Churn Prediction Project Documentation

- Introduction

Customer churn is a significant issue for businesses, especially in the subscription-based service industry. Churn prediction aims to identify customers who are likely to stop using a service, enabling businesses to take preventive measures to retain them.

By leveraging machine learning techniques, we can develop a model that predicts which customers are at high risk of leaving, based on historical data and customer behaviour.

This project focuses on building a churn classification model for a telecom company. The dataset contains various customer attributes such as gender, tenure, internet service provider, payment method, and more, alongside the target variable "Churn," which indicates whether a customer has left the service or not. The goal is to use these features to predict customer churn, allowing the marketing team to take actionable steps in retaining high-risk customers.

- Dataset Overview

The dataset consists of 21 variables and 7043 observations. These variables describe different aspects of customer behaviour and demographics. Below is a description of the columns in the dataset:

- CustomerId: Unique identifier for each customer.
- Gender: Gender of the customer (Male or Female).
- Senior_Citizen: Whether the customer is a senior citizen (1 for Yes, 0 for No).
- Is_Married: Whether the customer is married (Yes or No).
- Dependents: Whether the customer has children or dependents (Yes or No).
- Tenure: Number of months the customer has been with the company.
- Phone_Service: Whether the customer has phone service (Yes or No).
- Dual: Whether the customer has both phone and internet service (Yes or No).
- Internet_Service: The customer's internet service provider (DSL, Fiber optic, or No).
- Online_Security: Whether the customer has online security (Yes or No).
- Online_Backup: Whether the customer has online backup (Yes or No).
- Device_Protection: Whether the customer has device protection (Yes or No).
- Tech_Support: Whether the customer receives technical support (Yes or No).
- Streaming_TV: Whether the customer has streaming TV (Yes, No, or No Internet service).
- Streaming_Movies: Whether the customer has streaming movies (Yes, No, or No Internet service).
- Contract: The type of contract the customer has (Month-to-month, One year, or Two year).

- Paperless_Billing: Whether the customer has opted for paperless billing (Yes or No).
- Payment_Method: The customer's payment method (Electronic check, Postal check, Bank transfer (automatic), or Credit card (automatic)).
- Monthly_Charges: Amount charged to the customer monthly.
- Total_Charges: The total amount charged to the customer over their tenure.
- Churn: Whether the customer has churned (Yes or No).

The target variable is Churn, which indicates whether the customer is still with the service or has left.

- Data Preparation

The data preparation process involves several important steps to ensure the dataset is ready for machine learning models. These include data cleaning, handling missing values, feature encoding, and scaling numerical features.

- Missing Values Handling:

  Customers with a tenure of 0 months were identified as new customers, and for these customers, the Total_Charges were imputed as 0, as they haven't been charged yet.

- Feature Encoding:

  Categorical features such as Gender, Senior_Citizen, Is_Married, Dependents, Phone_Service, Dual, Internet_Service, Online_Security, Online_Backup, Device_Protection, Tech_Support, Streaming_TV, Streaming_Movies, Contract, Paperless_Billing, and Payment_Method were encoded using Label Encoding for binary features and One-Hot Encoding for multi-class categorical variables.

- Handling Outliers:

  Outliers in the Total_Charges and Monthly_Charges columns were identified and handled appropriately. These outliers are important to investigate as they may represent long-term or high-spending customers who churned.

- Skewed Distributions:

  The numerical columns, such as Tenure and Total_Charges, exhibited right-skewed distributions. Log transformations were applied where necessary to normalize the data and make it more suitable for machine learning models.

- Exploratory Data Analysis (EDA)

EDA is crucial to understand the underlying patterns in the data, identify key features, and detect relationships between variables. Here's a summary of the insights gained during the analysis:

- Churn Distribution:

Approximately 26.5% of customers churned, while 73.5% stayed, indicating a class imbalance.

- Univariate Analysis:

Gender: The dataset shows a near-equal distribution between male and female customers.

Senior_Citizen: A small proportion of senior citizens, as expected.

Is_Married: A balanced distribution between married and unmarried customers.

Dependents: Most customers do not have dependents.

Internet_Service: A growing trend in customers opting for fiber optic internet service.

Payment_Method: Electronic check is the most common payment method.

Paperless_Billing: Most customers have opted for paperless billing.

- Bivariate Analysis:

Churn and Tenure: Customers with shorter tenure are more likely to churn.

Churn and Monthly_Charges: Higher monthly charges slightly correlate with a lower churn rate.

Churn and Contract: Month-to-month contracts have a significantly higher churn rate compared to one-year or two-year contracts.

- Feature Correlations:

Senior_Citizen and Is_Married have a strong influence on churn behavior, with married customers being more likely to churn.

Internet_Service (Fiber optic) customers tend to churn more, likely due to higher expectations.

