - **Missing values >40%:**
    - The columns 'How did you hear about X Education', 'Lead Quality', 'Lead Profile', 'Asymmetrique Activity Index', 'Asymmetrique Profile Index', 'Asymmetrique Activity Score' and 'Asymmetrique Profile Score' are having the missing values more than 40%.
    - Removing these columns to build the model.
  - **Missing values < 2%:**
    - The columns 'Lead Source', 'TotalVisits', 'Page Views Per Visit' and 'Last Activity' are having missing values less than 2%. So, we can remove those missing rows from the dataset.
  - **Balance columns with missing values:**
    - 'City', 'Tags', 'Specialization', 'What matters most to you in choosing a course', 'What is your current occupation', 'Country'
    - Still we have 6 columns which are having missing values in the dataset. Let's validate each column and handle the missing values.

Data cleaning – Handling missing values:

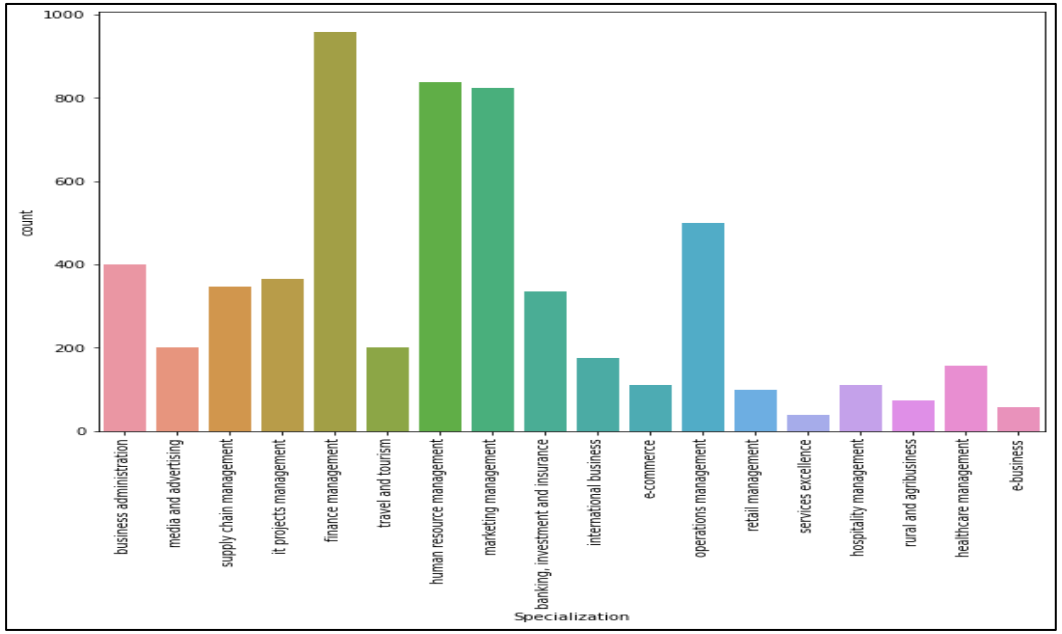
- **Column – ‘City’:**
  - The column ‘City’ is having missing value of 39.4%.
  - Replace the missing values in the column 'City' as 'unknown'.
  - Also, we can change the categorical labels 'Other cities', 'other cities of maharashtra','other metro cities' and 'tier ii cities’ as 'others’ to convert them into a single category.

- **Column – ‘Country’:**
  - In the column 'Country’, categorical label 'India’ is having more counts and the data is highly skewed. So, we can drop this column.

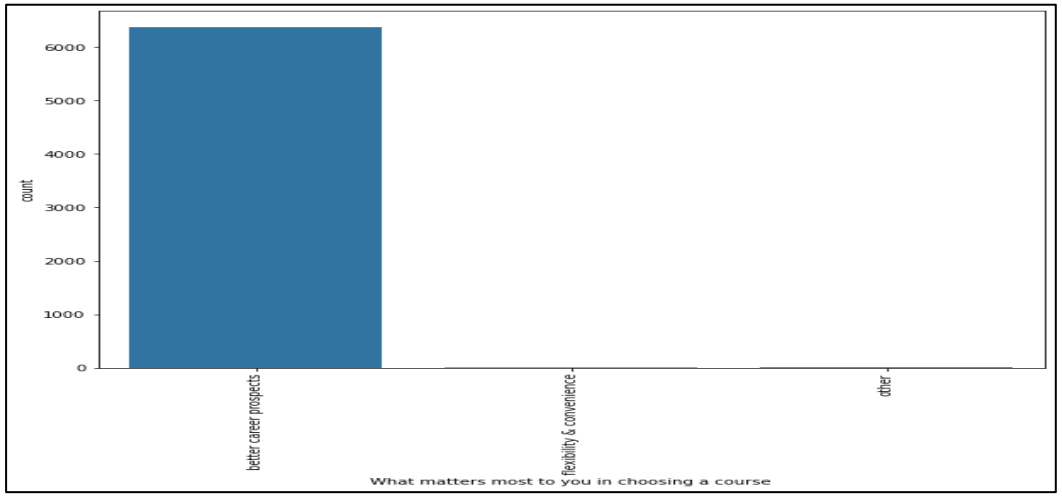


Data cleaning – Handling missing values:

- **Column - 'Specialization’:**
  - In this column, we can see that all labels are having decent counts.
  - We cannot update the missing value as the categorical label which is having higher count. Which will impact the model prediction.
  - So, we can replace missing values as 'other’.

