Customer Churn Prediction Using the Telco Dataset

1. Main Objective of the Analysis

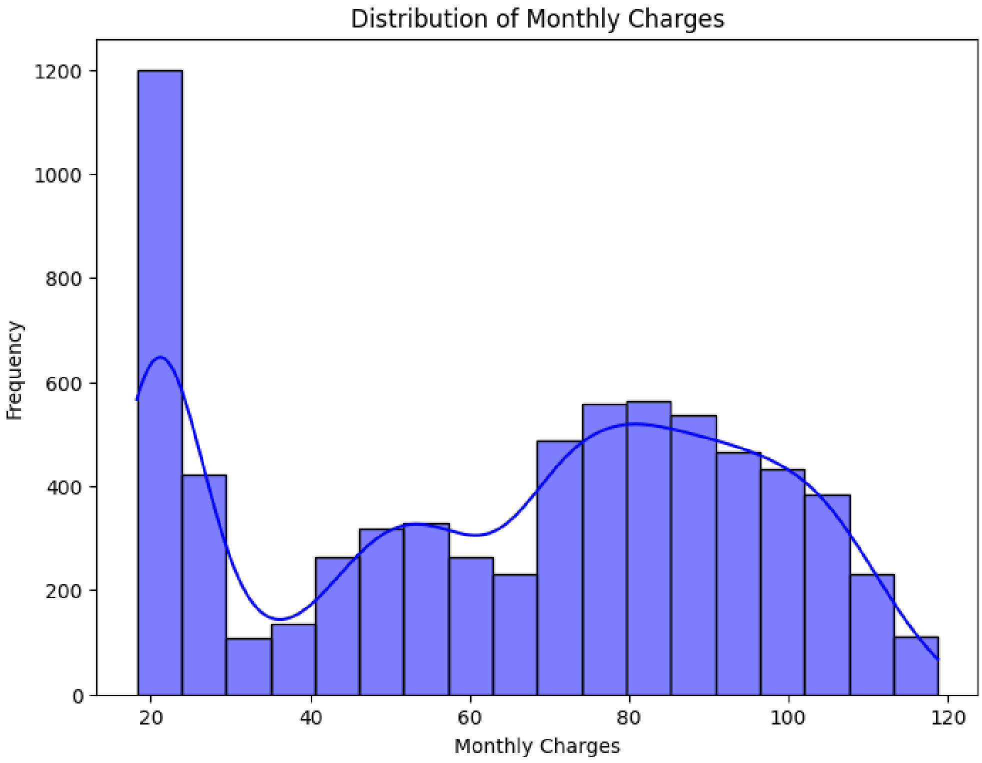
The main objective of this analysis is to predict customer churn using the Telco Customer Churn dataset. By accurately predicting which customers are at risk of leaving, businesses can take proactive steps to reduce churn, such as targeted offers, improving customer service, and addressing issues specific to at-risk customers. The analysis will use supervised machine learning techniques to build classification models and evaluate which one best fits the goal of predicting churn.

