

# Predicting Customer Churn and Retention Strategies README

## # Churn Prediction Project

This project involves data preprocessing, exploratory data analysis (EDA), and machine learning models (Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting) for predicting customer churn in a telecom dataset. It uses Python and several data science libraries.

## ## Requirements

Before running the project, ensure that the following dependencies are installed:

### 1. **\*\*Install dependencies\*\***:

Use the following commands to install the necessary Python libraries:

```
``bash
pip install kagglehub
pip install kaggle
pip install pandas
pip install numpy
pip install seaborn
pip install matplotlib
pip install plotly
pip install scikit-learn
``
```

### 2. **\*\*Kaggle API Credentials\*\***:

To access the Kaggle dataset, you need to set up Kaggle API credentials. Ensure that you have your `KAGGLE\_USERNAME` and `KAGGLE\_KEY` as environment variables:

```
``bash
export KAGGLE_USERNAME="your_kaggle_username"
export KAGGLE_KEY="your_kaggle_key"
``
```

## ## Project Structure

- **\*\*final\_complete\_code.py\*\***: The main Python script containing all steps: data cleaning, EDA, modeling, and evaluation.

- **cleaned\_stage1.csv**: Intermediate data after initial cleaning.
- **cleaned\_stage2.csv**: Final cleaned dataset after handling outliers and missing values.
- **monthly\_charges\_summary\_3\_months.csv**: Summary statistics for monthly charges, churn, and customer counts in 3-month intervals.
- **monthly\_charges\_summary\_6\_months.csv**: Similar to the above but for 6-month intervals.
- **tenure\_bin\_counts\_monthly\_charges\_summary.csv**: Contains customer counts, churn percentage, and monthly charges by tenure bin.

### ## How to Run the Code

#### 1. Clone the Repository:

Clone this repository to your local machine:

```
``bash
git clone https://github.com/your_username/churn-prediction.git
cd churn-prediction
``
```

#### 2. Dataset Download:

The dataset used in this project is `WA_Fn-UseC_-Telco-Customer-Churn.csv` from Kaggle. Ensure you have Kaggle API credentials set up (as mentioned above) to download the dataset by running:

```
``bash
!kaggle datasets download -d blastchar/telco-customer-churn -p ./ --unzip
``
```

#### 3. Run the Script:

Once the dataset is downloaded, you can run the main script (`final_complete_code.py`):

```
``bash
python final_complete_code.py
``
```

#### 4. Results:

- The script performs data cleaning, visualizes key trends, and evaluates various machine learning models.
- The final outputs include models' performance metrics and various visualizations (such as feature importance and churn rates).

## ## Key Sections of the Code

### ### 1. \*\*Data Cleaning\*\*:

- Duplicate and missing data are handled.
- Columns with categorical values are encoded.
- Outliers are detected and removed based on the IQR method.

### ### 2. \*\*Exploratory Data Analysis (EDA)\*\*:

- Plots like histograms, bar charts, and box plots are used to explore features like `MonthlyCharges`, `Tenure`, and `Churn`.

### ### 3. \*\*Modeling\*\*:

- Models such as Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting are trained.
- Hyperparameter tuning is performed using GridSearchCV.
- The models are evaluated based on accuracy, precision, recall, F1-score, and AUC.

### ### 4. \*\*Results\*\*:

- Feature importance is analyzed to understand the key drivers of customer churn.
- The best-performing models are evaluated using precision-recall curves and ROC curves.

## ## Notes

- **Customization**: If you want to use a different dataset, make sure to adjust the column names and any preprocessing steps accordingly.
- **Further Improvements**: You can experiment with other models, such as Support Vector Machines (SVM) or XGBoost, for potentially better performance.

## ## Conclusion

This project demonstrates a comprehensive workflow for predicting customer churn, from data cleaning and visualization to model development and evaluation. You can further extend it by adding new features or fine-tuning the models.