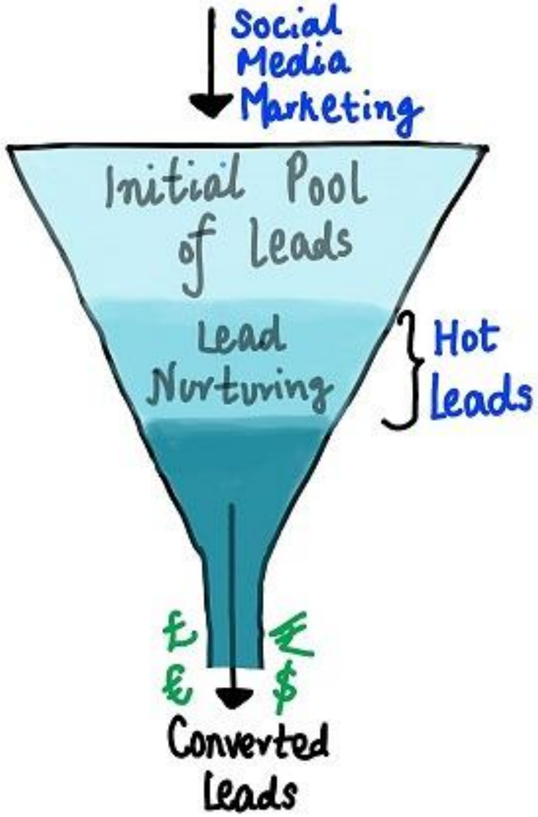
Case Study on Lead Scoring

KRUNAL TANNA

ALAKNANDA AGARWAL

Introduction

- An online education company X Education sells online courses to industry professionals. Although, X Education gets a lot of leads, its lead conversion rate is very poor. The typical lead conversion rate at X education is around 30%.
- The company wishes to identify the most potential leads, also known as ‘Hot Leads’ to enable the sales team to focus more on communicating more with the potential leads. A typical lead conversion process can be represented using the adjoining funnel.
- We have been given a ballpark of the target lead conversion rate to be around 80%.
- The goal 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.



Data Cleaning and Outlier Handling

- Replaced all ‘Select’ values with Nulls.
- Converted all columns with Yes-No to 1-0.
- Removed columns with no value addition to the dataset, and those which were extremely skewed(having equal to or more than 90% of one category). Columns found irrelevant have been dropped directly.
- Imputed nulls in other columns with appropriate values.
- Combined categories occurring very less(less than 4% of the total values) into a single category.
- For columns having very less nulls, dropped those null records
- There were some outliers in the continuous columns. Performed upper limit capping at 99% for them.

Exploratory Data Analysis

From Fig. 1, we can see that Avg Time Spent per visit is not affecting the conversion so much, because the non-converted group has many outliers. Suggests that maybe the company has something on their website turning people away or is missing something that they are specifically looking for.

In Fig. 2, we can see that even though Total Visits to website are more for non-converted, the total time spent on website is more for those converted.