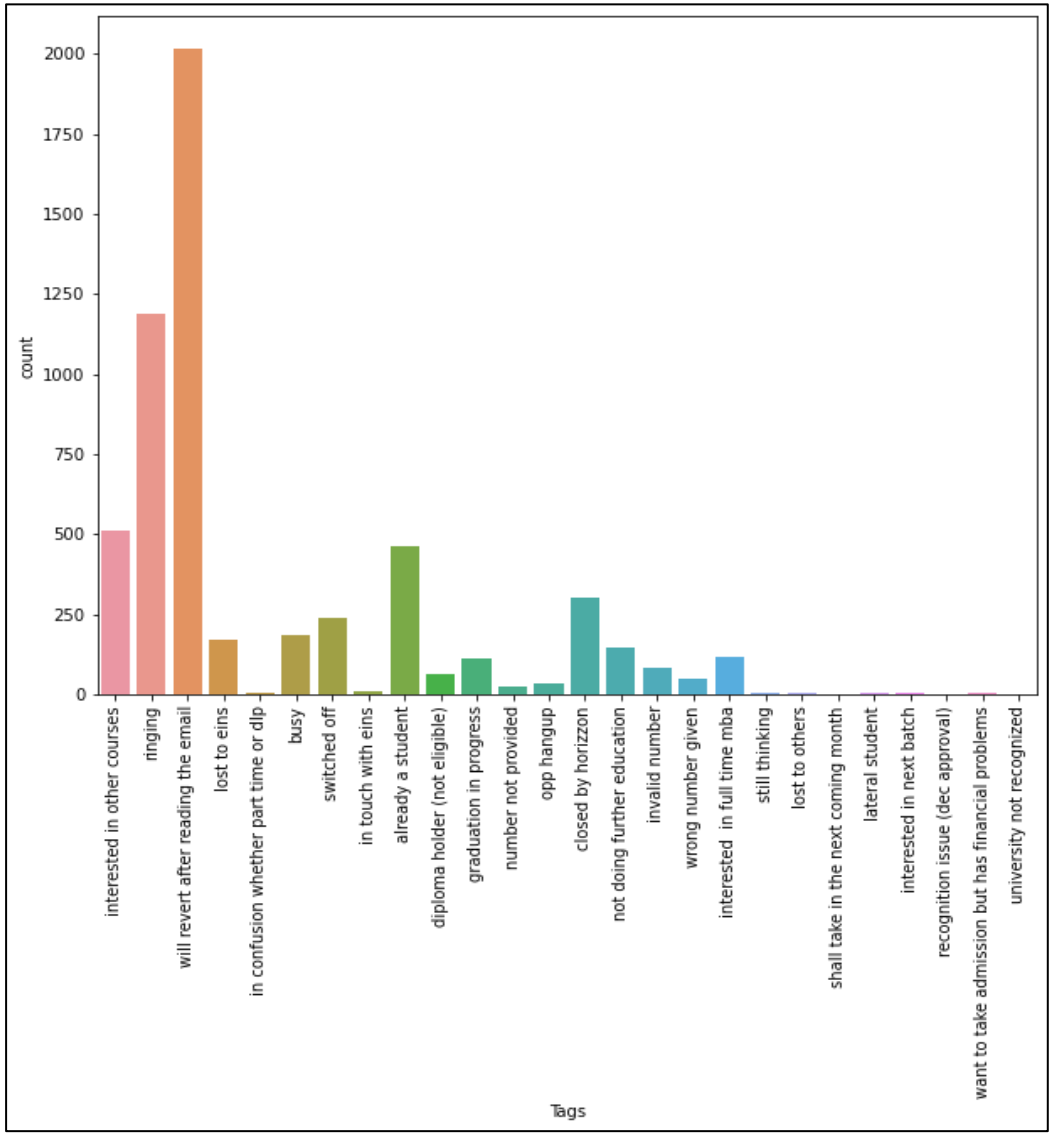


- **Column - 'What matters most to you in choosing a course':**
  - Data in the column 'What matters most to you in choosing a course' is highly skewed. So, we can remove this column from the dataset.



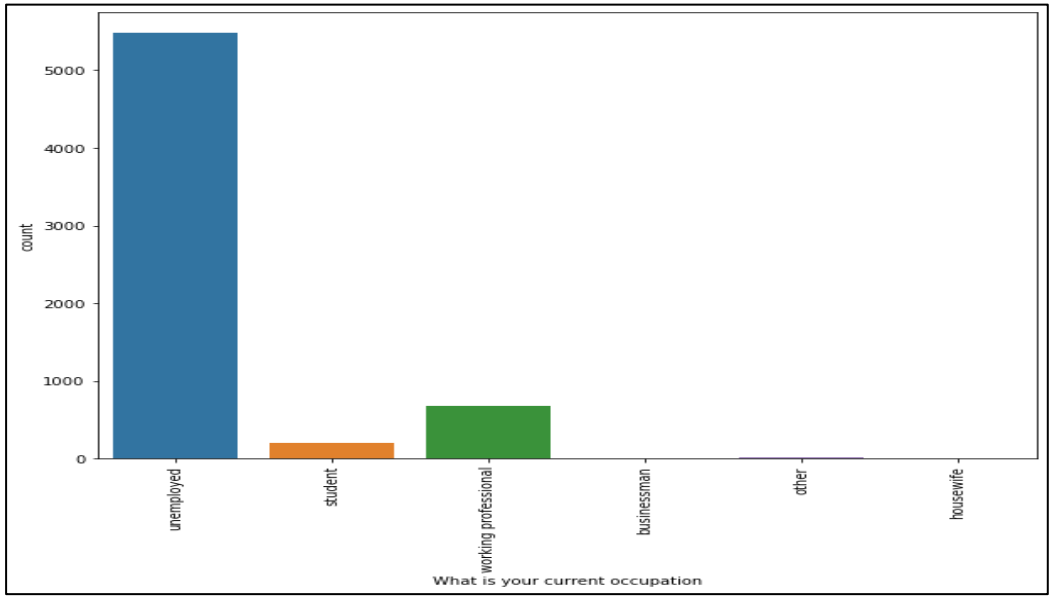
Data cleaning – Handling missing values:

- **Column - 'Tags’:**
  - In this column, we can see that spread is across all labels.
  - So, we can impute the missing value as 'unknown’.
  - Also, merge the labels which are having low counts into a category 'others' to avoid categorical labels having high number of categorical labels with very low frequency.
  - Replace the values which are having count less than 200.



Data cleaning – Handling missing values:

- **Column - 'What is your current occupation':**
  - In this column, we have 29% of missing data.
  - So, we can impute the missing value as 'other'.