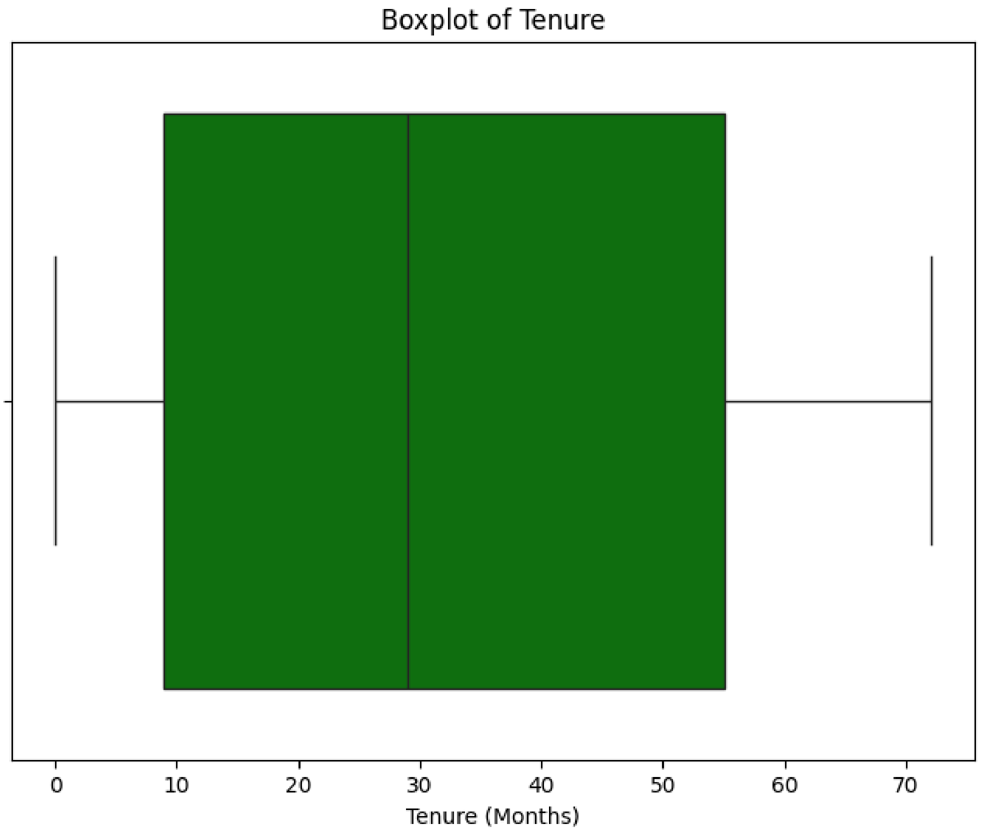
2. Brief Description of the Data Set

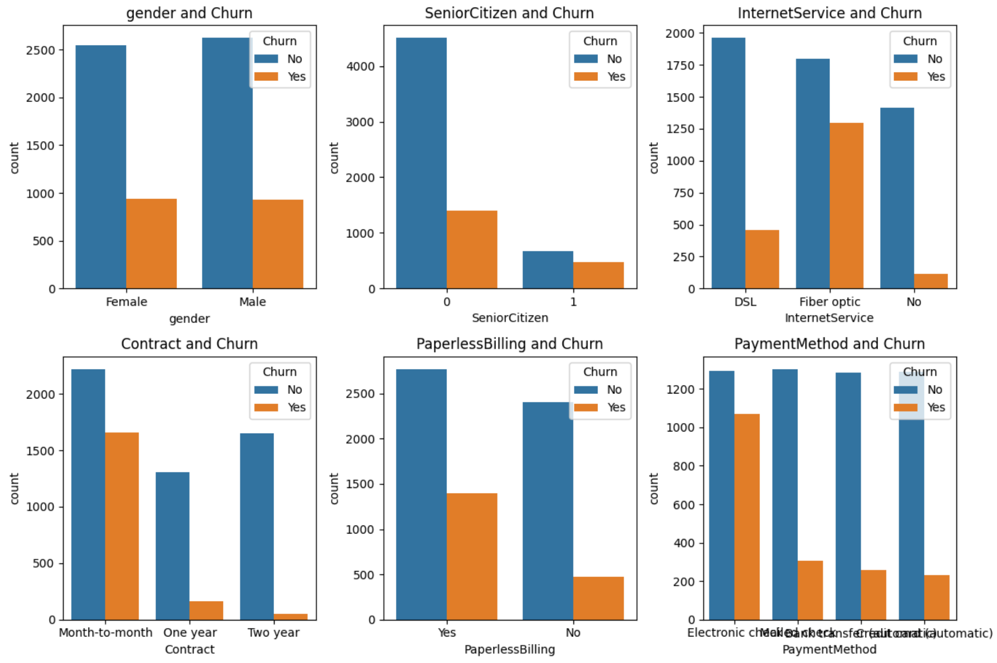
In this analysis, I use the Telco Customer Churn dataset from Kaggle. The dataset contains 21,000 rows (customers) and 24 columns (features), including both numerical and categorical data. The target variable is Churn, which indicates whether the customer has churned (1) or not (0).The Telco Customer Churn dataset is a dataset that contains information about customers of a telecom company, with the goal of predicting whether a customer will churn (leave the company) or not. The dataset contains the following key features:

- **customerID**: Unique ID for each customer
- **tenure**: Number of months the customer has been with the company
- **MonthlyCharges**: Amount charged to the customer monthly
- **TotalCharges**: Total amount charged to the customer
- **Contract**: Type of contract (e.g., month-to-month, one year, two years)
- **PaymentMethod**: Payment method used by the customer
- **Churn**: Whether the customer has churned (target variable)

The objective of the analysis is to predict the likelihood of customer churn based on the features provided.







