Report: Customer Churn Prediction

Submitted By:
Adarsh Kumar
+91 7484940432
theadarshkr@gmail.com

Brief Summary, and Steps:

1. Data Preprocessing and Cleaning: First of all, we read the data and removed unnecessary variables like Name and CustomerID.Then, we checked if any null values were present or not. There were no null values in our data which means our data was clean. Then, we described the data to check for the presence of outliers.And then, we plotted different distributions to check for anomalies in our data. We also plotted churn with respect to different variables to understand customer churning patterns.

- From the values of maxima and minima in the data description for the five variables, it was evident that the values lie between an acceptable and reasonable range for these variables, and these are 'Age'[18,70], 'Subscriptions_Length_Months'[1,24], 'Monthly_Bil'[30,100], 'Total_Usage_GB'[50,500],and 'Churn'[0,1].The ranges were already within the expected domain for these attributes, which means that any values within these ranges were not inherently outliers.
- The presence of outliers might not be a significant concern if the ranges align with the context of our dataset and if the data follows a natural distribution. That's why we visualized our data using plots to get a better sense of the distribution and any potential anomalies or irregularities in the plots.
- From the obtained visualization, we could say that for variables like 'Monthly_Bill', and 'Total_Usage_GB' seemed to have a relatively uniform distribution, meaning the data points were spread out relatively evenly across the range of values.But for 'Age' and 'Subscription_Length_Months',there was increase in frequency at some intervals,to a fixed value.The histograms for 'Age' and 'Subscription_Length_Months' showed two peaks,with one peak having a much higher frequency around 5600-5700 for 'Age', and around 8000 for Subscription_Length_Months' and other being smaller around 3700-3800 for 'Age' and around 4000 for 'Subscription_Length_Months'.
- We had two dominant peaks in the distribution of 'Age' and 'Subscription_Length_Months', indicating the presence of two major clusters or segments in our data.We could apply K-Means clustering to these variables to identify and label different segments based on their behaviors.By grouping customers with similar age or subscription patterns together, we could create a new feature that captures these behaviors.And in our 'feature engineering' process, we would include this

feature from clustering,to capture and utilize these patterns, and enhance the overall effectiveness of our churn prediction model.

2. Feature Engineering: a. In the first step, we introduced a new feature called,'Cluster' Because of the two dominant peaks utilizing KMeans Clustering. And that's why we had chosen 2 as the number of clusters.

b. Then ,we applied MinMaxScaler to bring the entire data between [0,1] range, then we did feature selection, selecting only those features which helped increase our recall, and accuracy and found the currently included features in our input variable X to be the most appropriate. Then, we splitted the data into train and test data in a ratio of 8:2 respectively, to train our model, then test it.

3. Model Building and Evaluation: a. Here, we utilized artificial neural network as our machine learning algorithm because different algorithms like logistic regression , random forest regression, gave a recall less than 50 and almost similar accuracy so that was the last algorithm which seemed appropriate.First of all,though we knew the shape of our training data but we confirmed it to put the input shape in first layer of model architecture.

b. We defined the model architecture and compiled it using adam optimizer and binary_crossentropy as loss because of the binary nature of output,then we trained the model and predicted on the test data. And finally we evaluated the model performance using accuracy, precision, recall, F1-score.

4. Model Optimization: a. We had tried to optimize the model using Keras Tuner which was used to perform hyperparameter tuning for our Keras model. It searched for the best hyperparameters to optimize the F1 score. In this step, first we defined a custom F1 metric class using TensorFlow's Metric base class. This class calculates F1-score during training.

b. Then, we defined a function build_model(hp) that constructed the model architecture. It was parameterized with hp (HyperParameters) from Keras Tuner.And we created an instance of Objective to define the optimization objective with the goal is to maximize the F1-score.We created a RandomSearch tuner instance with the defined build_model function, the objective, and other tuning parameters and also retrieve the best model architecture found by the tune.

c. Then,we ran the tuner's search method to explore the hyperparameter search space and find the best hyperparameters, and used the best model to predict on the test data and calculate various evaluation metrics like accuracy, precision, recall, and F1-score.And then plotted the confusion matrix and the performance metrics.

5. Model Deployment: To deploy the model, we used Flask, a popular web framework for Python.The code deployed our machine learning model using Flask, allowing it to take new customer data as input and predict whether a customer was likely to churn or not based on selected features. This is achieved through a server that receives JSON data and returns churn predictions using the pre-trained model.