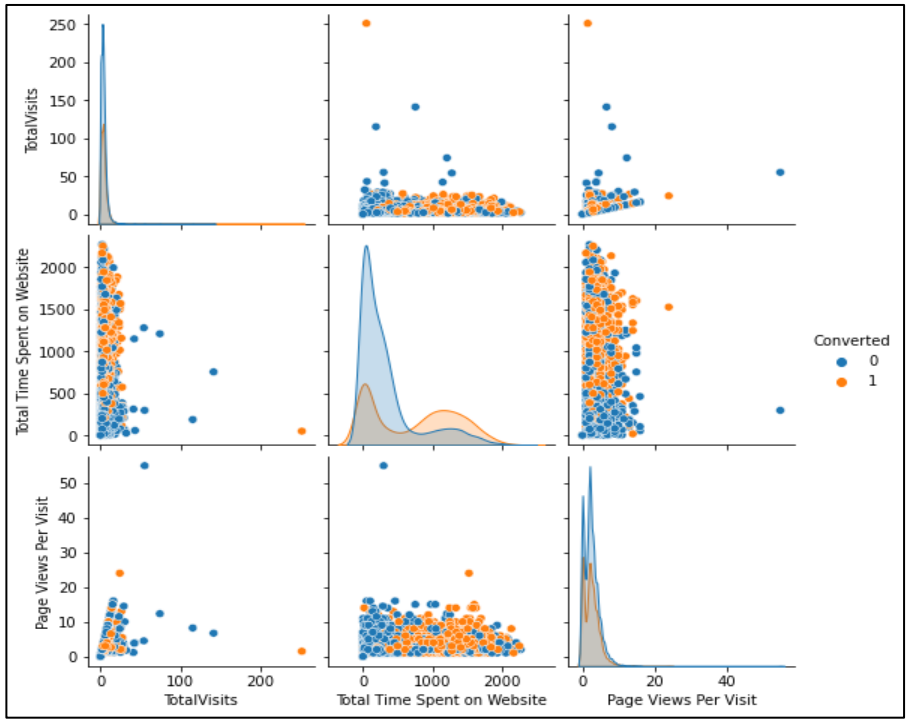
- **Column - 'Prospect ID' and 'Lead Number':**
  - The columns 'Prospect ID' and 'Lead Number' are unique for each rows.
  - So, we can remove these columns from our dataset which are not giving any insights for analysis.

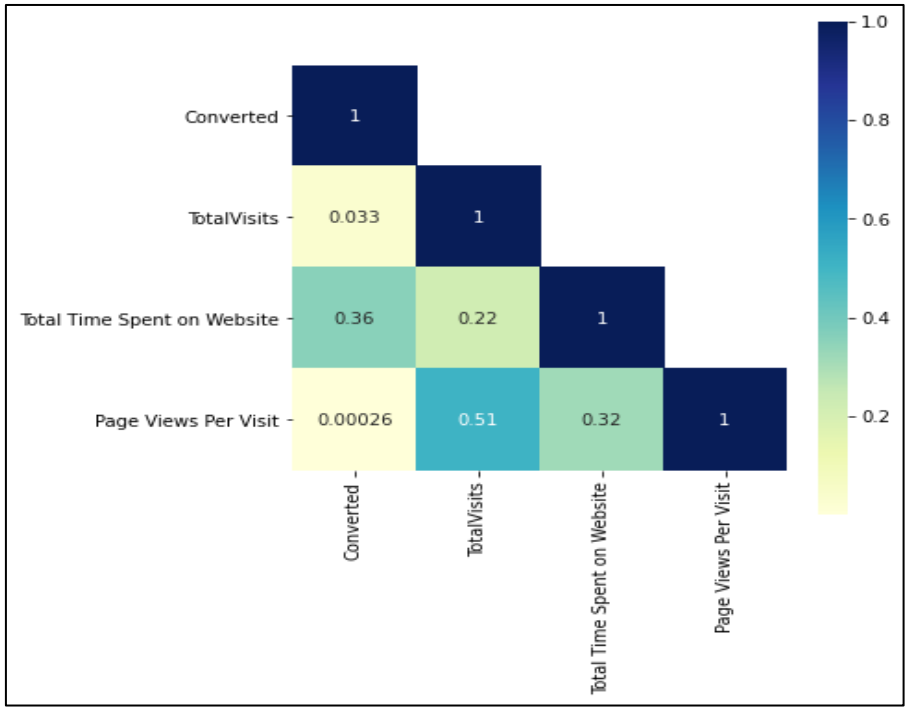
EDA:

- **Remove sales team’s data:**
  - In the current dataset, we have the data which are provided by the sales team.
  - We are building model to predict the hot leads or cold leads and provide them to the sales team to improve the conversion.
  - For model building, we need to use the details which are provided by the leads.
  - So, we are removing the columns that are filled by the sales team.
  - **Columns removed:**
    - 'Last Activity’
    - 'Tags’
    - 'Last Notable Activity'

EDA:

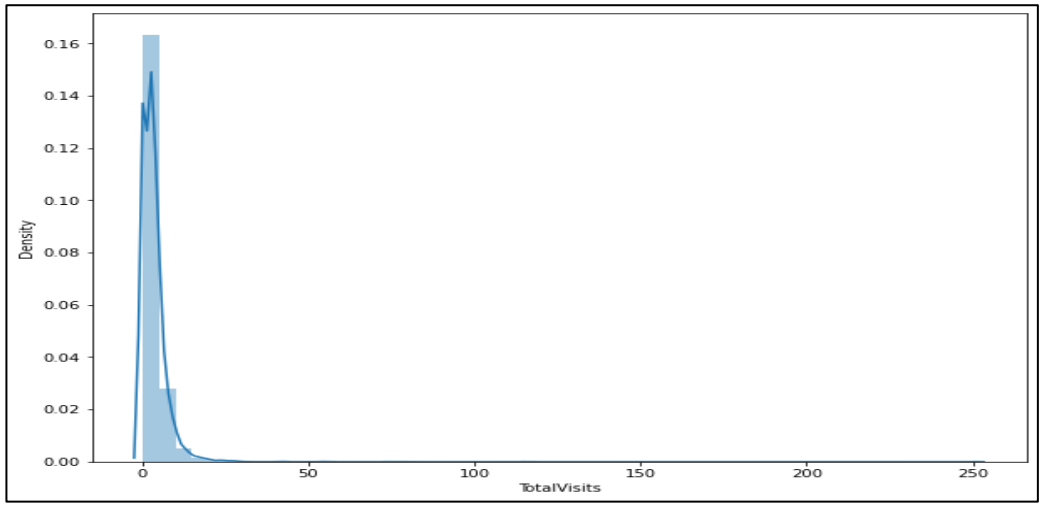
- **Pairplot and heatmap:**
  - There is a strong correlation between the columns 'TotalVisits' and 'Page Views Per Visit'
  - As per the below plots, we could see that the person who is spending more time on the website and high number of visits having more probability of conversion.