- Data Splitting

After preparing the dataset, it was split into training, validation, and test sets for model evaluation. The data was divided as follows:

- Training Set: 70% of the data used to train models.
- Validation Set: 15% of the data used to tune the model.
- Test Set: 15% of the data used to evaluate the final model.

The test set was reserved for final evaluation, ensuring that the performance metrics reflect how the model would perform on unseen data.

- Model Development and Evaluation

Multiple models were tested for their ability to predict churn. Below is a list of the models and techniques used:

- Logistic Regression: Used as a baseline model for comparison. Grid search was applied to optimize hyperparameters.
- Decision Tree: A decision tree model was trained with grid search for hyperparameter optimization.
- Random Forest: An ensemble method was used to improve the prediction power by averaging multiple decision trees.
- Gradient Boosting: A boosting algorithm was applied to improve performance by sequentially adding trees to correct errors from previous models.
- XGBoost: An optimized version of gradient boosting, known for its performance and speed, was used to predict churn.
- Bootstrap Sampling with Balancing: SMOTE (Synthetic Minority Over-sampling Technique) was used to oversample the minority class (churn), and UnderSampling was applied to the majority class (non-churn). Stacking was used to combine the predictions of multiple models (Random Forest, SVM, and XGBoost) with Logistic Regression as a meta-learner.

- Evaluation Metrics

Evaluation of model performance was based on several metrics to ensure balanced performance across both churn and non-churn classes:

- Precision: Measures the accuracy of churn predictions (True Positives / (True Positives + False Positives)).
- Recall: Measures the ability of the model to identify actual churn customers (True Positives / (True Positives + False Negatives)).
- Macro F1-Score: The harmonic mean of precision and recall, giving a balanced measure of performance across both classes.

- Model Results and Observations

The following summarizes the performance of the models:

- Logistic Regression: High recall for churn (0.77) but low precision (0.54), indicating that while it correctly identifies churned customers, it also produces many false positives.

- Decision Tree: Slightly higher recall (0.78) but still suffers from low precision (0.51).
- Random Forest: Better precision (0.59) but lower recall (0.62) compared to other models.
- Gradient Boosting: Struggled with churn detection, with a recall of only 0.50, making it less effective for this task.
- XGBoost: Balanced performance with a recall of 0.63 and a macro F1-score of 0.75, showing it as a strong contender.
- Stacking with Bootstrap Sampling: Although recall was slightly lower (0.55), stacking combined the strengths of different models, but overall performance was moderate.

- Best Model Selection
  - Best Model for Recall: Logistic Regression provided the highest recall for churn detection, crucial for identifying at-risk customers.
  - Best Model for Macro F1-Score: XGBoost, with a macro F1-score of 0.75, was selected as the final model for deployment.

  The XGBoost model provides the best balance between precision, recall, and F1-score, making it the ideal choice for predicting customer churn in this case. The marketing team can now use this model to identify customers at risk of churning and take preventive actions to retain them.

In the competitive landscape of subscription-based services, understanding why customers leave—commonly referred to as "churn"—is crucial for maintaining profitability and improving service offerings. So we are moving now to development and functionality of a Streamlit-based application designed to assist the marketing team in classifying customer churn. The application leverages a machine learning model to predict churn and provides explanations for these predictions through interactive chat.

- Technical Framework

  The application is built using several advanced technologies and libraries:

  - Streamlit: Used for creating the web-based interface that facilitates user interaction with the model in a conversational manner.
  - Groq API: Integrates a large language model to generate natural language explanations based on the churn prediction results.
  - Joblib: Loads the pre-trained churn prediction model for real-time predictions.
  - LangChain and ChatGroq: These libraries are employed to manage the conversational flow, ensuring that the chatbot effectively gathers all necessary customer details for churn prediction.

- Application Overview

  The application functions as a marketing assistant, guiding users through a series of questions to collect detailed customer data. This data is then processed and fed into a churn prediction model. The predicted results are explained using AI-generated responses, providing insights into the factors influencing each prediction.

- Core Functionality

  1. **Interactive Questionnaire**: The application presents a series of questions designed to gather comprehensive customer data. This includes demographics, service usage, and billing information.
  2. **Data Preprocessing**: Inputs collected from the user are encoded and formatted to match the requirements of the churn prediction model.
  3. **Churn Prediction**: Utilizes a pre-trained machine learning model to determine the likelihood of a customer discontinuing service.
  4. **Explanation Generation**: After predicting churn, the app queries the Groq API to generate a detailed explanation, which helps the marketing team understand the reasoning behind each prediction.

- Operational Flow

  - **Initial Setup**: When launched, the app loads necessary configurations and initializes variables to manage the flow of conversation and data collection.
  - **Data Collection**: Through an interactive chat interface, the app asks the user predefined questions to collect data about a customer.
  - **Processing and Prediction**: Collected data is preprocessed and used to predict churn using a stored machine learning model.
  - **Explanation Delivery**: An AI-generated explanation of the prediction is provided to give context and insights into the factors influencing the churn decision.

  You can use the chatbot from here