Lead Scoring Case Study Summary

Problem Statement:

X Education specializes in offering online courses to professionals within various industries. The company is seeking assistance in identifying the most favorable leads – those with the highest potential to become paying customers. They require a system that assigns a lead score to each potential customer, reflecting the likelihood of their conversion. The goal is to prioritize leads with higher scores, as they are more likely to convert into paying customers. Conversely, leads with lower scores are expected to have a lower chance of conversion. TheCEO has indicated a rough target lead conversion rate of approximately 80%.

Solution:

Step 1: Reading and Understanding Data.

- We imported the necessary packages and dataset.
- We inspected the dataset by checking its shape, null values, data types, and statistical summary.

Step 2: Data Cleaning:

We cleaned the data by:

- Dropping variables with a high percentage of missing values.
- Imputing missing values with median values for numerical variables and creating new classification variables for categorical variables.
- Identifying and removing outliers.

Step 3: Data Analysis

- We performed exploratory data analysis (EDA) on the dataset to get a better understanding of its structure and distribution.
- We identified three variables that had only one unique value in each row. These variables were dropped because they did not provide any additional information and could potentially skew the results of our analysis.

Step 4: Creating Dummy Variables

- We converted the binary variables in the dataset from "Yes" or "No" to 1 or 0, respectively.
- We then created dummy variables for the categorical variables in the dataset. This involved creating a new binary variable for each category in the original variable.
- We dropped the first dummy variable for each categorical variable, as this variable is essentially redundant.
- Finally, we concatenated the dummy data dataframe to the main data dataframe and dropped the columns for which we had created dummy variables.

Step 5: Test Train Split:

- We split the dataset into a training set and a test set.
- We used a 70-30 split, which means that 70% of the data was used for training and 30% was used for testing. This is a common approach that is used to ensure that the model is not overfitting to the training data.

Step 6: Feature Rescaling

- We used min-max scaling to scale the original numerical variables. This ensures that all of the variables are on the same scale, which can improve the performance of the model.
- We then used the statsmodels package to create our initial model. This model gives us a complete statistical view of all of the parameters of the model, which can be helpful for understanding how the model works and for making predictions.

Step 7: Feature selection using RFE:

- We used recursive feature elimination (RFE) to select the most important features in the dataset. RFE is a method of feature selection that starts with all of the features in the dataset and then recursively removes features that are not significant.
- We used RFE to select 35 features, which we then evaluated using p-values. We dropped features with p-values greater than 0.05, which left us with 15 features.
- We then calculated the variance inflation factor (VIF) for each of the remaining features. VIF measures the collinearity between features, and we wanted to ensure that the features were not too collinear. We dropped features with VIF greater than 5, which left us with 10 features.
- We then created a dataframe with the converted probability values. We assumed that a probability value of more than 0.5 means 1 and a probability value of less than 0.5 means 0.
- We then calculated the confusion matrix and the accuracy of the model. The confusion matrix shows the number of true positives, false positives, true negatives, and false negatives. The accuracy of the model is the percentage of predictions that were correct.

- We also calculated the sensitivity and specificity of the model. Sensitivity is the proportion of true positives that were correctly identified, and specificity is the proportion of true negatives that were correctly identified. These metrics can be used to understand how reliable the model is.

Step 8: Plotting the ROC Curve

- We then plotted the receiver operating characteristic curve (ROC curve) for the features. The ROC curve is a plot of the true positive rate (TPR) against the false positive rate (FPR). The TPR is the proportion of true positives that were correctly identified, and the FPR is the proportion of false positives that were incorrectly identified.
- In our case, the ROC curve was pretty decent with an area coverage of 90%. This means that the model was able to correctly identify 90% of the true positives and 90% of the true negatives. The area coverage of 90% further solidified the reliability of the model.

Step 9: Finding the Optimal Cutoff Point

- We then plotted the probability graph for the accuracy, sensitivity, and specificity for different probability values. The intersecting point of the graphs was considered as the optimal probability cutoff point. The cutoff point was found to be 0.37.
- Based on this new value, we could observe that close to 80% of the values were correctly predicted by the model. We could also observe the new values of the accuracy, sensitivity, and specificity. The accuracy was 81.5%, the sensitivity was 80.2%, and the specificity was 82.29%.
- We also calculated the lead score and found that the final predicted variables approximately gave a target lead prediction of 80%.

Step 10: Computing the Precision and Recall metrics

- We calculated precision and recall metrics on the training dataset. Precision is the proportion of true positives that were correctly identified, and recall is the proportion of true positives that were identified. The precision and recall metrics were 80% and 71.4%, respectively.
- A high precision indicates that the model is not predicting many false positives. A high recall indicates that the model is not missing many true positives
- In our case, the precision was 80%, which means that 80% of the predicted positives were actually positive. The recall was 71.4%, which means that 71.4% of the actual positives were predicted positive.
- Based on the Precision and Recall trade-off, we got a cut off value of approximately 0.41

Step 11: Making Predictions on Test Set

- We then implemented the learnings from the training model to the test model. We calculated the conversion probability based on the sensitivity and specificity metrics, and found that the accuracy value was 81.39%, with a sensitivity of 79.4% and a specificity of 82.58%.

We believe that this model can be used to improve the efficiency of our lead generation process by helping us to identify leads that are more likely to convert. This can save us time and money, as we will be able to focus our resources on leads that are more likely to be successful.