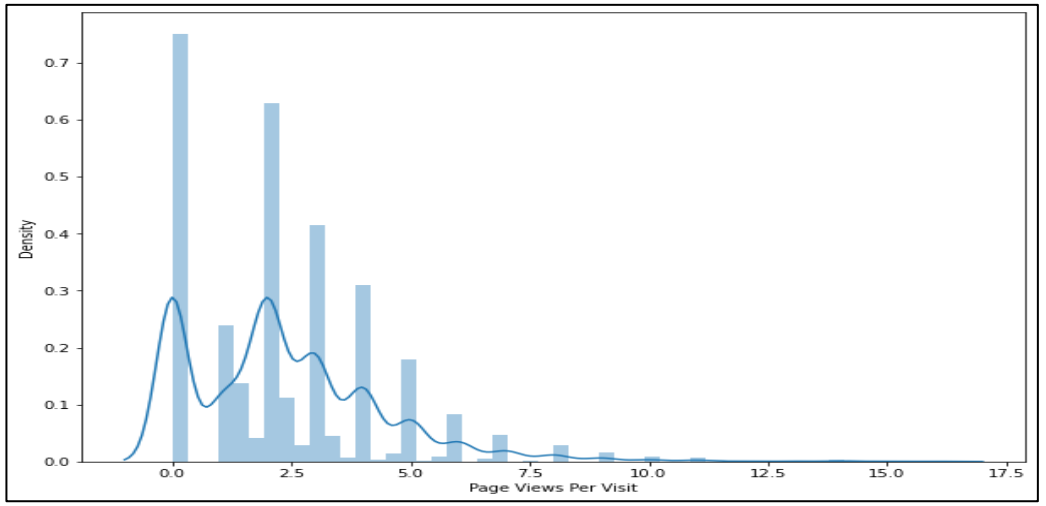


EDA:

- **Handling outliers:**
  - **Column – 'TotalVisits':**
    - Max value in this value is 251 and the 99th percentile value is 17.
    - So, we can remove the rows which are having value more than 99th percentile.

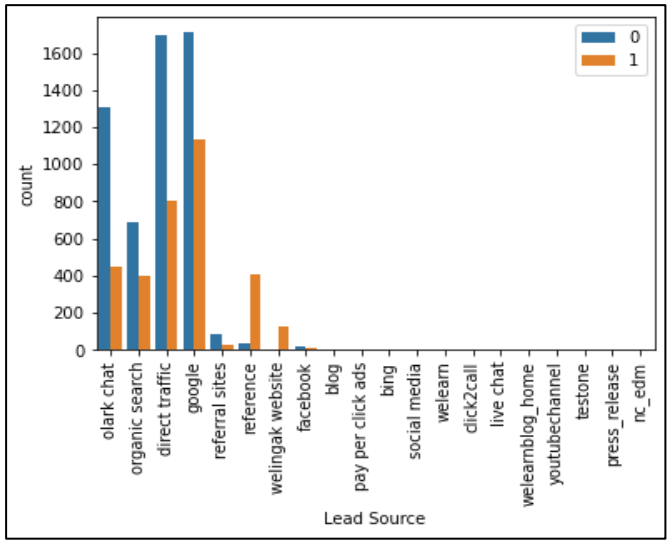


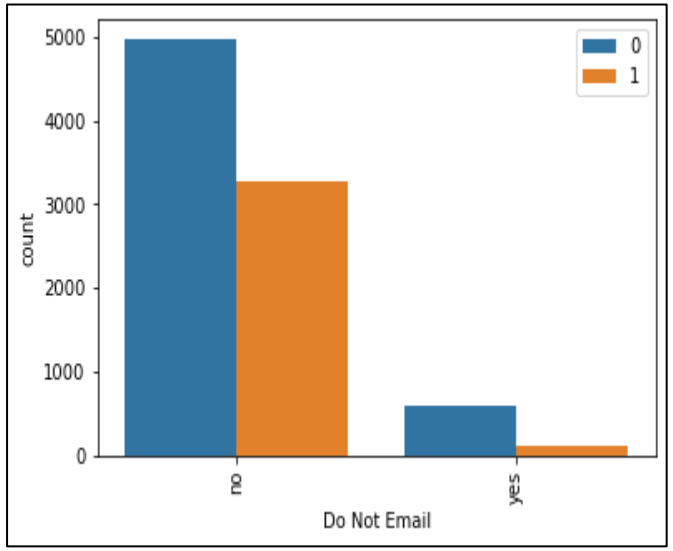
  - **Column- 'Page Views Per Visit':**
    - Max value in this value is 16 and the 99th percentile value is 9.
    - So, we can remove the rows which are having value more than 99th percentile.

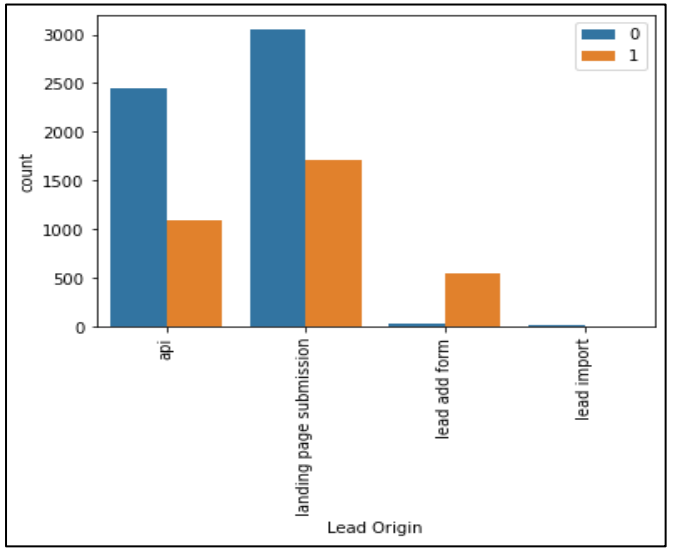


EDA:

- **Categorical columns:**
  - Lead source through google, direct traffic and reference are having higher conversion rate.
  - Most of the students not preferred to get the emails.
  - Most of the leads are originated from 'API' and 'Landing page submission'

