Lead Scoring Case Study

Gaurav Garg

Subrahmanyam Vedula

Problem Statement

- An education company named X Education sells online courses to industry professionals. The company markets its courses on several websites and search engines like Google. Once these people land on the website, they fill up forms providing details.
- 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 company requires you to build a model wherein you need 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%.

Data Understanding and Analysis

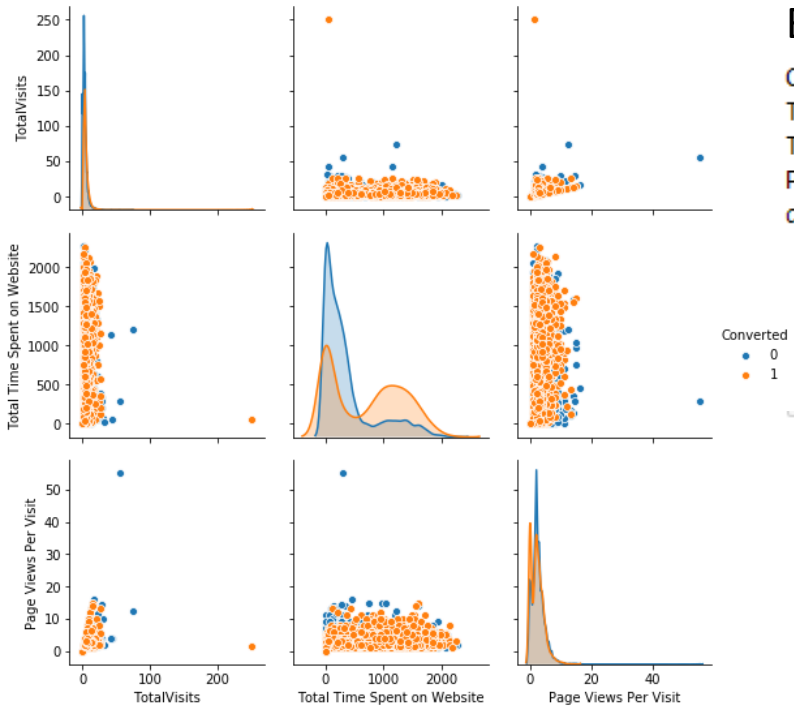
- The Data has details of 9240 leads that are most likely to convert into paying customers.
- Data includes columns like Lead Source, Total Time Spent on Website, Total Visits, Last Activity, etc.
- The target variable, in this case, is the column ‘Converted’ which tells whether a past lead was converted or not wherein 1 means it was converted and 0 means it wasn’t converted.
- Some columns in the data have more than 30 percent of null values, which we will need to clean.

Cleaning the data

- Dropped columns with null values greater than 45% since they are of no use to our analysis.
- Dropped Prospect ID, Lead Number, City & Country since they cannot be used in regression.
- Three categorical columns have ‘Select’ as one of the categories because if the field is left blank in form it’s value is Select. Out of those three we dropped Lead Profile, How did you hear about X Education as number of ‘Select’ in these two are very high.
- From printing value_counts of all columns we saw that few columns have 'No' in almost all rows causing data imbalance, hence those columns were also dropped.
- Dropped Rows which were null in ‘What matters most to you in choosing a course’ and ‘Lead Source’.
- Created a separate category for null values in Tags columns called ‘Unknown’ since number of nulls were large in number so we cannot drop the rows.

Exploratory Data Analysis

- From pair plot between numerical variables we saw that numerical values are right skewed.
- After power transformation, the skewness is gone.

