Lead Score Case Study Summary

Problem Statement

- X Education provides online courses to Industry Professionals. Their conversion rate is about 30%. Now, the CEO wants to assign the leads being generated to be assigned with a lead score so that they can identify which leads are 'hot' and increase the conversion rate to around 80%.
- By identifying the most potential leads as hot leads, the CEO aims to make the process more efficient and improve the current "poor" conversion rate

Our Approach

We approached this problem in a very straight forward and simple manner along the following red thread:

- Read & Understand the Data
- Data Cleaning (EDA)
- Data preparation, Pre-processing & Test-train split
- Feature selection & Building the Model
- Determining Optimal-Threshold
- Confusion Matrix & Performance Parameters
- Making Predictions

Read & Understand the Data

Beginning with uploading the file into "leads" dataframe and basically tried to get an understanding of the dataset at hand. WE checked the dimensions, datatypes, missing values, few rows& columns. Here's when we saw that there are many columns with "Select" as one of the values in it. On inspecting, we concluded that it as good as a null value because when we look at CRMs that sales professionals use, one of the options is select. Select is not actually an option but maybe was stored because no option was selected. Therefore we replaced that with NaN.

Data Cleaning

We started of with missing value treatment wherein we proceeded with columns in order of descending percentages of missing values. All the columns with more than 40% of missing values were removed, after inspection. For rest of the columns, below 40% missing values, categorical columns were analysed using countplots whereas numerical columns were explored using distplots/boxplots. When dealing with the former, missing rows were either removed or replaced by mode value. And with the latter, we used mean/median depending on the column itself.

In the next steps, univariate/bivariate/multivariate and outlier was conducted. Columns like 'Notable Last Activity' & 'Last Activity' were almost similar, therefore, one of them was dropped (one without any missing values). During Bivariate analysis we witness that Lead Source, Tags, Specialization etc did have some good association with the target.

NOTE: Tags column- "Lost to EINS" & "Lost to horizon" were dropped during EDA because the first time we created the model, those two were highly contributing to the probability of conversion of the lead. However, EINS & Horrizon are competitor and that doesn't make sense, so we dropped the rows where Tags had them as an entry.

Data preparation, Pre-processing & Test-train split

Once the data was clean, categorical columns with binary values, their values were replaced by 1 and 0. The ones with more than 2 levels, were converted into discrete values using pd.get_dummies(). After that, the data was first split into X (only independent variables) and y (target variable). Using these two, the complete dataframe was divided into test and train datasets for model preparation. AS the last step for this section, the numerical columns were scaled using standard scaler to start building the model.

Feature selection & Building the Model

Since there were so many feature post the previous step, the first step was to extract the initial 20. The same was done using the RFE (Recursive Feature Elimination) technique. Followed by manually eliminating features on the basis of their significance (p value) & VIF values (multicollinearity). The process was repeated till all the features in the model had a pvalue of less than 0.05 and VIF value under 5. In the end, 16 features were shortlisted. The sequence of elimination of variables

First	High p value	High VIF
Second	High p value	low VIF
Third	Low p value	High VIF
Last	Low p value	low VIF

Determining Optimal-Threshold

Firstly, we made the ROC curve to check the performance of the model. The area under the curve was 0.97 indicating the model is good. Following this, we proceeded to plot the True Positive rate vs False Positive rate to see where the two graphs intersect. The intersection was taken as the threshold value and the performance parameters we checked for 0.22, 0.23, and 0.24. WE conclude, 0.22 is the suitable one for our business problem where the Sensitivity was also in the required range. We looked closely at the Sensitivity because in order to increase the conversion, our model should correctly predict the potential leads. Post that we ensure the same from the Precision vs Recall trade-off plot.

Confusion Matrix & Performance Parameters

The predictions were made on both the train set and the test set and since the outcome was catering to the business problem we wanted to solve, we proceeded to finalise it.

The performance parameters obtained from our model:

Train Set		Test Set	
Sensitivity	90.8 %	Sensitivity	91.2 %
Specificity	91.1 %	Specificity	90.8 %
Recall	84.7 %	Recall	83.2 %