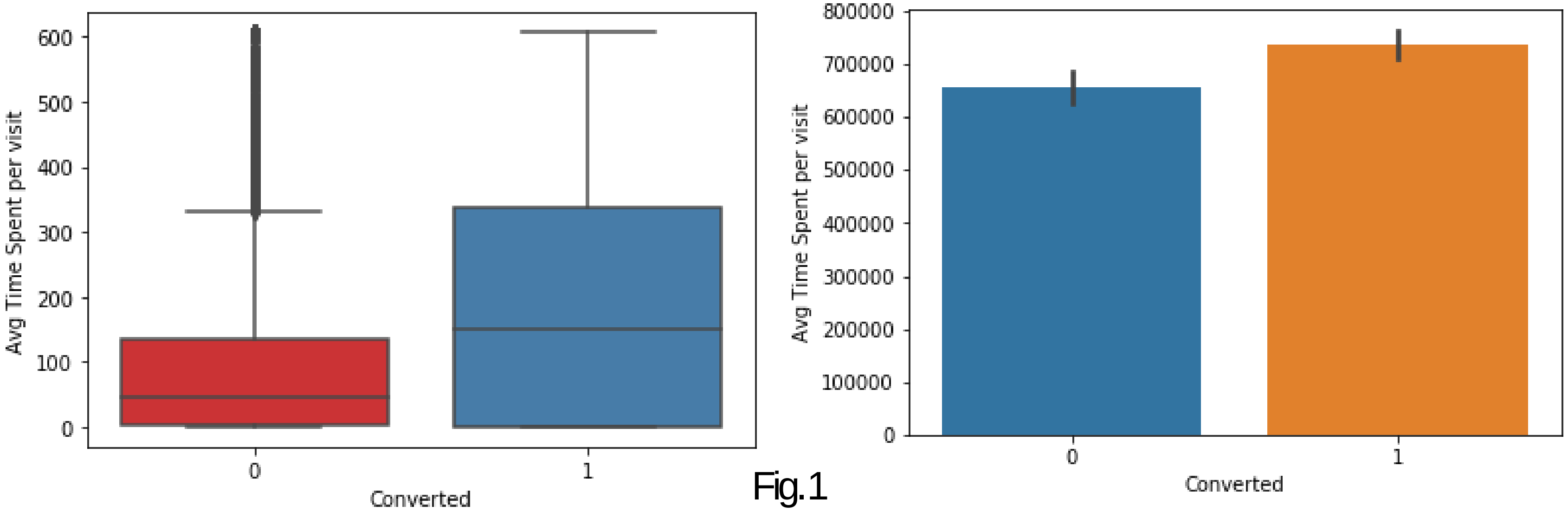


Fig.1

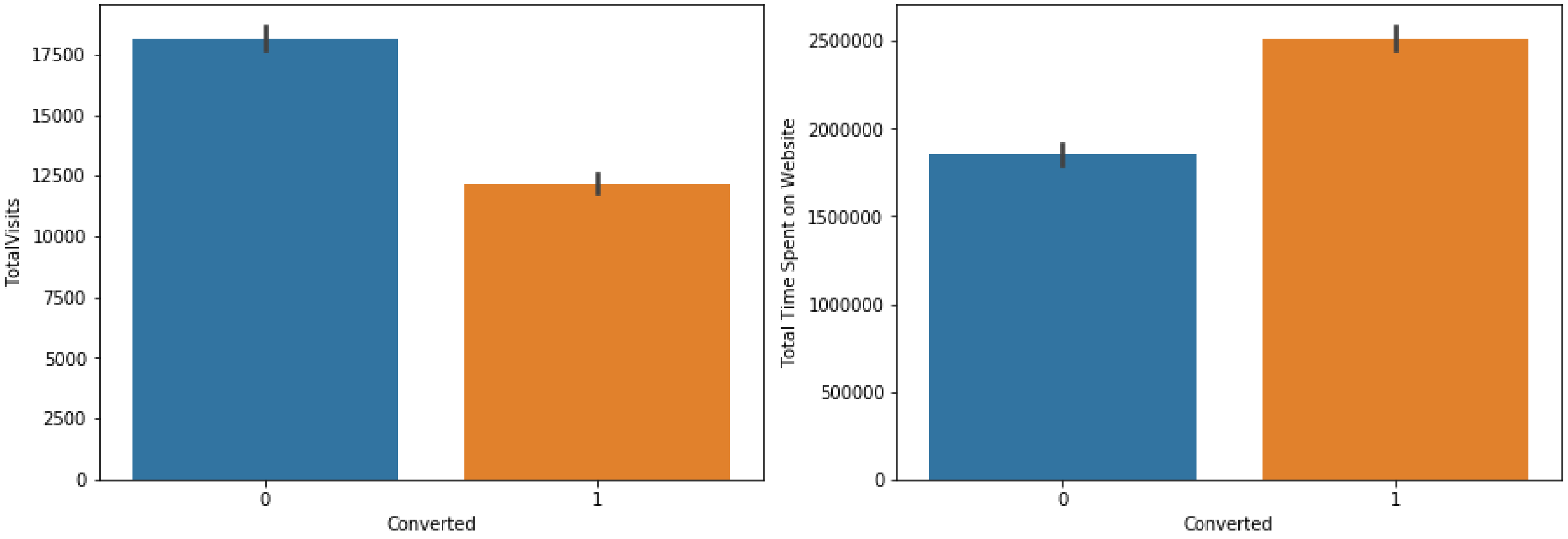


Fig.2

Fig. 3 shows that even though most leads are coming through Landing Page Submission, the conversion rate is maximum for Lead Add Form. The company needs to get more leads from Landing Page Submission.

Fig.3

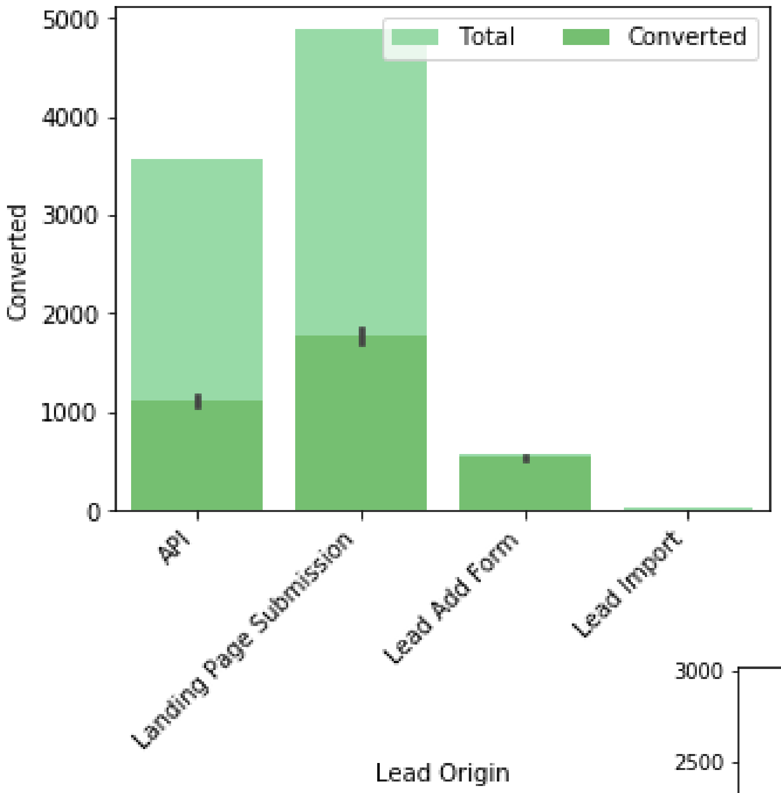


Fig. 4 shows that most leads are being sourced from Google and Direct Traffic, but conversion rate of leads through References are very high. Thus, company needs to try to increase leads through References, and increase lead conversion rate from Google.

