LEAD SCORING CASE STUDY

SUBMITTED BY:
Adwait Mohgaonkar
Ankush Sharma
Mohanapriya Pichandi

PROBLEM STATEMENT

X Education, an online education company for industry professionals, attracts numerous visitors to its website daily, where they explore various courses.

While some leads do convert, the majority do not. The typical lead conversion rate at X Education is approximately 30%.

To improve efficiency, the company aims to identify the most promising leads, termed 'Hot Leads.' By successfully pinpointing these leads, the sales team can concentrate their efforts on engaging with high-potential prospects, which is expected to increase the lead conversion rate compared to calling every lead.

The company needs us to develop a model that assigns a lead score to each prospect. This scoring system should ensure that leads with higher scores have a greater likelihood of conversion, while those with lower scores have a reduced chance. The CEO has indicated that the target conversion rate should be approximately 80%.

SOLUTION APPROACH

Step 1: Importing and analyzing data.

- Import the dataset and examine the data types of each column.
- Determine the percentage of missing values for each column.
- Assess the distribution of data within the numerical columns.
- Count the number of unique categories in each column..
- The dataset contains 9,240 rows and 37 columns.

Step 2: Data Cleaning

- Replace the 'Select' category with null values.
- Remove highly skewed and unnecessary columns:
  - Columns such as Prospect ID and Lead Number are irrelevant for prediction as they only contain unique identifiers.
  - Eliminate columns with significant skew, where one category has a high frequency while other categories have negligible frequency, such as 'Do Not Call', 'Country', 'Search', 'Magazine', and 'Newspaper Article'.

- Remove columns with 30% or more null values, as these columns are deemed irrelevant for prediction. Imputing them could introduce skewness that may impact model performance.
- Impute the nulls in those columns where null percentage is around 1%.
- Clean categorical columns by removing duplicates and consolidating categories with very low frequencies (less than 1%) into a single group.
- Handle the outliers by capping them.

