SunbaseData Internship Project

Customer Churn Prediction

Name : Urja Kumari
Date : 17-09-23

Objective: The aim of this project is to predict the Customer's Churn by implementing various machine learning and deep learning algorithms on the dataset that is composed of eight independent features. It is a supervised binary classification problem where the feature 'Churn' contains value either 0 or 1 depending on whether the customer Churned or not.

Highlights of the Work Done:

- Imported Required Libraries
- Data Analysis
- Data Visualisation
- Models Implementation
    - Supervised Machine Learning Models:
      - Logistic Regression (Cross Validation)
      - KNearest Neighbors (Used Elbow Method to find Best Value of K)
      - Decision Tree Classifier
      - Random Forest Classifier (HyperParameter Tuning Done)
      - Support Vector Machine
    - Deep Learning Model
      -Artificial Neural Network (Using Tensorflow Keras-Functional API)

- Compared Models Performance through Confusion Matrix, Classification Report and Accuracy Score Metrics
- Conclusion
- Model Deployment Tried Using Flask and HTML

About the Dataset:

- The dataset consists of eight independent features,namely ‘Name’, ‘CustomerID’,’Age’,‘Gender’,‘Location’,‘Subscription_length_months’,’Monthly_Bill’,’Total_Usage_GB’ and one dependent feature called ‘Churn’.
- There are a total of 100,000 records.
- There are no null values present in the dataset along any feature so there seems to be no need of preprocessing for this part.
- Features like ‘CustomerID’ and ‘Name’ are just the representation of record entries and hence seem not important for prediction.
- ‘Age’ column contains continuous integral values ranging from 18 to 70. ‘Subscription_length_months’ also contains values ranging from 1 month to 24 months. ‘Monthly_Bill’ ranges from 30 to 100. ‘Total_Usage_GB’ ranges from 50 to 500.
- There are two features ‘Gender’ and ‘Location’ consisting of categorical values and hence can be encoded to bring them into a form compatible for machine learning models. Gender consists of two values ‘Male’ and ‘Female’ and Location consists of five different values which are ‘Los Angeles’, ‘New York’, ‘Houston’ ,’Miami’, and ‘Chicago’.

Insights from Data Visualisation:

- The Dataset is well balanced since the number of ‘Churned’ and ‘Not Churned’ are around the same (~50000).
- The number of subscriptions in each month seems to be the same. Also the frequency of subscriptions cannot be distinguished on the basis of target variable’s values, i.e, the frequency of such subscriptions comes out to be the same each month and for Churned and Not Churned.
- The data usage seems to be the same for each categorisation of data and by both Churned and Not Churned number of customers which is a bit surprising.
- The number of males and females are almost the same, around ~50000. The number of males who churned, number of males who did not churn, number of females who churned, and the number of females who did not churn are all the same which is 25000 for each category.
- Mean value and the values for the other three quartiles of the Monthly Bill seem to be the same for the customers who churned and the customers who did not churn. Also this could not be distinguished based on gender.

- The distribution of total data usage is the same for the customers who churned and the customers who did not churn and same for both the gender as well.
- The distribution of Age of Customers seemed to be the same for both the Churned and Not-Churned Customers and also for both the genders.
- The count of Churned and Not-Churned number of customers are almost the same from all the five different Locations.
- The dataset does not show a linearly increasing relationship between total data usage and the monthly bill as was expected.
- The joint distribution of Total Data Usage and Monthly Bill based on the Churned and Not-Churned customers does not show any relationship and seems random and balanced.

Important Deduction from Data Visualisation:

The Dataset seems to be completely balanced along each feature and there is no inherent relationship between different features, even the combination of features which are expected to be related. The data seems to be random and balanced along each combination of categories possible. Therefore it will be difficult for any machine learning algorithm to predict the target due to the incapability to find any insight or pattern from the data.