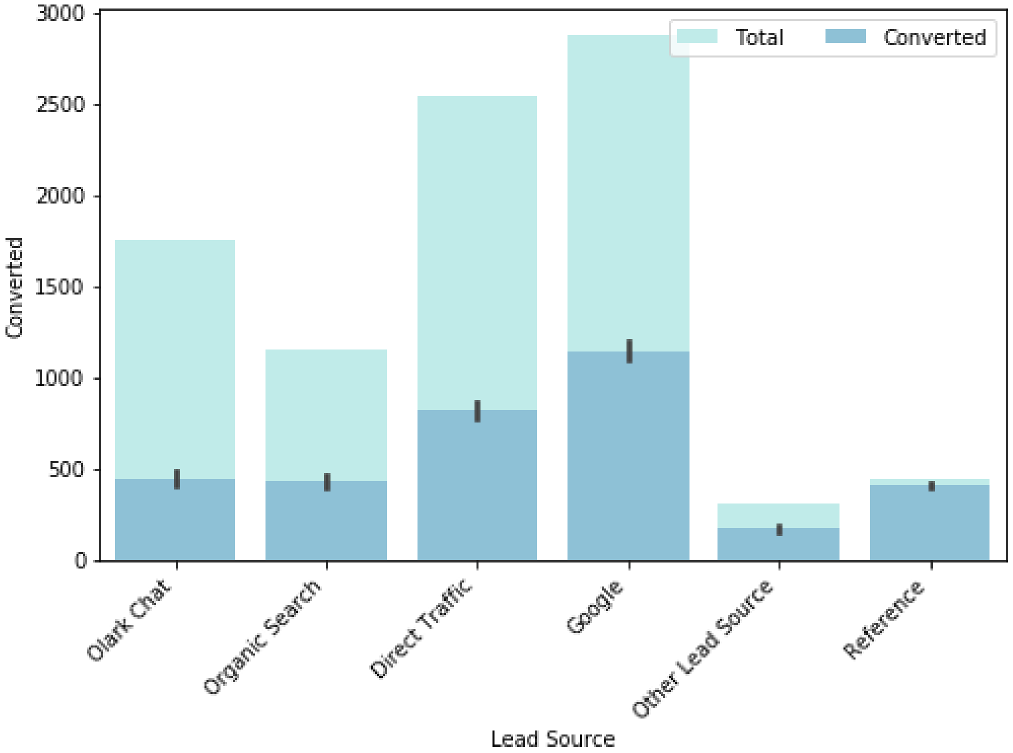
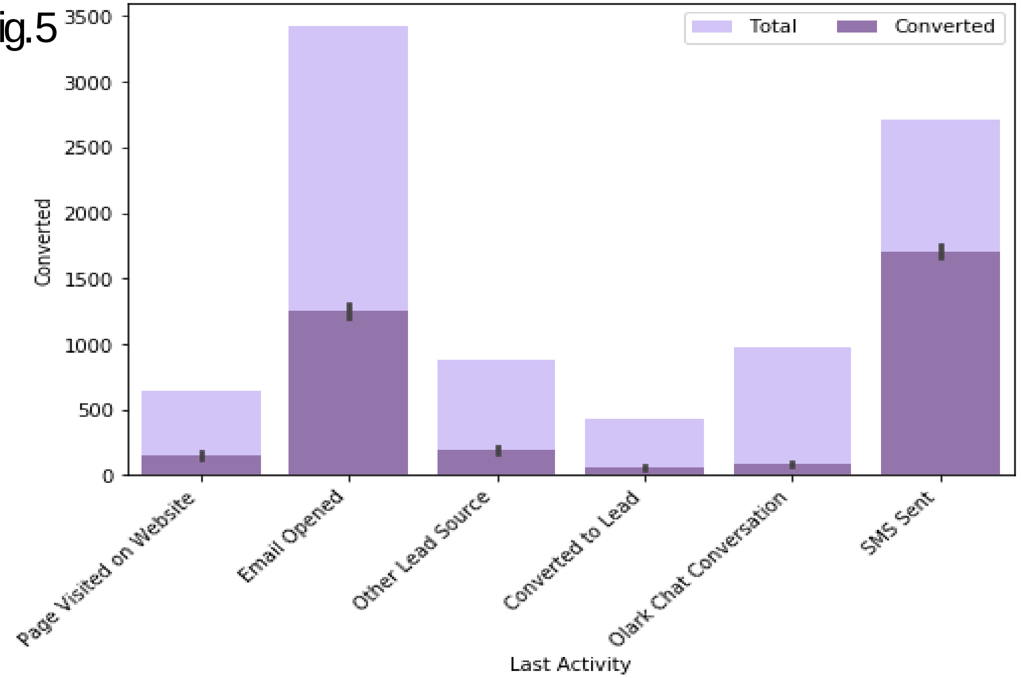


Fig.4

Fig. 5 we can see that even though the Last Activity of most people is Email Opened, the maximum conversion rate is when their last activity is SMS Sent, which suggests the sales team needs to step up its social media marketing. The last activity seeing most conversions is ‘SMS Sent’, and SMS are a very old technology. If their highest conversions are from such an old technology, it’s not good.

