Project Report

Description of design choices and Performance evaluation of the model:

Design Choices:

The problem statement is about predicting whether a customer will respond to marketing efforts directed towards them by making a purchase. Therefore, this is a **classification problem** where each potential customer needs to be assigned to either of two groups, **will respond**, and **will not respond**.

Two very popular and robust classification methods are **logistic regression** and **XGBoost** (Generally accepted as best among all ensemble techniques).

We faced difficulties in training the Logistic regression model due to **perfect separation** of the training sample into the two groups in consideration (responded vs not responded) which led to formation of **singular matrix**. Therefore, we proceeded to complete our task of building a classification model with the help of XGBoost.

Even with XGBoost, there was no impact of **hyperparameter tuning** since any combination of the hyperparameters was yielding a perfect train and test **AUC score (Area Under Curve)** of **1**. Therefore, we decided to keep the hyperparameters at levels which would make the model more generalized/simple.

Had there not been perfect separation, and effect of changing the hyper-parameters could be observed, we could have created multiple logistic regression and XGBoost models, recorded the model performance metrics in a **performance evaluation score card**, and have chosen the model with the best score for **AUC, recall, accuracy, or F1 score** as per our needs

Performance Evaluation:

We have used **AUC score** as an evaluation criteria. Since we neither want to lose any leads, not do we want to spend marketing budget on people whose probability of response is very low, we would want to have both; a low false positive rate, and a low false negative rate. Therefore, AUC represents it most appropriately. We also could have used **F1 Score.**

Discussion of future work:

1. Code for predicting the target variable using the trained model should be separated, and deployed on a cloud platform (Example: AWS Sagemaker). The model can be deploy to predict on a **real-time basis** basis, or on a **batch-transform basis**, depending on the business needs. In our case, leads could be bundled together, and batch-**transform** could be used.

2. Code for training the model needs also needs to be deployed using **CI-CD pipeline** such that **continuous improvement and model training** is possible.

How does this benefit the insurance companies:

Getting predictions about leads whether they will purchase or not can save the insurance company time and money by:

1. Directing marketing efforts towards leads who are likely to purchase as per the predictions. This would result in better acquisition of customers and drive revenue growth.
2. Saving marketing expense by not pursuing leads who are not likely to purchase as per the predictions.
3. Saving time (and therefore costs) by not pursuing leads who are not likely to purchase.