Telecommunications Customer Churn Prediction Project Documentation

Project Aim:

- This project aims to predict customer churn in a telecommunications company based on various customer attributes and service usage patterns. We'll utilize data analysis and machine learning techniques to achieve this.

Tools Used:

- Python for data analysis, EDA, feature engineering, and machine learning.

Libraries:

- pandas: Data manipulation and analysis
- matplotlib/seaborn: Data visualization
- scikit-learn: Machine learning algorithms

Project Benefits:

- Identify customers at risk of churning.
- Develop proactive retention strategies.
- Improve customer satisfaction and loyalty.

Initial Hypotheses:

- **Demographic Factors:** Churn behavior might differ based on age, partner/dependent status.
- **Service Usage:** Longer tenure, subscribed services (security, tech support) might influence churn.
- **Contract & Billing:** Longer contracts or paperless billing might reduce churn.
- **Payment Method:** Specific payment methods might correlate with churn rates.

Data Columns:

- CustomerID: Unique customer identifier
- Demographics: Gender, SeniorCitizen, Partner, Dependents
- Service Usage: Tenure, PhoneService, MultipleLines, InternetService, ... (Other services)
- Contract & Billing: Contract, PaperlessBilling
- Payment Method: Electronic check, Mailed check, Bank transfer, Credit card
- Charges: MonthlyCharges, TotalCharges
- Churn: Whether the customer churned (Yes/No)

Steps in project:

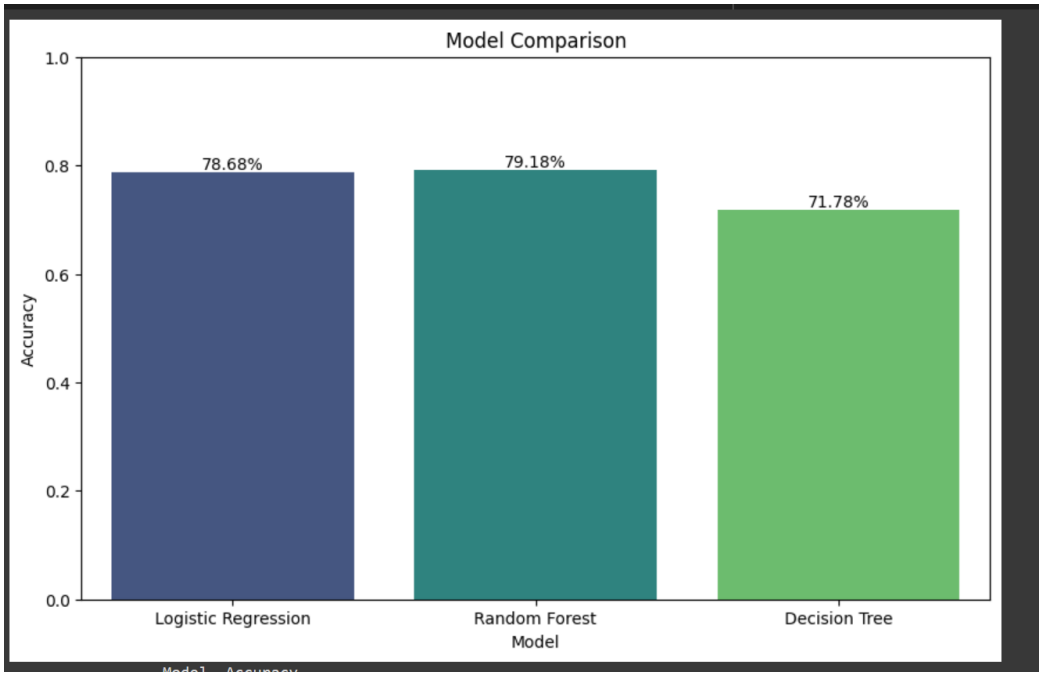
1. Importing Libraries and Loading Data

Dataset link: https://www.kaggle.com/datasets/blastchar/telco-customer-churn

2. Exploratory Data Analysis (EDA)

- **Data Overview:** Get basic information about data (number of rows, columns, data types).
- **Missing Values:** Check for missing values and handle them appropriately (e.g., imputation or deletion).
- **Descriptive Statistics:** Analyze numerical features (mean, median, standard deviation) and categorical features (counts, frequencies).
- **Visualizations (using seaborn):**
    - Distribution plots (histograms, density plots) to understand the spread of numerical features.
    - Count plots and bar charts to visualize categorical features.
    - Boxplots to compare distributions across different categories.
    - Scatter plots to explore relationships between features (e.g., tenure vs. churn).

Overall model comparision:



3. Data Preprocessing:

- **Feature Engineering:** Create new features if needed (e.g., combining categories).
- **Encoding Categorical Features:** Convert categorical features into numerical representations suitable for machine learning algorithms (e.g., one-hot encoding).
- **Feature Selection:** Select the most relevant features that might influence churn prediction.

4. Feature Importance:

- Use feature importance techniques (e.g., from scikit-learn) to identify features with the most significant impact on churn prediction.

5. Splitting Data:

- Divide the data into training and testing sets. The training set is used to train the machine learning models, and the testing set is used to evaluate their performance on unseen data.

6. Feature Scaling:

- Scale numerical features (if necessary) to ensure all features are on a similar scale and contribute equally to the model.

7. Machine Learning Models :

- **Logistic Regression:** A statistical model for binary classification problems (churn/no churn).
    - Train the model on the training data.
    - Evaluate the model's performance on the testing data using accuracy score, confusion matrix, and classification report.
- **Decision Tree:** A tree-based model that learns decision rules to classify data points.
    - Train the model and evaluate its performance.
- **Random Forest:** An ensemble method that combines multiple decision trees for improved accuracy and robustness.
    - Train the model and evaluate its performance.

8. Model Comparison :

- Compare the performance of different models using metrics like accuracy score.
- Visualize the comparison using a bar chart (using matplotlib/seaborn).