Fig. 6 shows that the highest number of people looking for these courses are Unemployed, but conversion rate is highest for Housewives and Working Professionals. Thus, the company needs to generate more leads with these two occupations and offer placement services for increasing conversion rate among Unemployed.

Fig.5



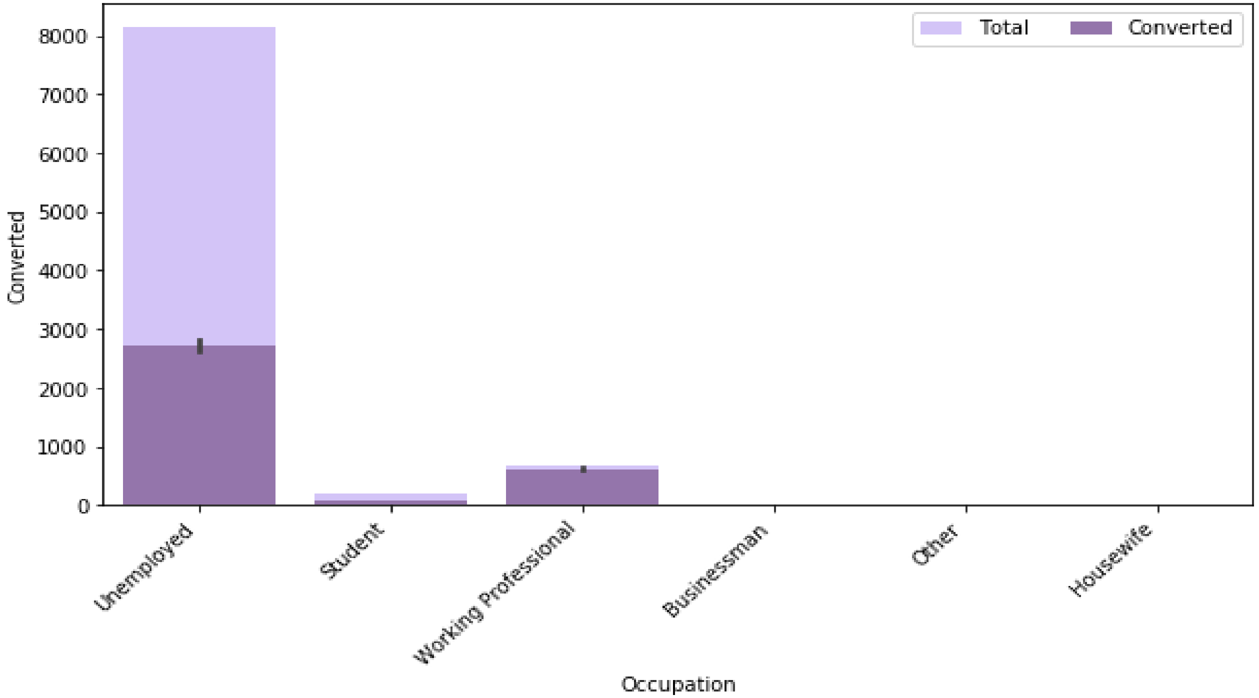


Fig.6

Fig. 7- Maximum leads are from Mumbai city, and even though conversion rate for all cities is approximately the same, it might increase for other cities if the company generates more leads from the other cities.

Fig. 8- Maximum of the converted leads are not opting for the freebie, suggesting that the freebie is not very instrumental in influencing their decision. A better freebie needs to be offered.