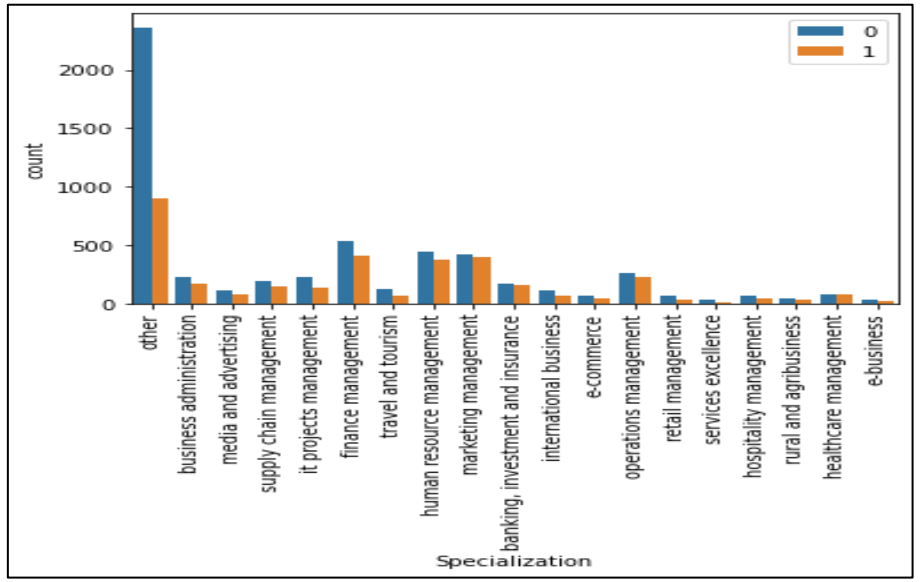


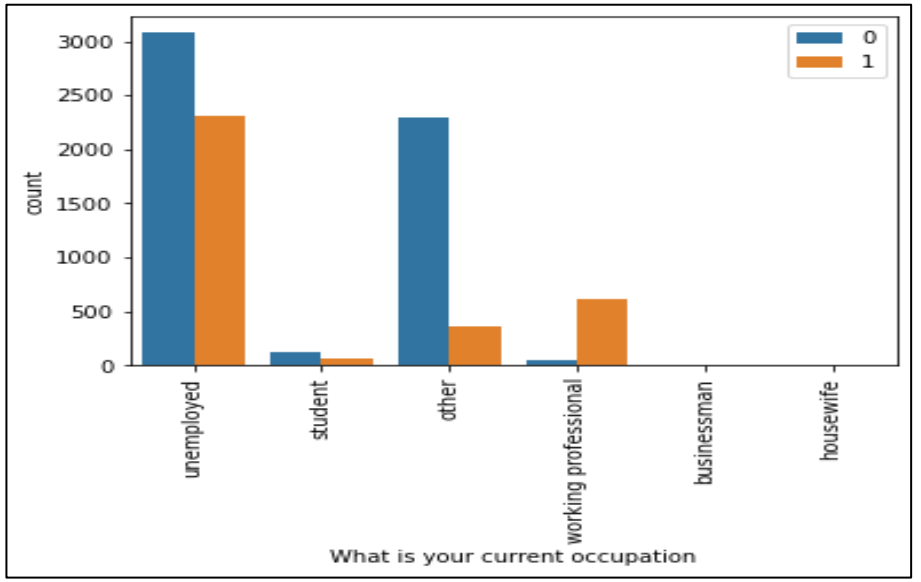




EDA:

- **Categorical columns:**
  - Leads who are having specialization as 'finance management', 'human resource management', 'marketing management' and 'operations management' are having higher percentage of conversion rate.
  - Working professionals are having higher percentage of conversion. Unemployed leads are having higher count in the conversion.





Data retained:

- After data cleaning and EDA, we have the total 8924 rows in our dataset.
- In the raw dataset, we had 9240 rows.
- So, we retained 96.58% of data from the original dataset for our model building

Data preparation for modeling :

- **Binary columns:**
  - Columns 'Do Not Email' and 'A free copy of Mastering The Interview’ are having the values as ‘yes’ and ‘no’.
  - So, we can convert them it to binary values 1 and 0.

- **Dummy variables:**
  - Columns 'City', 'What is your current occupation', 'Specialization', 'Lead Source’ and 'Lead Origin’ are having the categorical labels.
  - We can create dummy variables using the pandas.get_dummies() function.
  - 53 dummy variables are created for the categorical columns and dropped the original categorical columns from the current dataset.

Data preparation for modeling :

- **Input and output variable split:**
  - The column ‘Converted’ is out target variable and stored this column in a pandas series name ‘y’.
  - Except the column ‘Converted’, all other columns are out input variables. So, we stored these columns in a data frame called ‘x’.

- **Training and test split:**
  - Using sklearn. train_test_split() function, we are splitting the dataset in to train and test dataset.
  - We split the dataset into 70% and 30% respectively for training and testing.
  - Stored the training and test dataset in x_train, x_test, y_train and y_test variables with the below code.

```
x_train,x_test,y_train,y_test = train_test_split(x,y,train_size=0.7,test_size=0.3,random_state=100)
```

Data preparation for modeling :

