## **Machine Learning Pilot**

### **Objective:**

Predict the optimal discount level that maximizes the probability of winning sales deals within each customer cluster. Explore the impact of varying discount levels and discount variance bands on the probability of winning or losing deals for different customer and deal size clusters. Analyze the factors that significantly influence these probabilities and whether these factors work independently or in combination.

### **Scenario**

The dataset is drawn from 48 months of sales deal data, including a forecast for 2023. Each record corresponds to a sales deal, showing whether it was won or lost, along with associated attributes such as deal size, subscription period, subscription size, product, region, segment, sales team, route to market, and more. Deals are grouped by size, customer segment, solution, and region. Each deal's outcome (win or loss) is further qualified by a "Stage," providing qualitative information on the nature of the win or loss.

### **Key Variables**

The key variables for this analysis are the discount band and the discount variance band. The discount band represents ranges like 22-27%, 28 to 34%, 34 to 39%, 40 to 44%, etc. Discount variance represents the level of change from the planned discount - 0 - 3%, 3.1 to 5%, 5.1 to 7%, 8.1 to 10%, Above 10%.

### **Model Output**

The ML model(s) should predict the outcomes listed in the objectives and allow for backward testing (i.e., historical accuracy) and forward forecasting (i.e., future prediction).

## **Model Development Process**

Objective is to develop a ML process like a framework of repeatable steps.

While the actual models and the actual methods may vary from customer to customer based on the data and the influencing factors, our objective is to develop a systematic step by step process that can be repeated and the output of each step clearly recorded to show to the business customer and the other experts what is the outcome of that step.

### **Data Preprocessing and Feature Engineering**

Ensure that the categorical features are converted into numerical values and handle missing data, outliers, and erroneous entries. Apply feature engineering to create new features or transformations that better represent the underlying patterns in the data.

### **Model Selection and Training**

Use cross-validation during the model training phase to get a better estimate of model performance on unseen data and assist in hyperparameter tuning. Consider linear models, tree-based models, ensemble models, and possibly even neural networks, documenting the performance of each on the training and validation data.

### **Model Evaluation**

Establish a baseline model for comparison. Choose an evaluation metric aligned with your project needs (accuracy, precision, recall, F1 score, ROC AUC score, etc.) and apply this to both backward testing and forward forecasting.

### **Model Interpretation and Feature Importance**

Based on the chosen ML model, interpret the model's predictions and ascertain feature importance. Tree-based models can inherently measure feature importance, while linear models can provide insights through coefficients. Use tools like SHAP (SHapley Additive exPlanations) to understand complex models better.

### **Model Deployment and Monitoring**

Deploy the model to a production environment and set up an infrastructure for it to receive data, return predictions, and retrain with new data. Monitor model performance over time and retrain the model regularly to ensure it stays relevant as data patterns evolve.

Overall, the model building process should be modular, structured, repeatable, and not a one-off process. The process should continuously improve itself by learning from past performance and adjusting accordingly for future predictions.