Lead Scoring Case Study Summary

Problem Statement:

An education company called X education which sells online courses to industry professionals is suffering from poor conversion rates to their courses. So, **they want us to build a model which helps them identify their most potential leads ('Hot leads')**, so that they can be targeted easily by the Sales team to increase their conversion rate.

Herein, **while building the model, we 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%.

Solution to Problem Statement:

- **Importing libraries-**
  Some predefined libraries of python are required to be imported to so that we can perform various analysis on the dataset and eventually build a solution to the problem statement.
- **Reading the data-**
  Read the data from the excel file provided and created a Dataframe on it.
- **Inspecting the data-**
  By inspecting the given data, we know that there are 9240 rows and 37 columns. Then we understood it's summary, data distribution, data types and null values in each of the columns.
- **Imbalance Analysis-**
  To check the balance and data with respect to the target variable - 'Converted', we performed the Imbalance Analysis
- **Data Cleaning and EDA-**
  1. First, we dropped the columns with high number of unique values.
  2. Replaced the 'Select' value in the categorical columns to NaN as these values are mostly from dropdown menus where nothing is selected
  3. Dropped/imputed columns with high number of missing values.
  4. Eventually, we dropped the columns with null percentage more than 35%.
  5. The columns with binary values like 'Yes' and 'No' and the skewed columns that have more than 90% of their values in a single category. These kinds of skewed columns will not be helpful in our analysis. So, eventually we dropped them.
  6. Performed Outlier treatment on Numerical Columns.
  7. We used count plots to get the distribution of categorical columns and to understand which categories have high conversion rate, the boxplot to analyse the distribution of numerical variables and identify the outliers and heatmap was used to get correlation between columns.
  8. Outliers were identified and capped at 99th percentile

- **Data Preparation-**
    1. The columns with binary values, ‘no’ and ‘yes’ were mapped to 0 and 1 respectively.
    2. Dummy variables were created in the place of categorical variables.
    3. The data was split into train and test set in the ratio 70:30. X_train, X_test had all predictor variables while y_train, y_test had the target variable ‘Converted’.
    4. The numerical columns were standardised using StandardScaler, for our model to perform better
- **Model Building-**
    1. Selecting top predictor variables using RFE

    The top 15 predictors were selected using RFE technique. Then, we successively removed variables having high p-values (> 0.05) and high VIF (> 2) from the logistic regression model one by one.
    The final model had all the predictors with p-values < 0.05 and VIF < 2. This model consisted of 7 variables.
    2. Predicting conversion probability

    The conversion probabilities were predicted for the train data using the final model. Initially, we used 0.5 as the threshold value of probability, to predict whether a person will convert into a lead or not.
    3. Calculating initial model metrics

    Using confusion matrix, we calculated the model metrics such as accuracy, sensitivity, specificity, false positive rate, positive predictive rate and negative predictive rate.
    Also, ROC curve was plotted to check the behaviour of model and it was closer to the y-axis, which is a good sign.
    4. Finding the optimal threshold

    Optimal threshold was identified by plotting the accuracy, sensitivity, specificity against converted probability value and finding their intersection point. We got 0.3 as the optimum threshold value i.e. if converted probability > 0.3, then it will be considered as lead.
    5. Calculating final model metrics and lead score

    With 0.3 as threshold value for converted probability, model metrics were calculated. We also calculated the lead scores by multiplying the conversion probabilities by 100. Here, higher lead scores have higher probability of getting converted i.e., they are hot leads.
- **Model Evaluation-**
    1. After preparing the test data in a similar manner to that of train data, the conversion probabilities were predicted for the test data using the final model.
    2. Finally, model metrics were calculated and compared with the train data model metrics. It turned out to be almost similar. So, the model is effective.
    3. By comparing the average scores of the converted and not converted, we found out that converted has an average lead score of around 60 and not converted has 20.

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.

- The model metrics are almost similar for both train and test datasets that shows the model is performing well. Herein the values for them
    - **Train Dataset**
        - *% of final predicted conversions on train data is **86.88%** i.e approx **87%***
        - *Accuracy : **79.69%** i.e approx. **80%***
        - *Sensitivity : **86.88%** i.e approx. **87%***
        - *Specificity : **75.11%** i.e approx. **75%***
        - *False Positive Rate: **24.88%***
        - *Positive Predictive Value: **69.02%***
        - *Negative Predictive Value: **89.97%***
        - *Precision: **69.02%***
        - *Recall: **86.88%***
    - **Test Dataset**
        - *% of final predicted conversions on test data is **86.10%** i.e approx **86%***
        - *Accuracy : **79.88%** i.e approx. **80%***
        - *Sensitivity : **86.10%** i.e approx. **86%***
        - *Specificity : **76.15%** i.e approx. **76%***
        - *False Positive Rate: **23.84%***
        - *Positive Predictive Value: **68.38%***
        - *Negative Predictive Value: **90.14%***
        - *Precision: **68.38%***
        - *Recall: **86.10%***
- Our model has a sensitivity of around 0.86 which shows it is able to correctly predict 86% of the converted leads.
- The precision of our model is around 0.69 which shows that the 69% of the leads predicted by the model are truly converted leads.
- Also the lead score calculated in the trained set of data shows the conversion rate on the final predicted model as around **87%**
- Hence overall this model seems to be good.