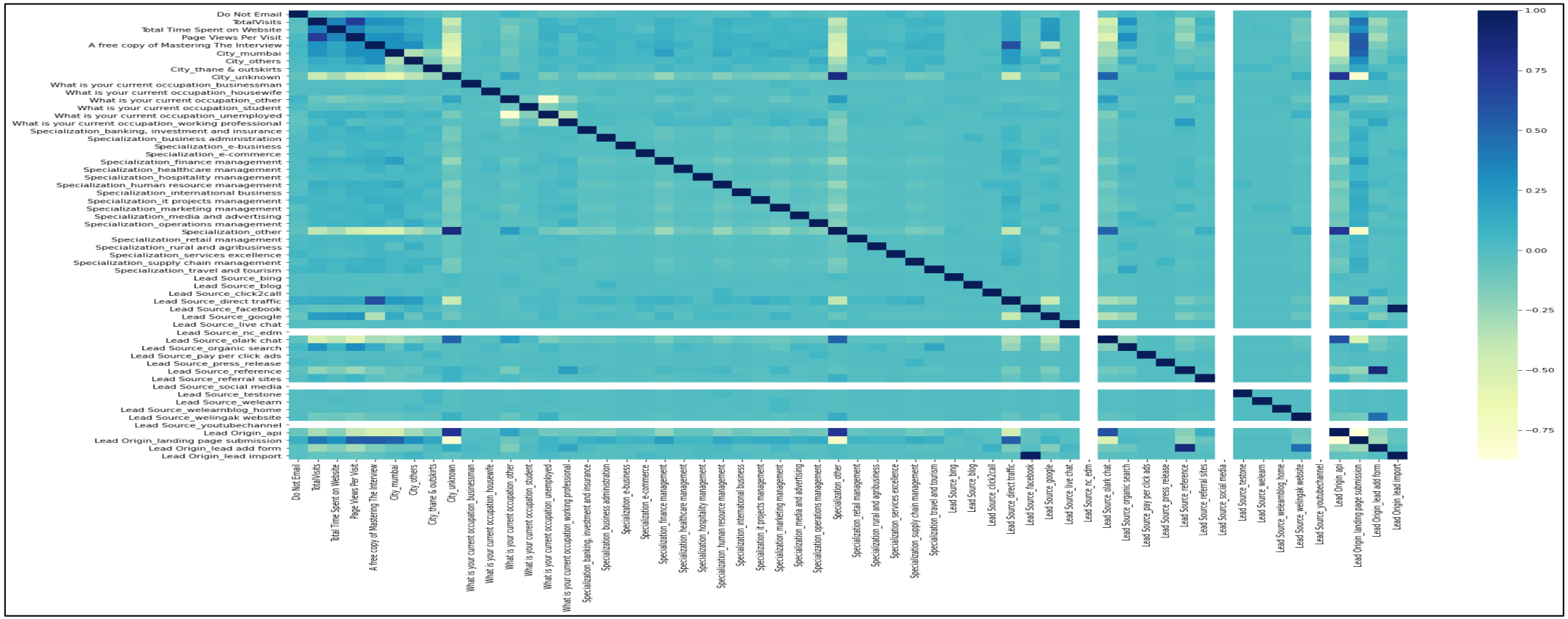
- **Scaling in training dataset:**

  - Change the scale of the columns 'TotalVisits','Total Time Spent on Website'and 'Page Views Per Visit' using the "MinMaxScaler"

```
scaler = MinMaxScaler()
scaler.fit_transform(x_train[['TotalVisits','Total Time Spent on Website','Page Views Per Visit']])
```

Data preparation for modeling :

- **Heatmap with correlation data:**
  - Heatmap plotted using the correlation between the variables of x_train dataset.
  - We could see that some variables are having high correlation between them.



Data preparation for modeling :

- **Collinearity:**
  - In the below table we could see that the features which are having high correlation between them.
  - In both traning and test datasets, drop the columns which are having correlation values more than 0.85 to reduce collinearity.

Variable 1	Variable 2	Correlation
Lead Source_facebook	Lead Origin_lead import	0.977941
Lead Origin_landing page submission	Lead Origin_api	0.872709
Lead Source_reference	Lead Origin_lead add form	0.869362
Specialization_other	City_unknown	0.854058
Lead Origin_landing page submission	City_unknown	0.817434
What is your current occupation_unemployed	What is your current occupation_other	0.805482
Lead Origin_api	City_unknown	0.771924

Model building:

- While selecting features for model building, we use both automated and manual approach.
- **Automated approach - RFE:**
  - We have 54 features in our dataset and we automated approach RFE (Recursive Feature Elimination) method to rank the features.
  - Used sklearn's LogisticRegression and RFE functions to build the logistic regression model recursively with the all feature and rank the features to select the top 20.
- **Manual approach – VIF and p-value:**
  - Used GLM (Generalized Linear Models) from statsmodels library to build the logistic regression model.
  - We can use the top 20 features which are obtained from RFE method in the manual approach of model building. Then we calculate the VIF (Variance inflation Factor) and p-values for each features.
  - Then drop the feature which is having high p-value or VIF value and build the model again till we reach all features are having VIF and p-value less than 5 and 0.05 respectively.

Final model - summary and VIIF values:

Generalized Linear Model Regression Results

Dep. Variable:	Converted	No. Observations:	6246
Model:	GLM	Df Residuals:	6231
Model Family:	Binomial	Df Model:	14
Link Function:	logit	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-2789.2
Date:	Wed, 12 Jan 2022	Deviance:	5578.3
Time:	13:46:39	Pearson chi2:	6.32e+03
No. Iterations:	7		
Covariance Type:	nonrobust		

	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.7509	0.120	-6.255	0.000	-0.986	-0.516
Do Not Email	-1.3483	0.172	-7.848	0.000	-1.685	-1.012
TotalVisits	0.9981	0.252	3.963	0.000	0.505	1.492
Total Time Spent on Website	4.5230	0.163	27.756	0.000	4.204	4.842
What is your current occupation_other	-1.2116	0.084	-14.475	0.000	-1.376	-1.048
What is your current occupation_working professional	2.3579	0.181	13.029	0.000	2.003	2.713
Specialization_hospitality management	-0.9221	0.331	-2.787	0.005	-1.571	-0.274
Specialization_other	-0.7973	0.119	-6.700	0.000	-1.031	-0.564
Lead Source_direct traffic	-1.3508	0.143	-9.419	0.000	-1.632	-1.070
Lead Source_google	-1.0307	0.126	-8.212	0.000	-1.277	-0.785
Lead Source_organic search	-1.1673	0.151	-7.740	0.000	-1.463	-0.872
Lead Source_referral sites	-1.6783	0.333	-5.037	0.000	-2.331	-1.025
Lead Source_welingak website	2.2064	0.754	2.926	0.003	0.728	3.684
Lead Origin_api	0.6384	0.126	5.073	0.000	0.392	0.885
Lead Origin_lead add form	3.1612	0.249	12.701	0.000	2.673	3.649

	Features	VIF
6	Specialization_other	4.74
12	Lead Origin_api	4.70
1	TotalVisits	3.64
8	Lead Source_google	2.74
2	Total Time Spent on Website	2.35
7	Lead Source_direct traffic	2.29
9	Lead Source_organic search	1.90
3	What is your current occupation_other	1.51
13	Lead Origin_lead add form	1.46
11	Lead Source_welingak website	1.35
4	What is your current occupation_working profes...	1.20
0	Do Not Email	1.11
10	Lead Source_referral sites	1.11
5	Specialization_hospitality management	1.02

Model evaluation:

- Predict the probability score using the training data (x_train).
- **Cutoff value – 0.5:**
  - Using the cutoff value as 0.5 and calculated the conversion rate from the predicted probability.
  - **Metrics scores:**

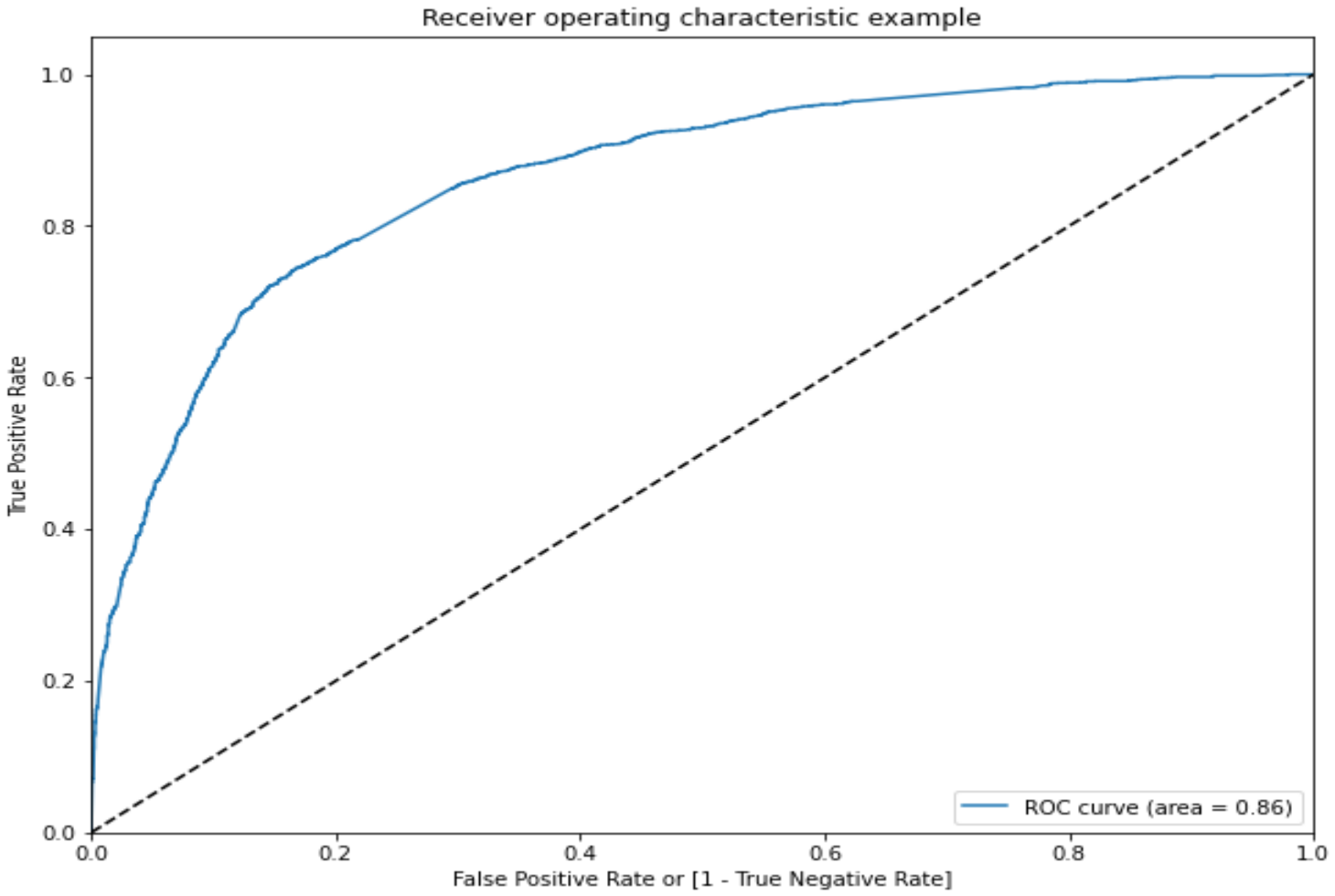
    Accuracy – 79.79%
  - **Confusion matrix:**

	Predicted No	Predicted Yes
Actual – No	3448	423
Actual - Yes	839	1536

  - **Sensitivity and Specificiy:**
    - Sensitivity = TP/ (TP+FN) = 64.67%
    - Specificity = TN/ (TN+FP) = 89.07%

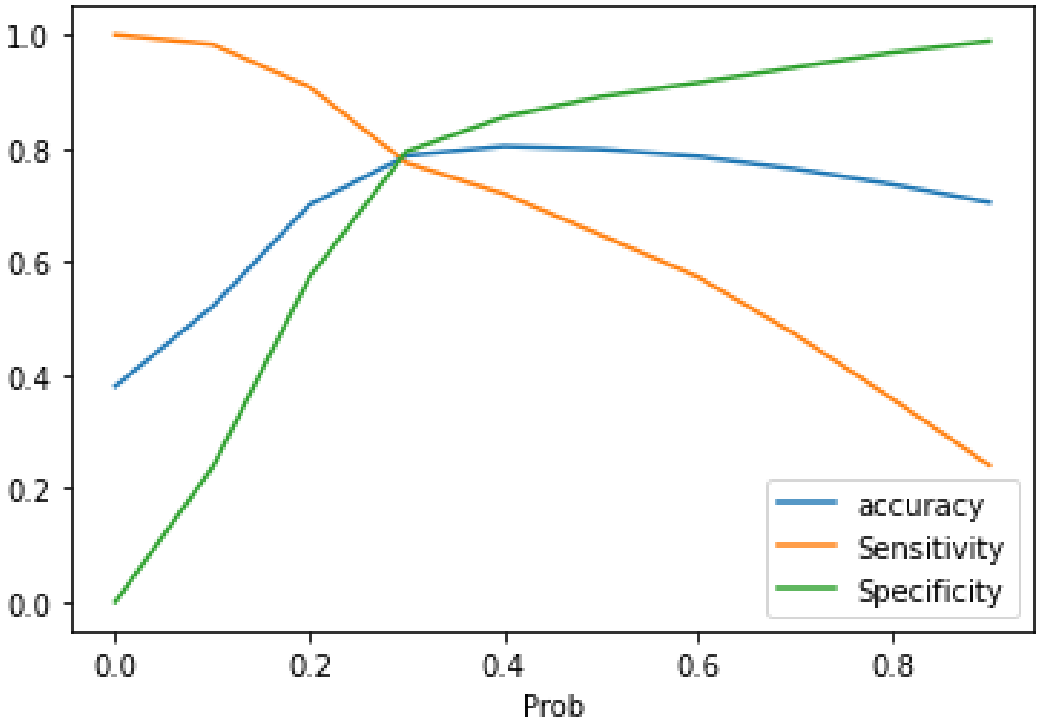
ROC curve:

- ROC curve area 0.86 is decent score.



