Customer Churn Prediction using Decision Tree

Introduction

This project aims to predict customer churn for a telecom company. Churn refers to when a customer leaves the company. Using historical data, we can identify which customers are at risk of leaving and take preventive actions.

What is Customer Churn?

Customer churn is the percentage of customers that stop using a company's product or service during a certain time frame. It is an important metric for businesses since retaining existing customers is cheaper than acquiring new ones.

How can customer churn be reduced?

Churn can be reduced by improving customer service, offering loyalty programs, providing discounts, customizing contracts, and proactively identifying customers at risk of leaving.

Objectives

1. Analyze customer behavior using historical data. 2. Identify key factors influencing churn. 3. Build a Decision Tree model to predict churn. 4. Evaluate model performance using Accuracy, Precision, Recall, and F1-Score. 5. Provide business insights to reduce churn.

Loading libraries and data

We imported essential libraries such as pandas, numpy, matplotlib, and scikit-learn modules. The Telco Customer Churn dataset was loaded and cleaned. Categorical values were encoded and missing values handled.

Understanding the data

The dataset includes features such as gender, age, contract type, tenure, charges, complaints, internet service, and the target variable 'Churn'. The target variable was mapped as: Yes = 1, No = 0.

Visualize missing values

Missing values in 'TotalCharges' were converted to numeric and imputed using the median strategy. Categorical missing values were imputed with the most frequent category.

Data Manipulation

Removed customerID as it does not contribute to prediction. Numeric and categorical columns were separated for preprocessing. OneHotEncoder was applied to categorical variables.

Data Visualization

Graphs such as churn distribution, ROC curve, and feature importances were plotted to understand the dataset and model behavior.

Data Preprocessing

We used ColumnTransformer to apply different preprocessing for numeric and categorical variables. Numeric values were imputed with the median, and categorical values with the most frequent, followed by one-hot encoding.

Standardizing numeric attributes

Numeric attributes were processed using imputation but not scaled, since Decision Trees are scale-invariant and do not require standardization.

Machine Learning Model Evaluations and Predictions in Decision Tree Model

A Decision Tree Classifier was trained using GridSearchCV for hyperparameter tuning. The best model was evaluated using metrics such as Accuracy, Precision, Recall, F1-Score, and ROC AUC. The confusion matrix and feature importances were also generated for better insights.