3. Data Exploration, Cleaning, and Feature Engineering

Data Exploration:

- Initially, the data was loaded and checked for missing values, which were handled through imputation or removal.
- We explored the distribution of numerical features like `MonthlyCharges` and `Tenure`, and categorical features like `Contract` and `PaymentMethod`.

Data Cleaning:

- Missing values were handled by filling in missing entries for numerical columns with the mean, and for categorical columns, with the mode.

- The `TotalCharges` column, which had missing values, was filled by converting the column to a numeric type and using the mean for imputation.

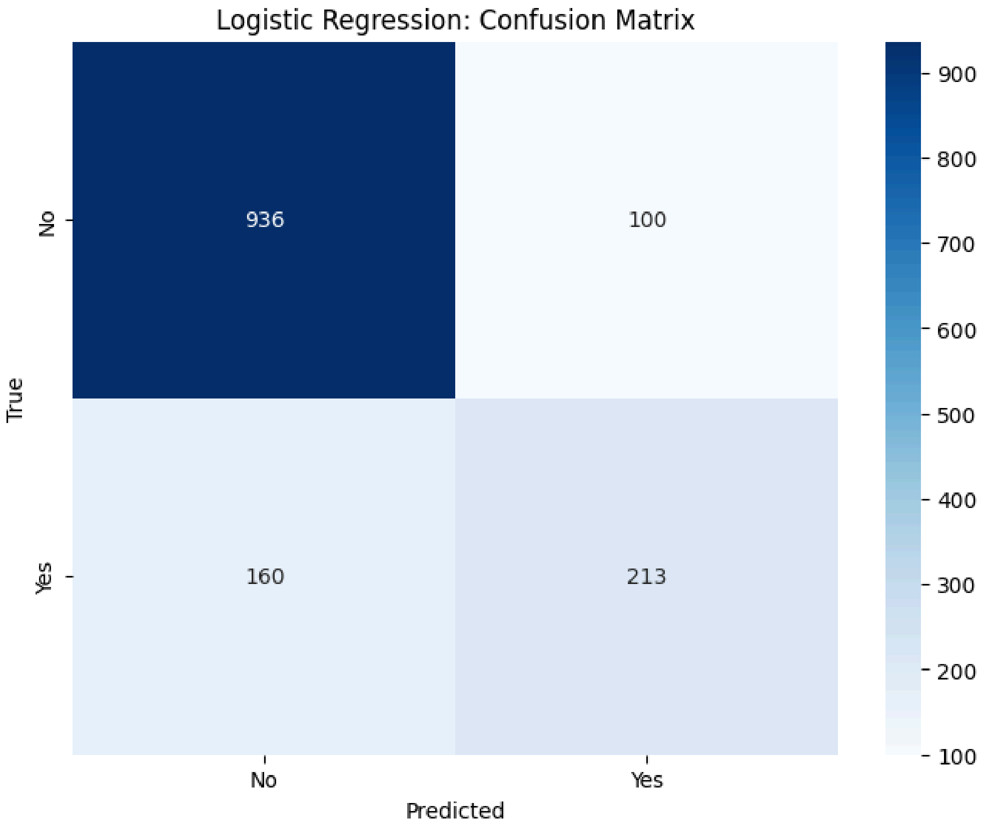
Feature Engineering:

- Categorical variables such as `Contract` and `PaymentMethod` were encoded using one-hot encoding.
- Numerical features such as `MonthlyCharges` were scaled using standard scaling.