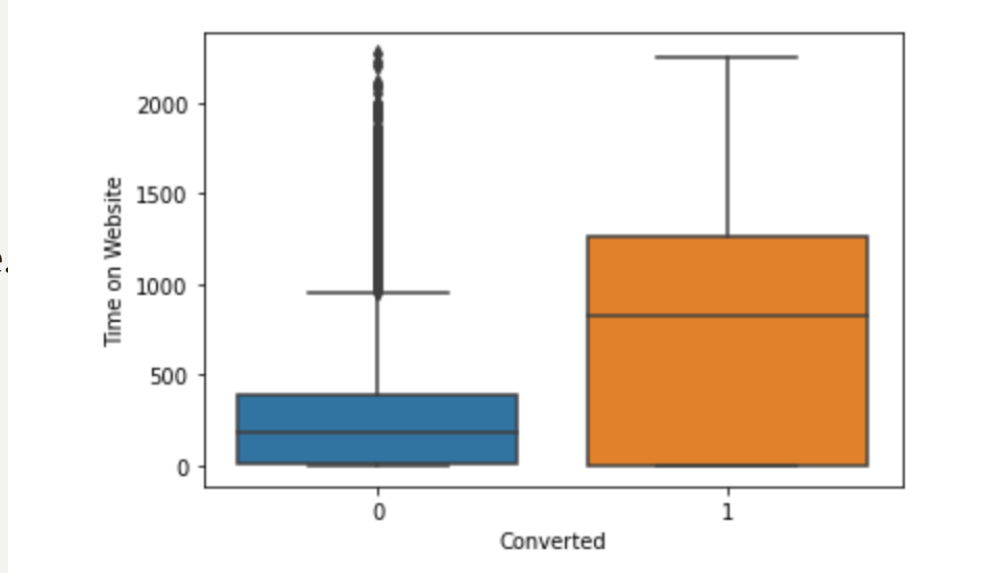
Step 3: EDA

- Conducted EDA on categorical variables

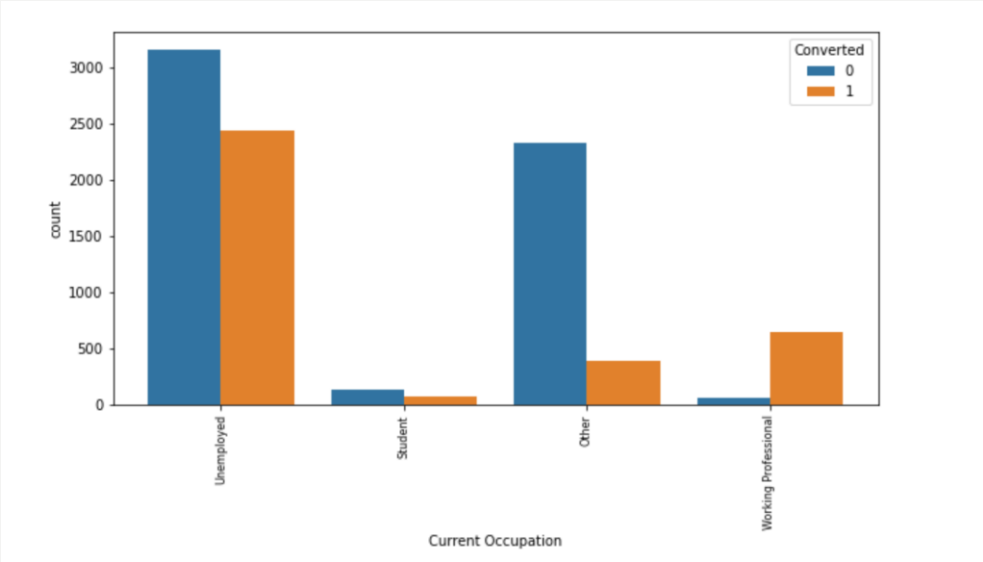
to determine their category-wise counts wrt Conversion type.

- Created boxplots for numerical attributes to

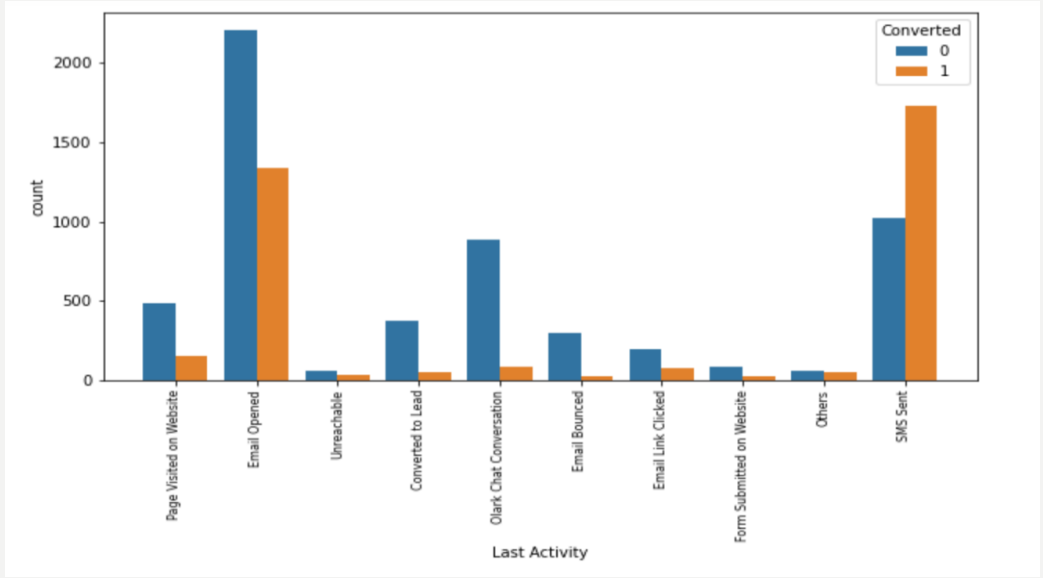
analyse their distribution wrt Converted type.