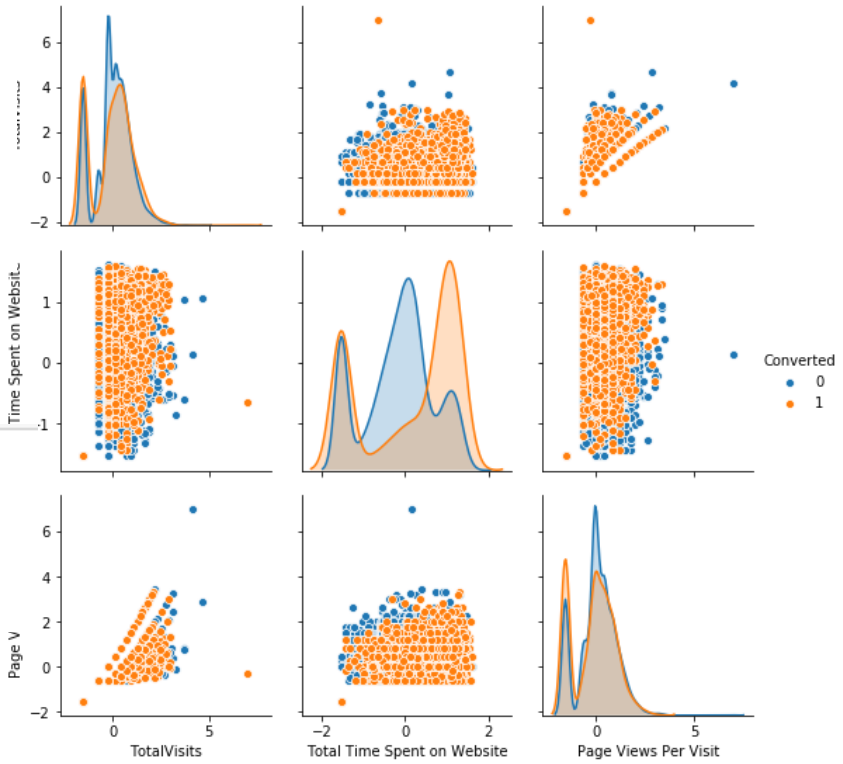


Before Power Transformation

```
Converted                       0.076030
TotalVisits                    22.460170
Total Time Spent on Website     0.796431
Page Views Per Visit            3.323551
dtype: float64
```

After Power Transformation

```
TotalVisits                    -0.004138
Total Time Spent on Website    -0.338492
Page Views Per Visit           -0.021681
Converted                       0.076030
dtype: float64
```



Creating dummy variables

- For creating dummy variables we used pd.get_dummies() function for all categorial columns.
- For Specialization and Tags, we customized dummy variable creations since these columns had ‘Select’ and ‘Unknown’ categories which required to be removed.
- Finally we have 97 dummy variables and 4 numerical variables in our data to be processed.

Preprocessing Data for the model

- We spilt the data into train and test (70/30) and scaled numerical variables using MinMaxScaler from sklearn.preprocessing.
- We used RFE (Recursive Feature Elimination) from sklearn.feature_selection to choose top 15 variables for our model building with Logistic regression.
- Below are the final 15 selected variables after RFE.

```
Index(['Lead Source_Welingak Website', 'Tags_Already a student',
       'Tags_Closed by Horizzon', 'Tags_Diploma holder (Not Eligible)',
       'Tags_Graduation in progress', 'Tags_Interested  in full time MBA',
       'Tags_Interested in other courses', 'Tags_Lost to EINS',
       'Tags_Not doing further education', 'Tags_Ringing',
       'Tags_Will revert after reading the email', 'Tags_invalid number',
       'Tags_number not provided', 'Tags_switched off',
       'Tags_wrong number given'],
      dtype='object')
```

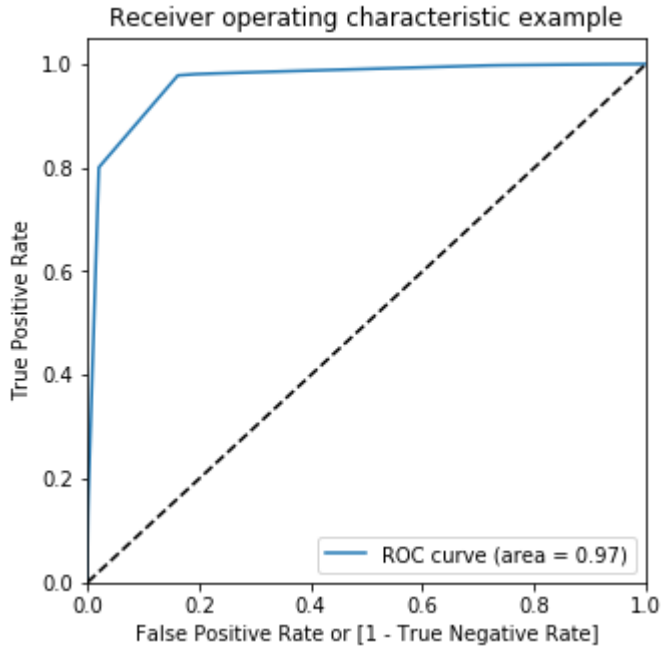
Model Building

- Using statsmodels.api GLM method for logistic regression algorithm.
- From first run we found out that three variables have a very high p-value of 0.999 and VIFs are all lower than 2.
- After removing these three variables one by one and also checking VIF values in between, we finalized with these 12 variables.