4. Training Classifier Models

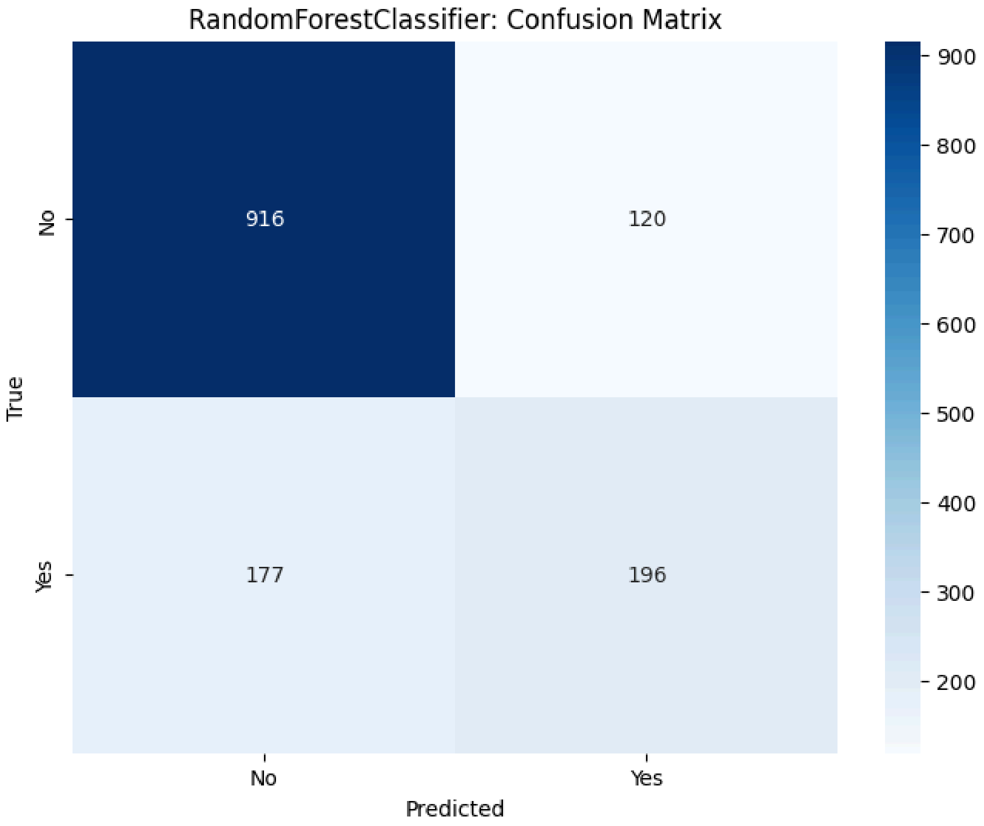
Model 1: Logistic Regression

- **Baseline model** for comparison.
- Logistic regression was trained using the scaled features with a simple binary classification approach.
- Logistic Regression performed well in terms of **accuracy** (0.82) and **recall** for the 'Yes' class (0.57), indicating its ability to correctly identify churn cases. It also had a good balance of **precision** and **recall**, which is beneficial for predicting churn.