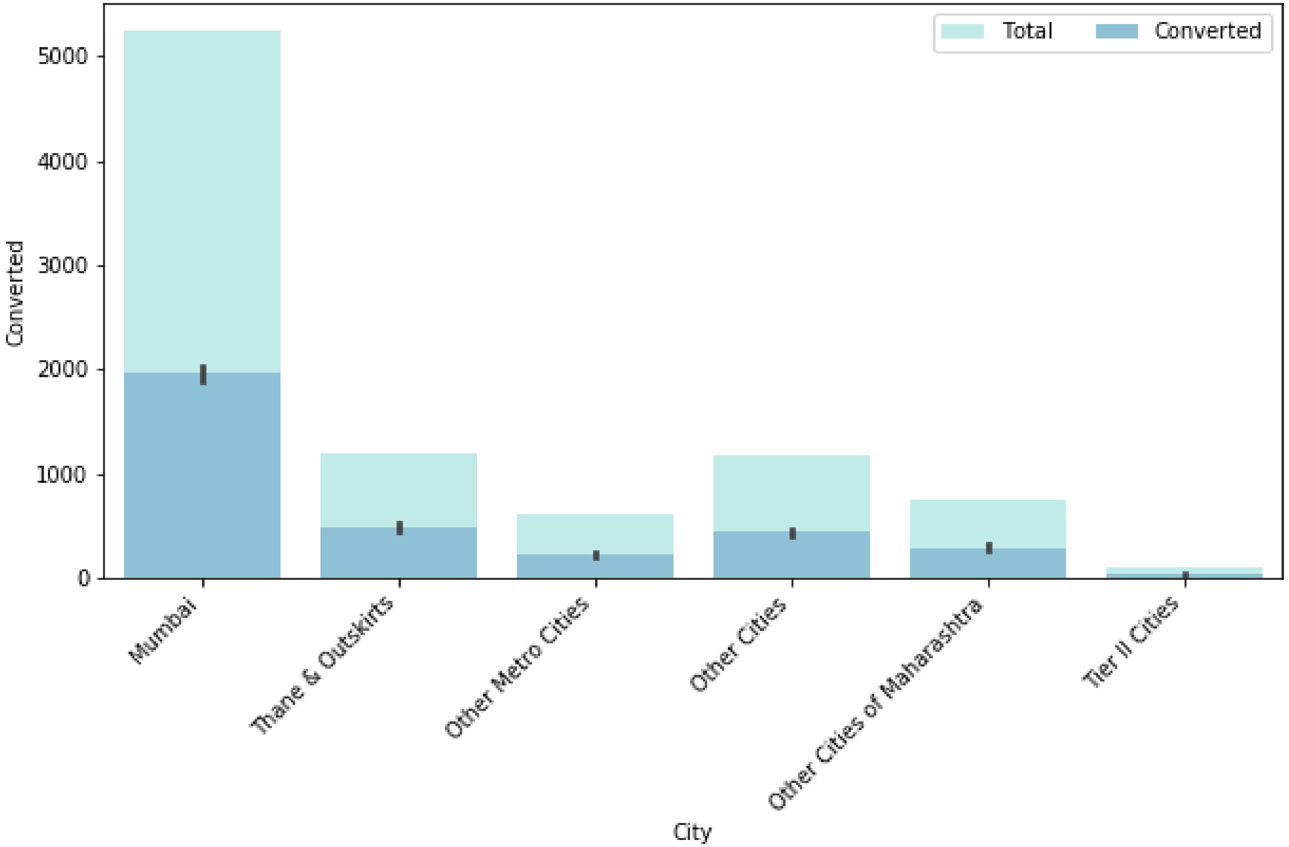


Fig.7

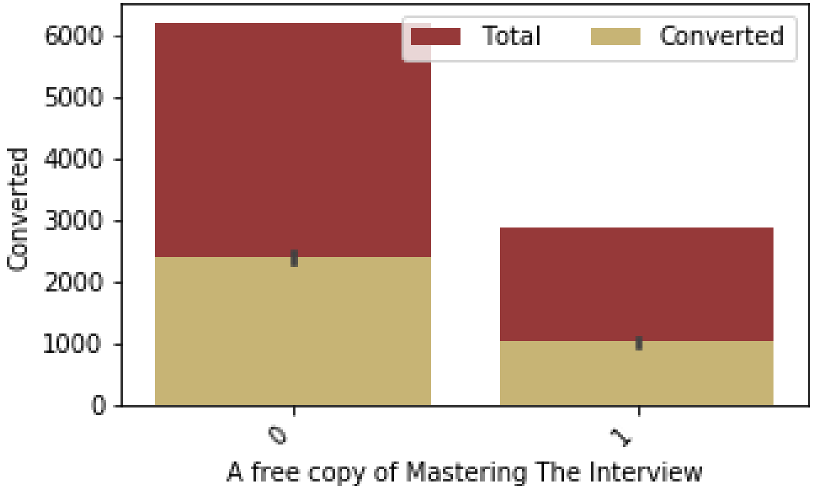


Fig.8

Dummy Variables, Scaling, Train-Test Split

- Created dummy variables for the 6 categorical variables in the dataset- 'Lead Origin', 'Lead Source', 'Last Activity’, 'Specialization', 'Occupation', 'City’.
- After creation of dummy variables, the dataset shape is (9074,41).
- Next, we split our total data into two parts- train and test using the ‘sklearn train_test_split’ library in the ratio of 7:3.
- Train set will be used to train the logistic regression model for learning all the data peculiarities. This model will then be fitted on the test set.
- After the split, we rescaled our train data. There are two common ways of rescaling:
  a) Min-Max scaling
  b) Standardization (mean-0, sigma-1)
- Here, have used Standardization Scaling.
- Before building the model, we have a conversion rate of 38%.

Model Building- Logistic Regression

- After all of this was done, a logistic regression model was built in Python using the function GLM() under statsmodel library.