Finding optimal cutoff:

- Calculating the accuracy, sensitivity and specificity for the cutoff values from 0 to 1 with the interval of 0.1.
- Plot all accuracy, sensitivity and specificity in line plot and find the cross over point.
- As per the plot we can see that the cross over is near 0.3 and taking optimal cutoff point as 0.3 for calculating the output variable.



Model evaluation with optimal cutoff:

- Predict the probability score using the training data (x_train) and calculate the output variable with the optimal cutoff value 0.3.
  - **Metrics scores:**

    Accuracy – 78.66%
  - **Confusion matrix:**

	Predicted No	Predicted Yes
Actual – No	3075	796
Actual - Yes	537	1838

  - **Sensitivity and Specificiy:**
    - Sensitivity = TP/ (TP+FN) = 77.39%
    - Specificity = TN/ (TN+FP) = 79.44%
  - **TPR and FPR:**
    - TPR = TP/ (TP+FN) = 77.39%
    - FPR = FP/ (FP+TN) = 20.56%

Prediction on test dataset :

- **Scaling:**
  - Scale the features 'TotalVisits','Total Time Spent on Website’ and 'Page Views Per Visit’ on the test dataset.
- **Add constant** to the test dataset and predict the probability values using the final model.
- Calculate the output variable (y) with the optimal cutoff value 0.3.

Model evaluation – Test dataset:

- Optimal cutoff value 0.3.
  - **Metrics scores:**

    Accuracy – 79.01%
  - **Confusion matrix:**

	Predicted No	Predicted Yes
Actual – No	1342	342
Actual - Yes	220	774

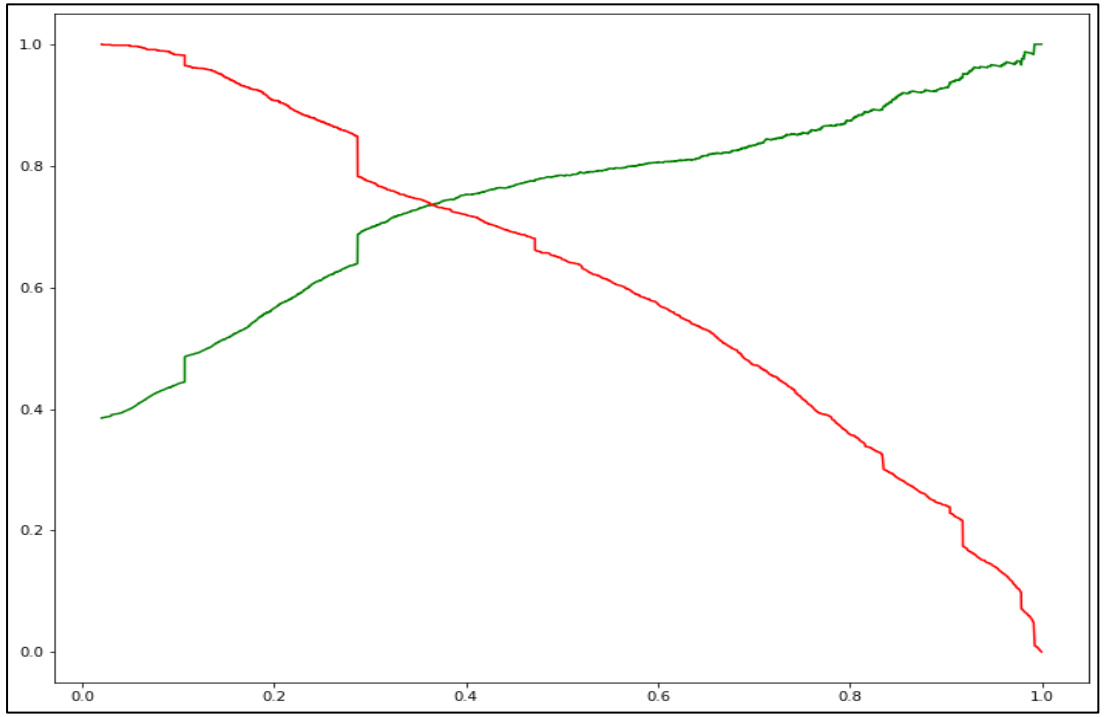
  - **Sensitivity and Specificiy:**
    - Sensitivity = TP/ (TP+FN) = 77.87%
    - Specificity = TN/ (TN+FP) = 79.69%
  - **TPR and FPR:**
    - TPR = TP/ (TP+FN) = 77.87%
    - FPR = FP/ (FP+TN) = 20.31%
- Model showing good scores on both training and test dataset.

Precision and Recall:

- **Test dataset:**
  - To find optimal cutoff using precision and recall using the below function.

```
precision_recall_curve(y_train_pred_final['Converted'],y_train_pred_final['Predicted_prob'])
```

- **Precision and recall tradeoff:**
  - Plotting line plot for precision and recall for each probability and we could see that the cutoff value is around 0.36.



Model evaluation with Cut off 0.36:

- **Training dataset:**
  - Accuracy = 79.94%
  - precision = TP / (TP + FP) = 73.43%
  - recall = TP / (TP + FN) = 74.02%
- **Test dataset:**
  - Accuracy = 80.55%
  - precision = TP / (TP + FP) = 73.25%
  - recall = TP / (TP + FN) = 74.95%

Conclusion:

- We have considered the optimal cut off using both sensitivity - specificity and precision-recall tradeoff.
- We got accuracy 79% , sensitivity 77% and specificity 79% for both training and test dataset.
- The conversion percentage as per the model prediction is around 77%
- With the final model, we can find that the below features are important to predict the hot leads.
  - The total time spent on website and the total visits.
  - When the lead source is 'welingak website' and 'google'.
  - Lead origin is 'lead add form'
  - Leads who are working professionals.