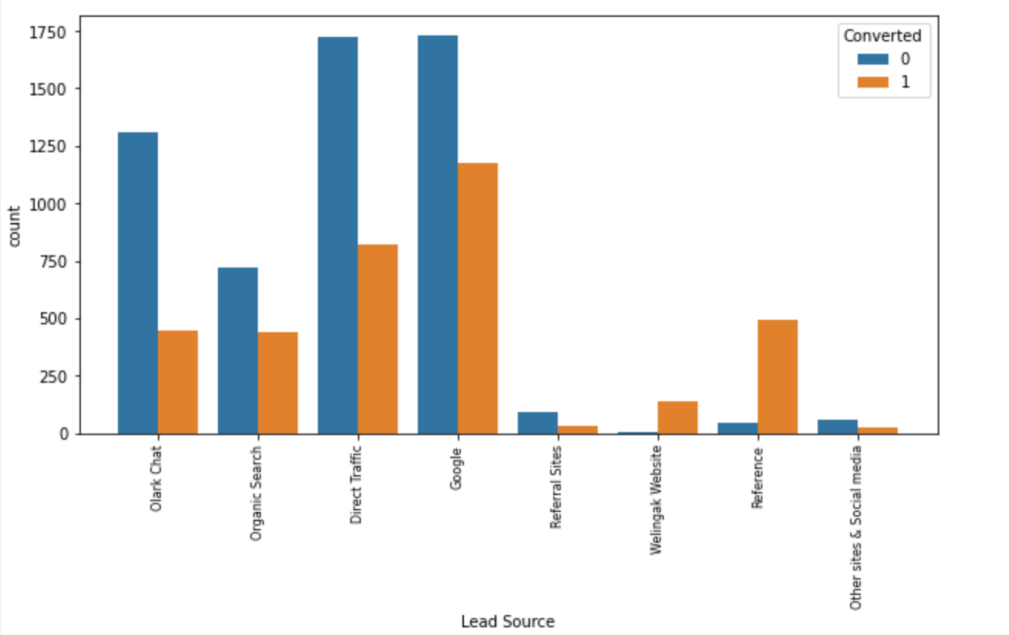
Leads who spent more time on website had higher conversion rates.



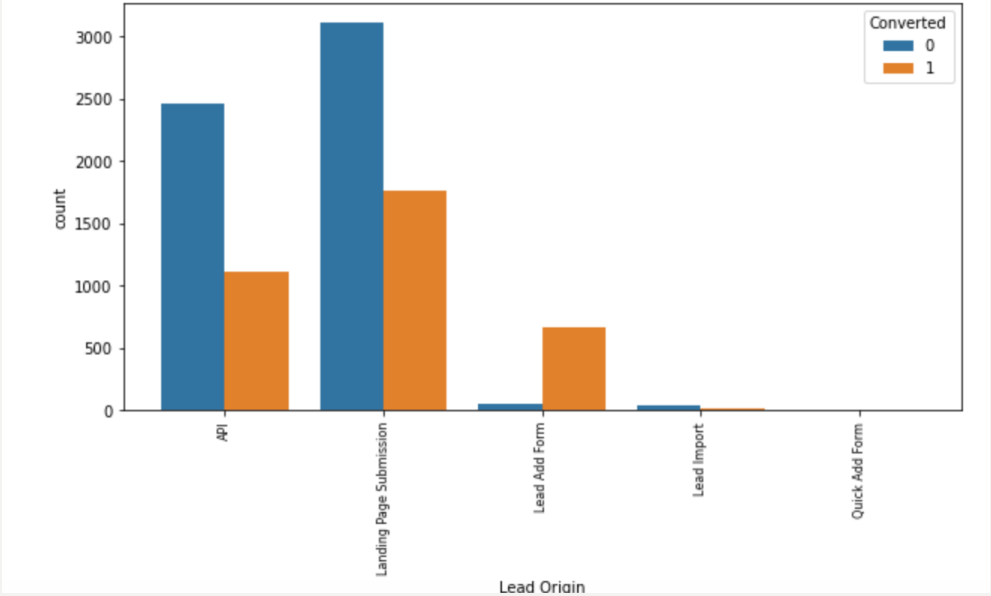
Working Professionals had highest conversion rate



