Project Summary

Lead Score –Case Study

`

Table of contents

`

Introduction

Problem Statement :

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%.

Now, although X Education gets a lot of leads, its lead conversion rate is very poor. For example, if, say, they acquire 100 leads in a day, only about 30 of them are converted. To make this process more efficient, the company wishes to identify the most potential leads, also known as 'Hot Leads'. If they successfully identify this set of leads, the lead conversion rate should go up as the sales team will now be focusing more on communicating with the potential leads rather than making calls to everyone

Business Goal **:**

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%.

`

Steps Involved

Loading and reading the data

- Relevant library is imported
- Data is loaded
- It is read
- Data is inspected
- Check the number of rows and columns in the data frame
- Check the column-wise info of the data frame
- Check the summary for the numeric columns

Data cleaning

- We checked for the null values and dropped all the columns that contained more than 35% null values .Columns that contained null values below that were either imputed (with median values in case of numeric variable and creating new variables in case of categorical variables) or dropped based on the data.
- Checked if there were any duplicate values in the dataset and we didn't find any.
- Checked if there were columns with one unique value since it won't affect our analysis and the columns that had only one value "No" in all the rows were dropped .
- Few columns had a value called select ,we converted those values as nan since the customer has not selected any options for these columns while entering the data.

EDA

- Checked for outliers and treated.
- Performed univariate analysis and found that few data were irrelevant and hence dropped.
- Checked for correlation using heat maps.

`

Creating Dummies

- Binary variables were converted to 0 or 1.
- Dummy variables were created for categorical variables

Splitting the data into test and train

- Relevant library was imported
- Feature variable was assigned to X
- Response variable was assigned to y
- Data was split into test (30%) and train (70%)

Feature Scaling

- Relevant library was imported
- Numeric variables were scaled using StandardScaler

Model Building

- Relevant library was imported
- GLM model was built
- Feature selection was done using RFE after importing relevant library
- We selected 20 top variables
- We removed rest of the variables manually by checking the p-value(<0.05 was kept) and VIFs(< 5 was kept)

Model evaluation

- We then created the data frame having the converted probability values and we had an initial assumption that a probability value of more than 0.5 means 1 else 0.
- Based on the above assumption, we derived the Confusion Metrics and calculated the overall Accuracy of the model.
- We also checked for Sensitivity, Specificity, False Positive Rate, Positive Predictive Value and Negative Predictive Value.

`

Plotting ROC Curve

- We then tried plotting the ROC curve for the features and the curve came out be pretty decent with an area coverage of 89% which further solidified the of the model.

Model prediction

- Optimal cutoff was found by plotting accuracy sensitivity and specificity for various probabilities
- Final prediction were made using 0.37 that cut off
- Based on the new value we could observe that close to 81% values were rightly predicted by the model.
- We could also observe the new values of the ‘accuracy=81%, ‘sensitivity=80.6%’, ’specificity=81.43%’
- Also calculated the lead score and figured that the final predicted variables approximately gave a target lead prediction of 80%

Precision and Recall

- We also found out the Precision and Recall metrics values came out to be 79.61% and 70.09% respectively on the train data set.

Making Predictions on Test Set

- We implemented the learnings to the test model and calculated the conversion probability based on the Sensitivity and Specificity metrics.
- We then found out the accuracy value to be 81.80%; Sensitivity=79.55%; Specificity= 83.26%
- We also found out the Precision and Recall metrics values came out to be 75.45% and 79.55% respectively.

`

Conclusion

- 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 82%, 80% and 83% which are approximately closer to the respective values calculated using trained set.
- Also the lead score calculated in the trained set of data shows the conversion rate on the final predicted model is around 80%
- Hence overall this model seems to be good.