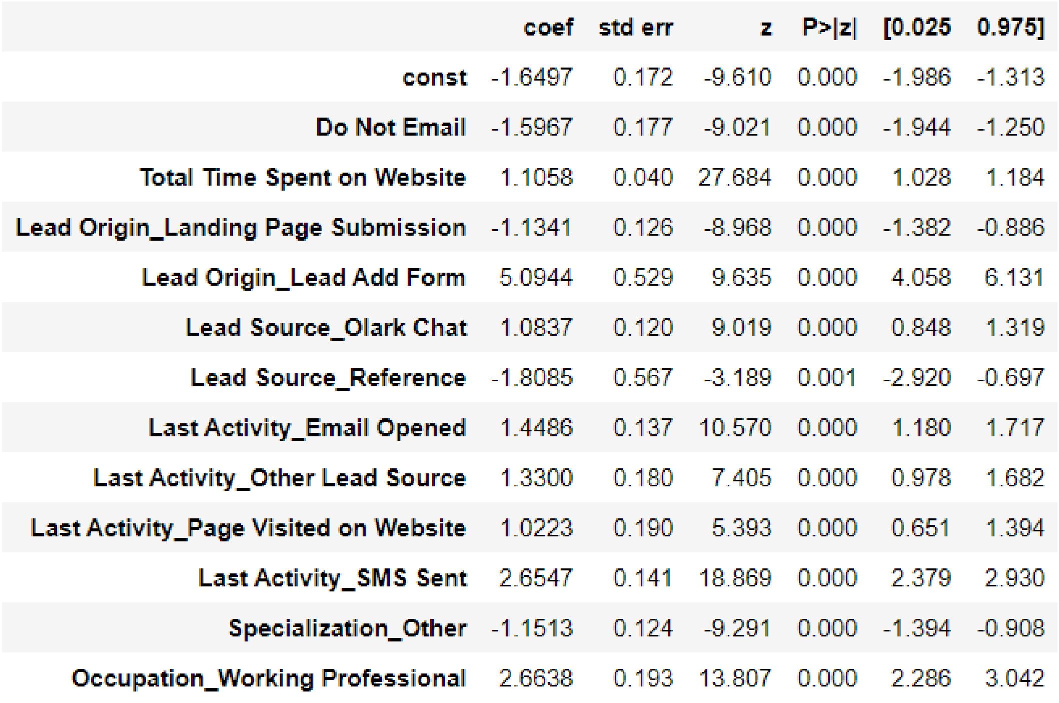
- This model contained all the variables, some of which had insignificant coefficients. Hence, some of these variables were removed first based on an automated approach, i.e. RFE.

- Variables having a high p-value were removed one-by-one, because they were insignificant. After this, the VIF was checked of all the remaining variables.

- Variables with VIF more than 5 were removed to avoid the problem of multi-collinearity in the model.

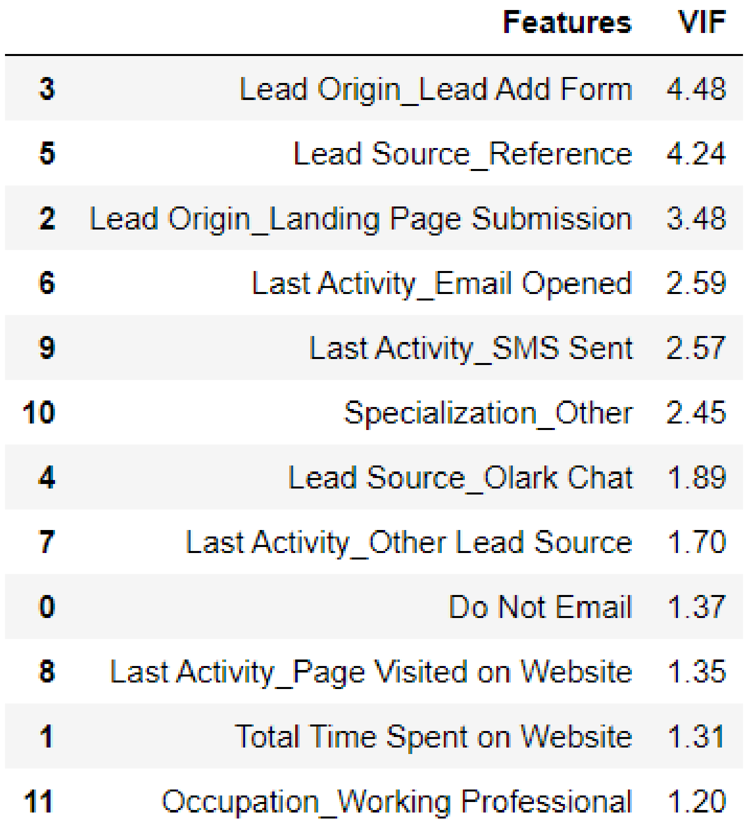
Assessing the Model

- After performing logistic regression and dropping variables one at a time, we have reached a level where all the P values are almost 0 and VIF is under control.
- It took 9 iterative models to reach this step.
- The final columns in 9th model are:
    - Lead_Source_Reference
    - Lead Origin_Landing Page Submission
    - Last Activity_Email Opened
    - Last Activity_SMS Sent
    - Specialization_Other
    - Lead Source_Olark Chat
    - Last Activity_Other Lead Source
    - Do not Email
    - Last Activity_Page Visited on Website
    - Total Time Spent on Website
    - Occupation_Working Professional



