Lead Score Case Study Summary:

Problem Statement: The company requires you to build a model wherein you need to assign a lead score to each of the leads such that the customers with a higher lead score have a higher conversion chance and the customers with a lower lead score have a lower conversion chance. The CEO has given a ballpark of the target lead conversion rate to be around 80%.

Dataset: In the dataset we have 9240 rows and 37 columns. In this dataset 'Converted' is our target variable. We saw that there are few column values as 'Select' for the ones which the customers have not selected any values, so we converted these values to null.

Data Cleaning: There were a few columns with about 20%,30%, 50% and 70% missing values. We replaced the null values with a new category or existing category or even dropped it was not necessary.

Exploratory Data Analysis: First, we checked the 'Converted' ratio which 37.8% which is pretty decent. Then, we analyzed each column one by one. While doing so we found a couple of important columns like 'Lead Origin', 'Lead Source', 'Do Not Email', 'Total Time Spent on Website', 'Last Activity', 'What is your current occupation', 'Tags'. We found that these mentioned columns made more sense in increasing the potential lead conversion. We found a few columns which do not have a proper spread of data. These columns categories are focused on just one particular category, therefore these columns do not provide much information while model building, so we dropped these columns.

Data Preparation: We mapped the few columns to '0' and '1'. Then we created dummy variables for categorical columns. We created a train test split. Next, we scaled our numerical variables using StandardScaler.

Model Building: We used Logistic Regression and for feature selection we used RFE. We took 20 columns with the help of RFE. Then we used stats model to analyze the how the columns were doing in our model by looking at the p-value. We found few columns whose p-value was greater than 0.05 so we dropped these columns and again created new model. Next, we checked the VIF of the columns and found a column whose VIF was greater than 5 so, we dropped the column and again created the model. After this the model looked good so we started predicting our y_train. First, we decided to take cut-off of 0.5 but after analyzing the ROC curve and optimal threshold we concluded to take the cut-off 0.3. Next, we calculated the Lead Score. With cut-off of 0.3 we got accuracy of 92%, Specificity: 0.927, Sensitivity:0.91, False Positive Rate:0.07, positive predictive value: 0.88, Negative predictive value:0.94 on the training dataset. Next, we calculated and got Precision as 0.88 and Recall as 0.91. Next, we calculated the Precision and Recall Trade-off.

Making Prediction on Test Dataset and Model Evaluation: At last, we applied our model to the test dataset and got a pretty good accuracy of 92%, Sensitivity: 0.921 and Specificity: 0.926

Conclusion: We have built a logistic regression model and assigned a lead score between 0 and 100 to each of the leads which can be used by the company to target potential leads. A higher score would mean that the lead is hot, i.e. is most likely to convert whereas a lower score would mean that the lead is cold and will mostly not get converted. The top three variables in your model which contribute most towards the probability of a lead getting

converted are Total Time Spent on Website, Lead Source, Lead Origin. The top 3 categorical/dummy variables in the model which should be focused the most on to increase the probability of lead conversion are Last Notable Activity_SMS Sent, Do Not Email, Last Notable Activity_Modified