Features
Lead Source_Welingak Website
Tags_Closed by Horizzon
Tags_Graduation in progress
Tags_Interested in full time MBA
Tags_Lost to EINS
Tags_Not doing further education
Tags_invalid number
Tags_switched off
Tags_Interested in other courses
Tags_Already a student
Tags_Will revert after reading the email
Tags_Ringing

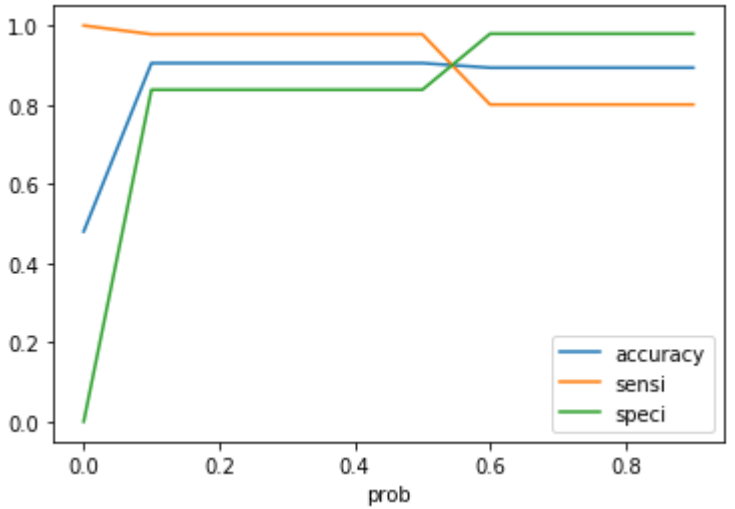
Model Evaluation

- Evaluated our final model on the train dataset with a randomly chosen cutoff of 0.5. We got 90.5 accuracy score with 97.8% sensitivity.
- Plotted ROC curve to check for the AUC and TPR-FPR trade-off, we got AUC of 0.97 which assures that our model makes good predictions.

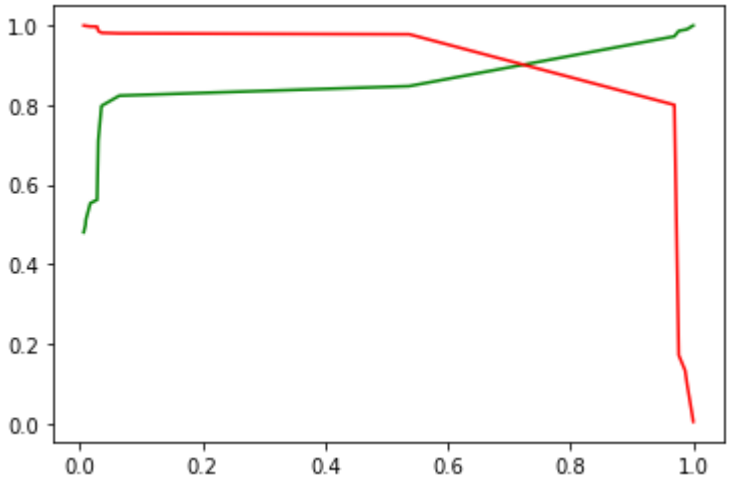


Finding optimal cutoff

- To analyze Sensitivity and Specificity trade-off, we plotted sensitivity, specificity and accuracy for 10 cut-off values between 0 to 1. We got the junction point of all these lines at **0.53**.
- Taking 0.53 as optimal cut-off we got 0.97 sensitivity and 0.83 specificity on training data.



- Also analysed precision-recall trade-off by plotting precision and recall lines for 10 cut-off values between 0 to 1. We got the junction point of both lines at 0.71.
- Taking 0.71 as optimal cut-off on train data we got precision value of 0.97 and recall of 0.80.



Prediction on Test dataset

- We took 0.53 as our optimal cutoff since business use case is inclined towards better sensitivity/recall to not miss any of the potential leads.
- After making predictions from our final model we arrived at sensitivity score of 0.979 and specificity of 0.824 with below confusion matrix.
- High true positive and low false negative counts.

```
[[815 173]
 [ 19 905]]
```

Recommendations on different scenarios

- We have provided lead scores to all the leads by multiplying predicted probabilities from logistic regression model by 100.
- In case we do not want to miss any of the potential leads and we have enough resources to make as many phone calls as required then we can go for a lead score which offers higher sensitivity. The highest sensitivity we get in our model is at the cut-off of 0.5 therefore, calls should be made to all leads with leads score greater than 50.
- In case we have shortage for resources to make phone calls we can go for a cut-off that offers higher precision. In our model precision value of 0.97 can be obtained from a cut-off of 0.71. Therefore, sales team should make phone calls to people only with lead score higher than or equal to 71.