Data Preprocessing:

- 'Gender' consists of categorical values namely 'Male' and 'Female' so these are one hot encoded to form two columns with names 'Male and 'Female' consisting of values 0 and 1.
- 'Location' consists of five categorical values so it is one hot encoded to form 5 columns namely 'Houston', 'New York', 'Chicago', 'Los Angeles', and 'Miami' containing values either 0 and 1.
- Features like 'CustomerID' and 'Name' are dropped from the dataset since they are just a representation of record entries and could not be useful in target value prediction.
- The dataset does not contain null value across any feature and there are no outliers, hence no preprocessing needs to be done along those lines.

Model Implementation:

This is a Binary Classification Problem and hence different Supervised Machine Learning Models like Logistic Regression, Decision Tree Classifier, Random Forest

Classifier, Support Vector Machine, and K-Nearest Neighbors are implemented on the preprocessed dataset. Deep Learning Model Artificial Neural Network was also implemented.

- Logistic Regression : Model implemented on two different datasets. One containing all the features after preprocessing and one containing only continuous valued features. Then Cross Validation was also implemented separately but the accuracy obtained was best obtained through the original dataset and even that was low.
- K-Nearest Neighbors : This model was first implemented using a base value of K=1. Then the best value of K was determined using the Elbow Method. The model was then implemented using the determined value of K. Although the accuracy improved with the new value of K, it was still not good enough than random guessing.
- Decision Tree Classifier: The Model performed more or less similar to the other models and accuracy obtained was poor.
- Random Forest Classifier : The model was first implemented with assumed parameters and the accuracy was low. Then hyperparameter tuning using Grid Search was done to determine the parameters. Although the accuracy improved by very less.
- Support Vector Machine : The model was first implemented using default parameters and the accuracy obtained was low similar to the all other models
- Artificial Neural Network: The model was implemented considering three hidden layers with 32, 64, and 128

Model Performance:

Performance of Supervised Machine Learning Algorithms:

MODEL	PRECISION	RECALL	F1-SCORE	ACCURACY
Logistic Regression	[0]: 0.51	[0]: 0.59	[0]: 0.54	0.503
	[1]: 0.50	[1]: 0.42	[1]: 0.45	
Decision Tree	[0]: 0.50	[0]: 0.51	[0]: 0.51	0.499
	[1]: 0.50	[1]: 0.49	[1]: 0.49	
Random Forest	[0]: 0.51	[0]: 0.55	[0]: 0.53	0.502
	[1]: 0.50	[1]: 0.46	[1]: 0.48	

KNN	[0]: 0.51	[0]: 0.60	[0]: 0.55	0.506
	[1]: 0.50	[1]: 0.41	[1]: 0.45	
Support Vector Machine	[0]: 0.50	[0]: 0.77	[0]: 0.61	0.501
	[1]: 0.49	[1]: 0.23	[1]: 0.32	

Performance of Artificial Neural Network

- Train Accuracy : 0.5013
- Validation Accuracy: 0.5042

Conclusion

All the models gave accuracy around 50% which is no better than random guessing. Support Vector Machine gave high recall value for class [0], i.e, class of customers who did not churn, hence its better than other models which gave around 50% for both the classes. At Least it can identify more instances belonging to the 'Not- Churned' category. Hence for Model Deployment I have used the support vector model.

By Data Analysis and Data Visualisation we know that the dataset is perfectly balanced along all the features and different possible feature combinations. Hence the models can't learn about any patterns to make the prediction. The accuracy thus obtained through all the models hovered around 50% which is no better than random guessing. Careful observation shows that the classes are non-separable and hence it simply becomes a complex task to improve the accuracy. There may be different ways possible to deal with this kind of problem:

- Maybe the target value is mislabeled.
- Maybe there are more features required for prediction.
- Different machine learning paradigms like semi-supervised learning, transfer learning, and clustering might be better suited for such problems.
- We might need to create custom loss functions that penalise specific types of errors more heavily.
- We might need to involve domain experts who could provide valuable insights.