	coef	std err	z	P>|z|	[0.025	0.975]
const	-1.6497	0.172	-9.610	0.000	-1.986	-1.313
Do Not Email	-1.5967	0.177	-9.021	0.000	-1.944	-1.250
Total Time Spent on Website	1.1058	0.040	27.684	0.000	1.028	1.184
Lead Origin_Landing Page Submission	-1.1341	0.126	-8.968	0.000	-1.382	-0.886
Lead Origin_Lead Add Form	5.0944	0.529	9.635	0.000	4.058	6.131
Lead Source_Olark Chat	1.0837	0.120	9.019	0.000	0.848	1.319
Lead Source_Reference	-1.8085	0.567	-3.189	0.001	-2.920	-0.697
Last Activity_Email Opened	1.4486	0.137	10.570	0.000	1.180	1.717
Last Activity_Other Lead Source	1.3300	0.180	7.405	0.000	0.978	1.682
Last Activity_Page Visited on Website	1.0223	0.190	5.393	0.000	0.651	1.394
Last Activity_SMS Sent	2.6547	0.141	18.869	0.000	2.379	2.930
Specialization_Other	-1.1513	0.124	-9.291	0.000	-1.394	-0.908
Occupation_Working Professional	2.6638	0.193	13.807	0.000	2.286	3.042

Stats of Final Variables



	Features	VIF
3	Lead Origin_Lead Add Form	4.48
5	Lead Source_Reference	4.24
2	Lead Origin_Landing Page Submission	3.48
6	Last Activity_Email Opened	2.59
9	Last Activity_SMS Sent	2.57
10	Specialization_Other	2.45
4	Lead Source_Olark Chat	1.89
7	Last Activity_Other Lead Source	1.70
0	Do Not Email	1.37
8	Last Activity_Page Visited on Website	1.35
1	Total Time Spent on Website	1.31
11	Occupation_Working Professional	1.20

VIF Score of Final Variables

Making prediction on Train Set

- Now we have applied this model on our Train set using all the probability Scores.
- To do that, we must choose an arbitrary cut-off point to find the predicted labels.
- We start by choosing 0.5 as ideal value
- Creating a new column ‘predicted’ with 1 if Converted Probability >0.5 else 0

Confusion Matrix

- A **confusion matrix** is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known.

- It is basically a performance measure to get the overall status of the model

- We have used confusion_matrix() function from metrics library of sklearn

- TP denotes: True Positive values
- FP denotes: False Positive values
- FN denotes: False Negative values
- TN denotes: True Negative values

- Using this values we have calculated the overall accuracy, sensitivity and Specificity of our model

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Performance Measures

- After calling all measures the results are:

    - Accuracy of our model: **80.6%**
    - Sensitivity: **66.8%**
    - Specificity: **89.4%**
    - False Positive rate: **10%**
    - Positive Predictive value: **79.8%**
    - Negative Predictive value: **81.10%**

- We found out that our specificity was good around 89% but our sensitivity was only 66%. Hence, this needed to be taken care of as Sensitivity is the ratio of number of conversions correctly predicted to the number of actual conversions. So, having a high Sensitivity value will lead to an increase in efficiency of our model.

Plotting ROC Curve

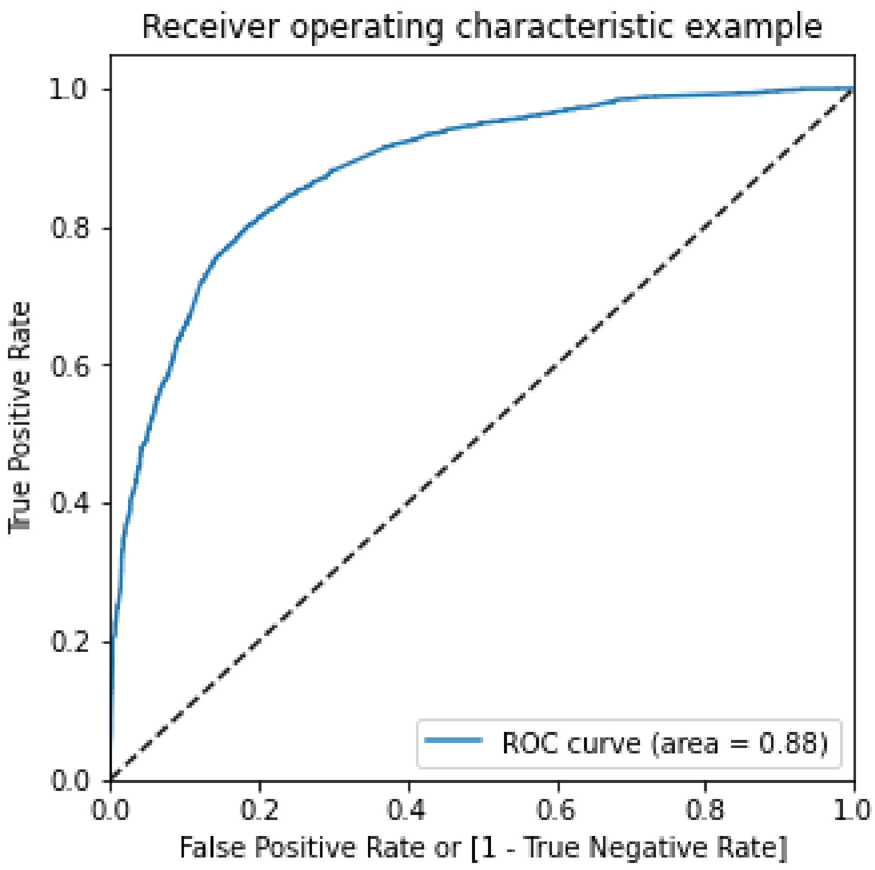
- We have got sensitivity of only **66%** and this was mainly because of the cut-off point of **0.5** that we had arbitrarily chosen.

