Case Study Summary

Lead scoring case study using logistic regression in python

Our approach to Case Study comprises of 4 parts namely Data Cleaning, EDA, Model building and testing, Metrics evaluation.

For Encoding and Scaling we decided to try out mutiple scenarios, to understand its effect on the data, those scenarios are available in the notebook.

Data Cleaning:

- Column renaming:
    a. We noticed columns that were too lengthy to read, therefore renamed it based on the context.

- Constant Features:
    a. The dataset had many single labelled categorical features, since these features contribute less to the model's performance, we decided to drop all those features.

- Missing value imputation
    a. Initially when looked at the dataset, there were a lot of unknowns, also it had a lot of features, understanding the dataset took some time,
    b. Additionally, there were a lot missing values and labels like 'select' in some features,
    c. We came to conclusion that they are of no meaning to a particular feature,
    d. We therefore had to devise a plan to either impute or drop those values, making this call took some time, in the end we dropped most of those values.

- Outlier treatment
    a. Very few features were numerical in nature, such features had few outliers. we decided to cap those outliers.

- Label Encoding/Dummy variable creation for categorical variables
    a. Since there were lot of categorical features, for one of the approach we decided to use label encoding, since label encoding converts the labels to a numerical value.
    b. whereas dummy encoding increases the features count multifold, so we tried out both the approaches, the outcome results differed significantly

- Test-train split of the data
    a. For Model Building purpose we split the data into train and test set at a ratio of 70:30

- Scaling of variables:
    a. We tried using Standardscaler as well as MinMax scaler.
    b. Since some features have different scales, Standardscaler brings all the features to the same scale.
    c. For one of the approach we used MinMax scaler and we felt standardscaler performed better in this situation.

Overall, the data cleaning took a lot of time. As we progressed towards the solution, we were able to solve most of the unknowns. therefore, it is necessary to allocate more time to understand the data and chalk out steps required, before developing the solution. Also, a little bit of flexibility is necessary to overcome any unforeseen challenges.

Exploratory Data Analysis:
For EDA we performed Univariate, Bivariate and Multivariate analysis Obesevations are provided in the notebook.

Model Building:
For this Case Study, we tried to experiment the solution in two ways.
1.One without dropping rows - by converting the null and unwanted labels as ‘unknown’, then using dummy encoding - to drop that particular label.
2.And another approach by dropping all the rows based on a particular feature and therefore reducing the dataset size to 69%

Model Building:

a. A logistic regression model was built using the function GLM() under statsmodel library.
b. This model contained all the variables, some of which had insignificant coefficients.
c. Hence, some of these variables were removed first based on an automated approach, RFE - Recursive Feature Elimination and then a manual approach based on the VIFs and p-values

Observations:
In first approach, we found out that 'tags' were having higher significance and contributing much to the prediciton of the target variable. In other approach we dropped this feature before model building, due to higher % of null values, which resutled in higher coefficients for other dummy variables. Therefore Feature selection plays an important role in the final outcome of the model.

Metrics Evaluation:

a. For the first approach, the model resulted in 90% accuracy whereas in the second aproach the model accuracy dropped to 80%, since accuracy can be biased therefore we have to evaluate other metrics as well.
b. We evaluated the sensitivity & Specificity as well as precision & recall scores using the confusion matrix. The optimum cutoff for solution 1 was around 0.30 and 0.38 and for solution 2 was around 0.43 and 0.45
c. Thererfore, It is very important to understand the business requirement based on which we have to set a cutoff to recalibrate our final outcome. Whether to have higher Precision or higher Recall scores with a drop in accuracy.

Overall, working on the case study was a great experience, it helped us understand the various challenges that one faces and what are approaches that one can use to overcome, when building a Logistic Regression model.

Thanks for reading