Leads with SMS sent as Last activity had highest conversion rate



Leads through reference had highest conversion rate



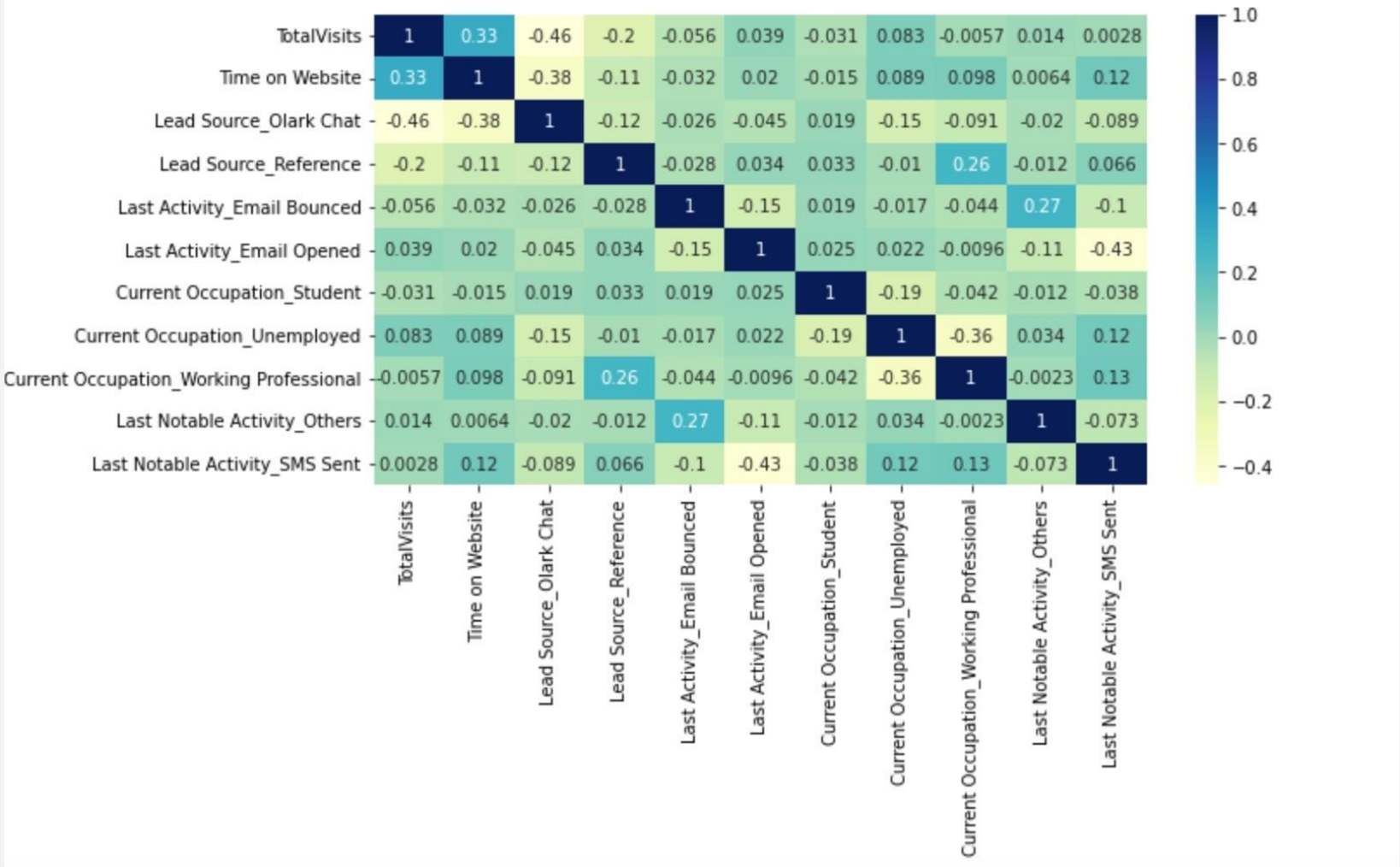
Leads with origin as Lead Add form had the highest conversion rate

Step 4: Data Preparation

- Generate dummy variables for categorical columns..
- Split the data into training and testing sets with a 70:30 ratio.
- Scale the numerical columns in the training set using MinMaxScaler. Scaling is necessary because we are using a Logistic Regression model; without scaling, higher-range numerical values would disproportionately influence the coefficients, reducing the model's interpretability.

Step 5: Model Building

- Initialize Logistic Regression model.
- Perform coarse feature selection using Recursive Feature Elimination and select top 15 features.
- Refine feature selection by eliminating features with high p-values and high Variance Inflation Factor (VIF).
- Ultimately, the final model included 11 features, all with p-values less than 0.05 and VIF values below 5.



