Lead Scoring Summary

Problem Description

An education company named X Education sells online courses to industry professionals. On any given day, many professionals who are interested in the courses land on their website and browse for courses.

The company markets its courses on several websites and search engines like Google. Once these people land on the website, they might browse the courses or fill up a form for the course or watch some videos. When these people fill up a form providing their email address or phone number, they are classified to be a lead. Moreover, the company also gets leads through past referrals. Once these leads are acquired, employees from the sales team start making calls, writing emails, etc. Through this process, some of the leads get converted while most do not. The typical lead conversion rate at X education is around 30%.

Approach

From above problem description we conclude that the above problem is the classification problem, hence we choose logistic Regression to calculate the Lead rate.

Below are the steps followed to solve this problem:

1. Data Reading and Understanding
2. Data Cleaning
3. Data visualization and Outlier Treatment
4. Feature Scaling
5. Model Building
6. Model Evaluation on Train set
7. Prediction on Train Set
8. Conclusion

Data Reading and Understanding:

Here we tried to get the look and feel of the data, we observed following things:

- Number of rows and columns
- Data types of each columns
- Checking first few rows how data looks
- Checking how the data is spread.
- Checking for duplicates, if any.

Data Cleaning:

Here we checked for discrepancies in the dataset.

- Checking for any column names correction
- Checking for null values and imputing them with appropriate methods

1. We used mode imputation for categorical columns.
2. We used mean imputation for numerical columns if there is no skewness in data.
3. We used median imputation for numerical columns if there is skewness in the data.

Data Visualization and Outliers Treatment:

- We performed univariate analysis on categorical column to see which columns makes more sense and removed those columns whose variance is nearly zero.
- We performed bivariate analysis on categorical columns to see how they vary w.r.t Converted column.
- We performed univariate analysis on numerical columns by plotting box plots to see are there any outliers in the data or not.
- We performed bivariate analysis on numerical columns with Converted column to see how the leads are related to these columns.
- In this step we also plotted the correlation matrix to identify the columns which are correlated.

Feature Scaling:

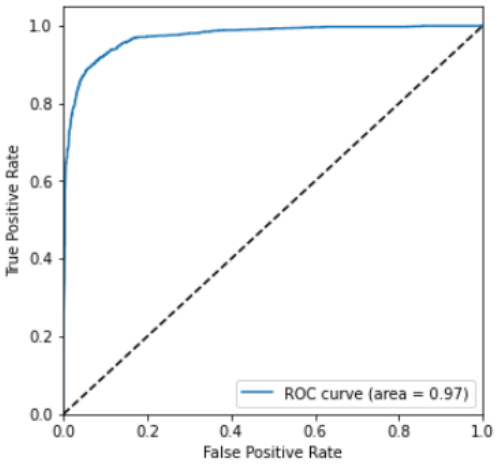
We know that logistic regression takes the input parameters as numerical values. Hence, we converted all the categorical columns to numerical.

- Columns which have only two levels “Yes” and “No” were converted to numerical using binary mapping. Columns which have more than two levels were converted to dummies using pd.get_dummies function.

Now, the data contained only numerical columns and dummy variables. Before proceeding for model building.

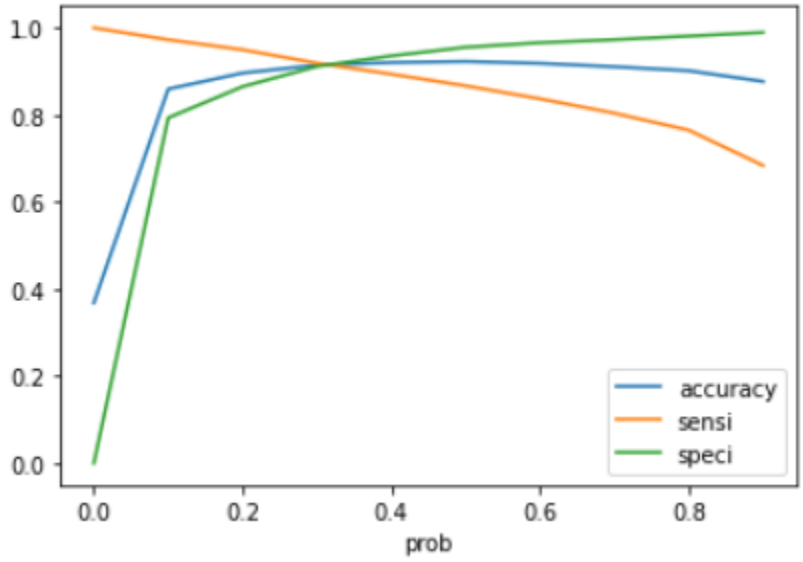
Model Building:

- We have used Recursive Feature Elimination Technique to remove attributes and built a model on those attributes that remain.
- RFE uses the model accuracy to identify which attributes (and combination of attributes) contribute the most to predicting the target attribute. In this step we made the model stable by using stats library, where we checked the p-values to be less than 0.05 and vif values to be under 5.
- Variance inflation factor(vif) is used to treat the multi collinearity. Once the stable model was created, we predicted probabilities on the train set and created a new column predicted with 1 if probability is greater than .5 else 0.
- We calculated the confusion matrix on this predicted column to the actual converted column.
- We also calculated the metrics sensitivity, specificity, precision, recall and accuracy. We also plotted roc curve to find the area under the curve.

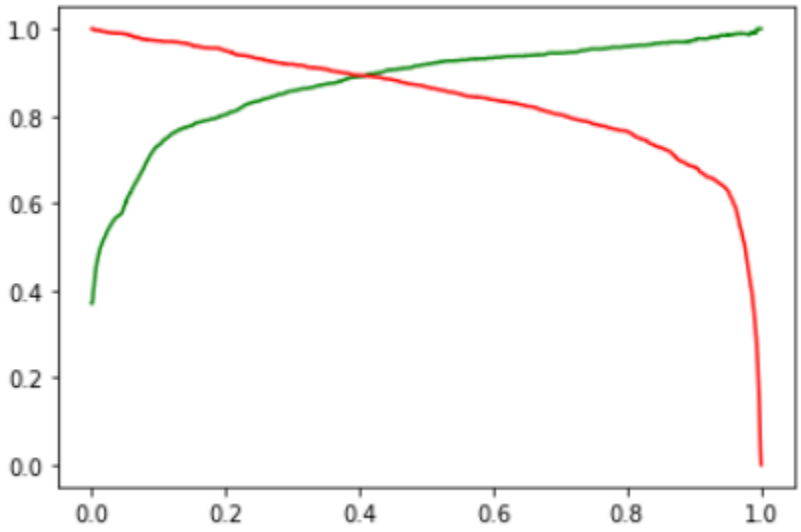


Model Evaluation on Train Set:

- We took 0.5 as the cut-of. To confirm that it was the best cut, we calculated the probabilities with different cut-offs.
- With probabilities from 0.0 to 0.9, we calculated the 3 metrics -accuracy, sensitivity, and specificity.
- To make predictions on the train dataset, optimum cutoff of 0.3 was found from the intersection of sensitivity, specificity and accuracy as shown in below figure.



- To make predictions on the test dataset, optimum cutoff was considered as obtained from Precision recall graph of the train dataset as shown below figure:



- We can observe that 0.37 is the tradeoff between Precision and Recall.

Predictions on Test Set:

After finalizing the optimum cutoff and calculating the metrics on train set, we predicted the data on test data set.

Below are the observations: After running the model on the Test Data these are the figures we obtain:

- Accuracy: 91.5%
- Sensitivity :91.8%
- Specificity: 91.3%

Below are the values for the train data:

- Accuracy: 91.40%
- Sensitivity :91.90%
- Specificity: 91.12%

Conclusion:

- Data in the 'Country' column was highly skewed and thus, was dropped from the model.
- Most of the values in the Specialization columns were missing.
- The company seems to be doing pretty good in metropolitan areas
- While we have checked both Sensitivity-Specificity as well as Precision and Recall Metrics, we have considered the optimal cut off based on Sensitivity and Specificity for calculating the final prediction.
- Accuracy, Sensitivity and Specificity values of test set are around 91%, 91% and 91% which are approximately closer to the respective values calculated using trained set.