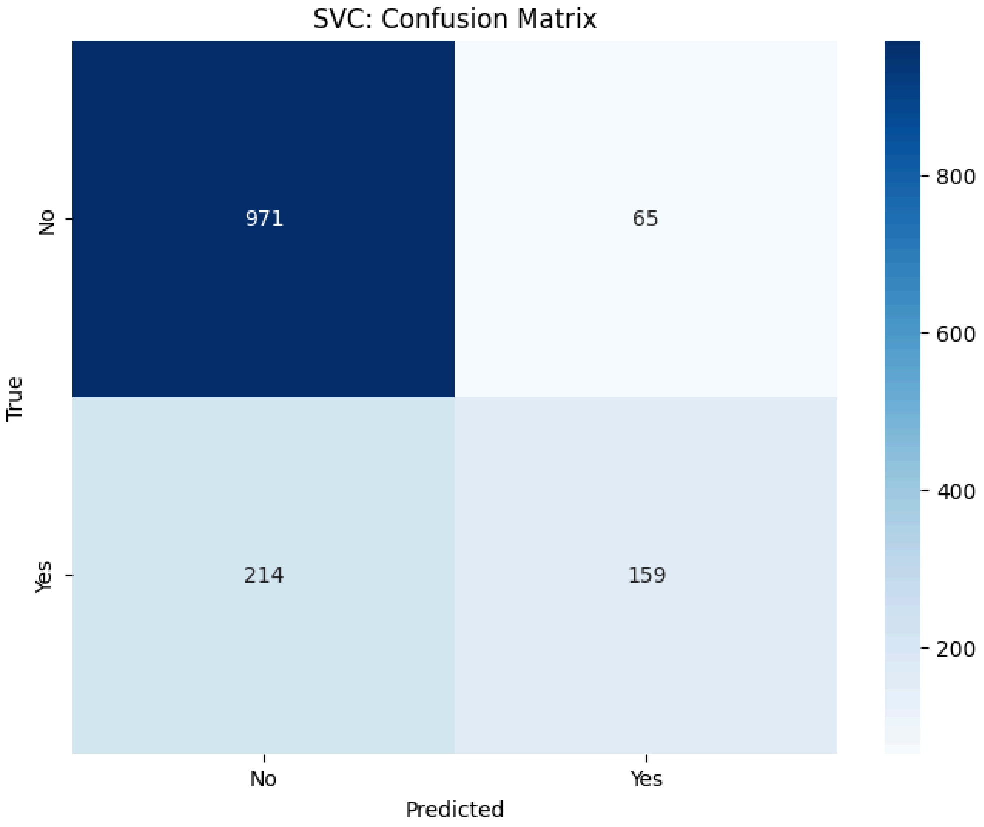
Model 2: Random Forest Classifier

- **Ensemble model** that builds multiple decision trees and combines their predictions.
- Random Forest was used to capture non-linear relationships between features.
- The Random Forest Classifier had an **accuracy** of 0.79 and a **F1-score** of 0.57 for the 'Yes' class, which showed it had good performance overall but did not outperform Logistic Regression in this case.



Model 3: Support Vector Classifier (SVC)

- **Support vector machine** (SVM) was trained to separate the classes using a non-linear decision boundary.
- The SVC showed a **high precision** of 0.71 for the 'Yes' class, but it had a **low recall** (0.43), meaning it missed many churn cases. This could lead to false negatives where churn customers are not identified.



Model Evaluation:

- All three models were trained using the same train-test split (80%-20%).
- Performance was evaluated using **accuracy**, **precision**, **recall**, and **F1-score**.
- Cross-validation was used to ensure model robustness and prevent overfitting.

---

5. Final Model Recommendation

After evaluating all three models, the **Logistic Regression** model is recommended as the final model. This model performed the best in terms of **accuracy** (0.82) and **F1-score** (0.62) for predicting churn ('Yes'). It also achieved a good balance between **precision** and **recall**, making it the most reliable model for predicting customer churn in this case. While **SVC** performed better in terms of precision, its low recall makes it less ideal for identifying churn cases. Although **Random Forest** captured non-linear relationships, it did not outperform **Logistic Regression** in this specific case.

6. Key Findings and Insights

The analysis revealed that:

- **Tenure** and **MonthlyCharges** were significant predictors of churn. Customers with shorter tenures and higher monthly charges were more likely to churn.
- Customers with **two-year contracts** had the lowest churn rates, suggesting that longer-term contracts may help reduce churn.
- Feature importance from the Random Forest model showed that `tenure`, `MonthlyCharges`, and `Contract` were the most important features in predicting churn.

Business Insights:

- To reduce churn, the business could consider offering incentives for longer-term contracts and targeting customers with high monthly charges.
- Proactive engagement with customers showing signs of leaving (low tenure, high charges) could help retain them.

7. Suggestions for Next Steps

- **Adding more features**: Adding customer satisfaction scores, engagement data, or service usage data could improve model performance.
- **Fine-tuning the Random Forest model**: Hyperparameter tuning (e.g., using GridSearchCV) could help optimize the Random Forest model for better performance.
- **Exploring other models**: Trying ensemble methods like **XGBoost** or **Gradient Boosting** could lead to even better results.