- So, this cut-off point had to be optimized in order to get a decent value of sensitivity and for this we have used the ROC curve

- An ROC curve shows the tradeoff between sensitivity and specificity (any increase in sensitivity will be accompanied by a decrease in specificity). The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the test. The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the test.

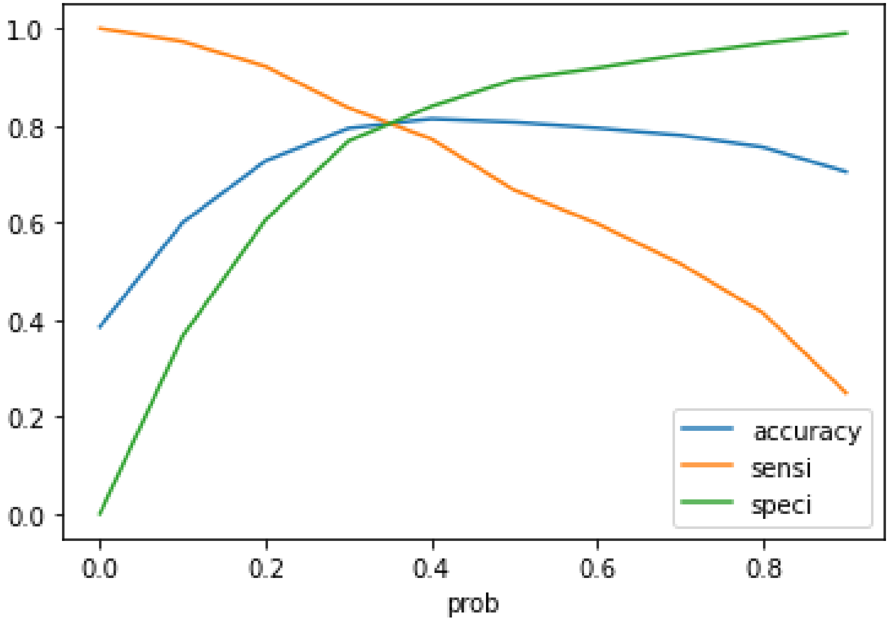
ROC Curve

- As we can see from the figure, we have higher (0.88) area under the ROC curve
- Therefore our model is a good one



Finding Optimal Cut-off

- Optimal cutoff probability is that prob where we get balanced sensitivity and specificity
- So we plotted the accuracy, sensitivity and specificity using a line chart, the point of intersection is the point with optimal cut-off value.
- From the figure, It is observed that 0.35 is the point of intersection and so we chose that value.
- We predicted the new values using **0.35** as cut-off and check the performance measures.



Performance Measures with Optimal Cut-off

- After calling all measures again the results are:

    - Accuracy of our model: **80.6%**
    - Sensitivity: **80.5%**
    - Specificity: **80.7%**
    - False Positive rate: **19%**
    - Positive Predictive value: **72.4%**
    - Negative Predictive value: **86.8%**

- We also checked the Precision and Recall
- Precision refers to the percentage of the results which are relevant: **72%**
- Recall refers to the percentage of total relevant results correctly classified by the algorithm: **80.5%**

- **Thus, Now we have a very good and balanced Accuracy, Sensitivity and Specificity score.**
  So, We can finalize this model.

Predictions on Test Data

- The finalized model is now used to make prediction on Test data to get an overall confirmation of our model's efficiency, After running the model on the Test Data , we obtain:

  - Accuracy : **80.8 %**
  - Sensitivity : **80.4 %**
  - Specificity : **81.0 %**

Conclusion

- Thus we have achieved our objective of reaching the ballpark target of lead conversion rate to be around 80% assigned by the CEO of X Education

- Our Model seems to predict the conversion rate very well and so Company's sales team can make useful calls based on the model to get a higher lead conversion rate of 80%

- Moreover, The top 3 variables which can be considered by the sales team are:

  - **Lead Origin_Lead Add Form-** More leads need to be scouted through this lead origin, since it has the highest conversion rate among all lead origins.
  - **Occupation_Working Professional-** X Education needs to reach out to more Working Professionals because they have a very high conversion rate.
  - **Last Activity_SMS Sent-** Market the courses on social media more aggressively.