These are our final set of features in Prediction model

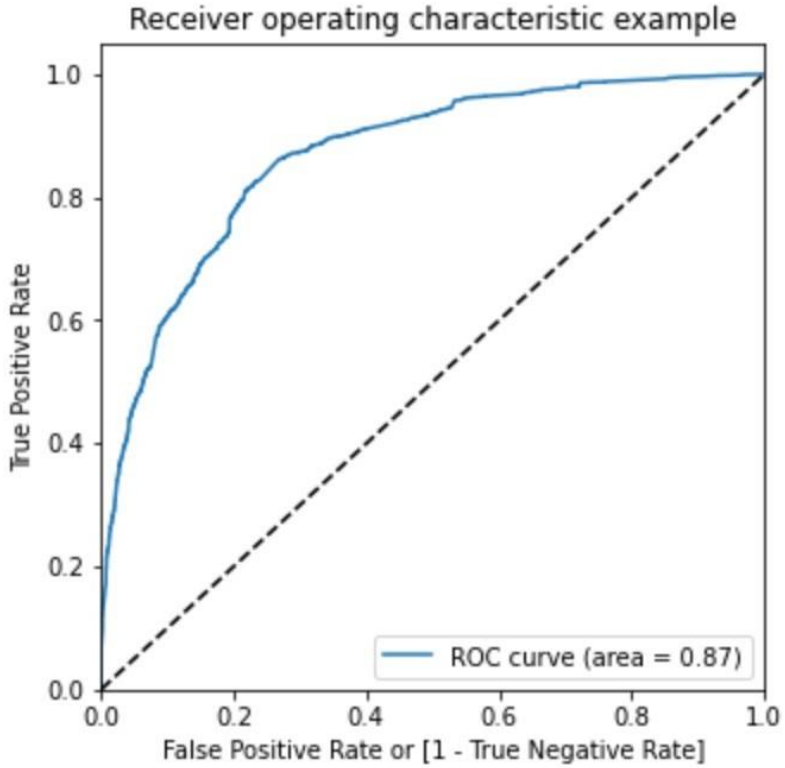
	coef	std err	z	P>|z|	[0.025	0.975]
const	-3.4948	0.117	-29.779	0.000	-3.725	-3.265
TotalVisits	0.2110	0.181	1.163	0.245	-0.145	0.567
Time on Website	4.0452	0.157	25.817	0.000	3.738	4.352
Lead Source_Olark Chat	0.8722	0.105	8.325	0.000	0.667	1.078
Lead Source_Reference	3.3581	0.215	15.653	0.000	2.938	3.779
Last Activity_Email Bounced	-1.7839	0.316	-5.638	0.000	-2.404	-1.164
Last Activity_Email Opened	0.5938	0.080	7.410	0.000	0.437	0.751
Current Occupation_Student	1.1486	0.224	5.135	0.000	0.710	1.587
Current Occupation_Unemployed	1.1460	0.084	13.719	0.000	0.982	1.310
Current Occupation_Working Professional	3.3497	0.188	17.797	0.000	2.981	3.719
Last Notable Activity_Others	1.6558	0.262	6.322	0.000	1.142	2.169
Last Notable Activity_SMS Sent	1.9421	0.090	21.512	0.000	1.765	2.119

The top features for prediction based on their coefficients are:

