Churn Prediction Model Report

➢ Introduction

This report documents the process of building a churn prediction model using the Random Forest algorithm. The goal is to predict customer churn based on a given dataset. The process involves preprocessing the data, handling outliers, splitting the data, standardizing features, training the model using GridSearchCV for hyperparameter tuning, and evaluating the model's performance.

➢ Code Overview

1. Data Preprocessing and Outlier Handling –

- The dataset is loaded using pandas.Numerical features are identified, and Z -scores are calculated to detect outliers.
- Outliers are identified using a specified Z-score threshold. Outliers are handled by capping their values to the 95th percentile.

2. Data Splitting –

- The preprocessed data is split into training and testing sets using train_test_split.

3. Feature Standardization –

- Numerical features in the training and testing sets are standardized using StandardScaler.

4. Categorical Feature Encoding –

- Categorical features are identified.
- The ColumnTransformer is used to apply scaling and one-hot encoding to different subsets of features.

5. Model Training and Hyperparameter Tuning –

- A Random Forest Classifier is initialized. A parameter grid is defined for hyperparameter tuning using GridSearchCV. GridSearchCV is used to find the best hyperparameters using cross-validation.
- The best model is selected from GridSearchCV results.

6. Model Evaluation –

- The best model is used to predict outcomes on the test set. Accuracy is calculated using accuracy_score.
- The confusion matrix is calculated using confusion_matrix.
- The classification report, including precision, recall, F1-score, and support, is generated using classification_report.

Note: I am familiar with the various model deployment techniques but not yet used and being very less time for the assignment submission deadline I was not able to utilize time and deploy my model as I am a working professional.