1. TotalVisits
2. Lead Source_Reference
3. Current Occupation_Working Professional

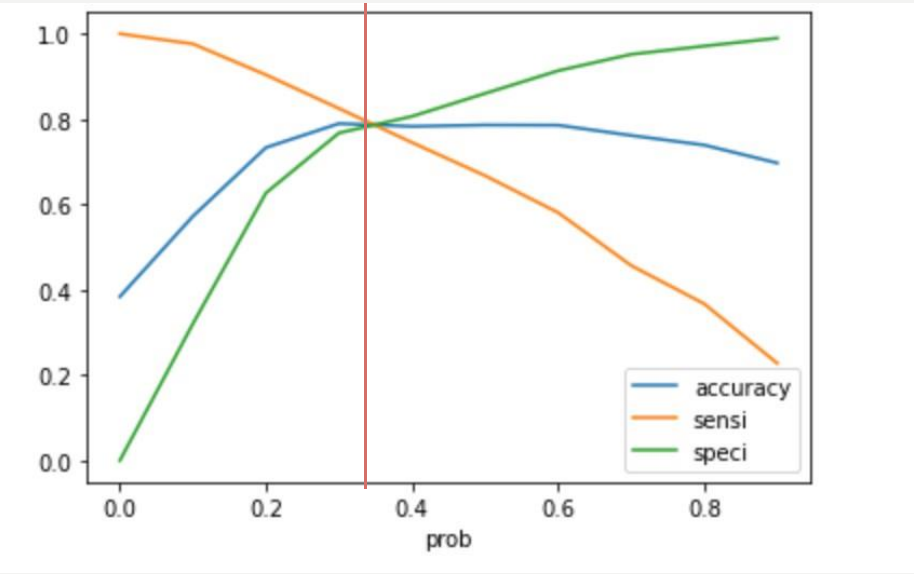
Step 6: Model Performance Evaluation

- Generate predictions with the final model using the training set.
- Plot the ROC curve for the model, which showed a strong performance with the curve closely aligned to the top left corner and an AUC of 0.87.
- Use the confusion matrix to calculate and plot Accuracy, Sensitivity, and Specificity at various probability thresholds
- Sensitivity (or Recall) measures the proportion of actual converted leads that were correctly predicted as converted.
- $\text{Recall} = \frac{\text{True Positive (TP)}}{\text{Total Actual Positives (TP + FN)}}$
- Precision is the ratio of leads correctly predicted as converted to the total number of leads predicted as converted
- $\text{Precision} = \frac{\text{True positives (TP)}}{\text{Total Predicted positives (TP+FP)}}$
- In our case, a slightly lower precision may result in contacting some non-potential leads. Therefore, we are willing to trade off a bit of precision to achieve higher recall.

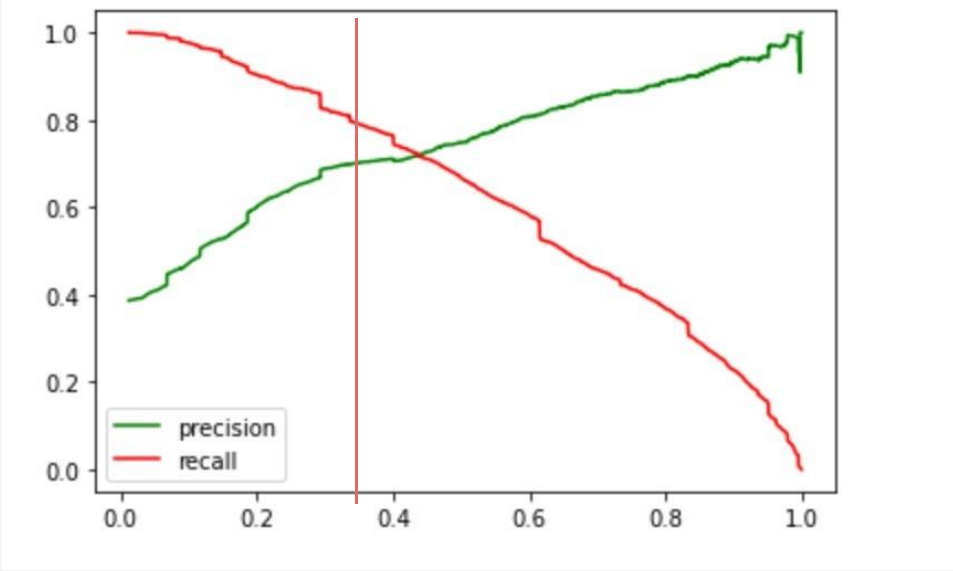


ROC curve of the model

- Our objective is to contact as many leads as possible to maximize the conversion rate. To achieve this, we need to set a probability cutoff that boosts recall, allowing us to identify most of the potential leads (Hot Leads), while maintaining a reasonable level of precision to minimize unnecessary calls to non-potential leads.
- Thus, we set the cutoff probability at 0.35 to balance higher recall with adequate precision. As a result, our model achieved a target lead conversion rate of approximately 79%, meaning we were able to reach out to 79% of the leads who ultimately converted, thereby supporting a high conversion rate.



Accuracy – Sensitivity – Specificity tradeoff



Precision – Recall tradeoff

The Red line indicates the probability cutoff above which all leads who be classified as Converted.

Step 7: Evaluation of model on the test set

- Apply the MinMaxScaler to transform the numerical columns in the test set.
- Select the columns from the test set that were used in the final prediction model from the training set.
- Generate predictions on the test set and assess the performance metrics.

	Accuracy	Precision	Recall	f1-score
Train	0.790816	0.70247	0.789855	0.743604
Test	0.797619	0.711823	0.805014	0.755556

Step 8: Forming the Scores dataframe

- Merge the initial dataframe with the predictions dataframe using the 'ID' column, which is the index from the initial dataframe.
- Compute the Score as Probability of Conversion×100Probability of Conversion×100.
- The final scores dataframe will include the Prospect ID, Lead Number, Actual Converted status, Predicted Converted status, and the Score.

	Prospect ID	Lead Number	Converted	final_predicted	Score
0	7927b2df-8bba-4d29-b9a2-b6e0beafe620	660737	0	0	18.59
1	2a272436-5132-4136-86fa-dcc88c88f482	660728	0	1	38.12
2	8cc8c611-a219-4f35-ad23-fdfd2656bd8a	660727	1	1	73.17