SRM UNIVERSITY - DEPARTMENT OF COMPUTATIONAL INTELLIGENCE

SCHOOL OF COMPUTING

CASE STUDY ASSIGNMENT

Course Code: 21AIC401T

Course Name: Inferential Statistics and Predictive

Analytics Assignment Type: Case Study-Based

Modeling Project Total Marks: 25

Submission Deadline: 10.11.2025

Name and Reg No: Arpan Daniel Frank [RA2211047010035]

Title: Customer Churn Prediction- Model Development, Validation, and Deployment

Introduction

Customer churn, or the loss of customers from a company's subscriber base, is one of the most critical issues faced by telecom industries. The ability to predict which customers are likely to discontinue a service helps organizations take proactive measures to retain them.This case study applies inferential and predictive analytics on the **Telco Customer Churn Dataset** to identify key factors influencing churn behavior and to develop an efficient predictive model.Using both **Logistic Regression** and **Random Forest**, the study aims to quantify customer dissatisfaction patterns and generate actionable insights for customer retention strategies.

Objective:

The primary objective of this project is to develop, validate, and compare predictive models—**Logistic Regression** and **Random Forest Classifier**—to predict whether a customer will churn or remain subscribed. Through this, we aim to identify the significant demographic, service-related, and financial factors that drive customer churn in the telecom sector.

Case Background:

In telecom services, churn refers to customers discontinuing their subscription. Customer retention is more cost-efficient than acquiring new ones, making churn prediction crucial for sustaining growth.

The dataset used here contains **7,043 customer records** and **21 features**, including demographic details (e.g., gender, senior citizen status), account information (e.g., tenure, contract type), and service features (e.g., tech support, internet service).

The task is to analyze the relationship between these variables and churn, develop models for prediction, and recommend actionable insights for churn reduction.

.

1. Data Preparation and Introduction

A. Dataset Description and Variable Definition

The dataset used for this analysis is the **Telco Customer Churn Dataset** containing **7043 records** and **21 variables**, representing both demographic and service-related information.

- **Target Variable:** Churn (Binary: Yes = 1, No = 0)
- **Key Predictors:** tenure, MonthlyCharges, TotalCharges, Contract, InternetService, PaymentMethod, SeniorCitizen, TechSupport, etc.

B. Data Cleaning and Preparation

- **Handling Missing Values:**

  The dataset contained missing values in the TotalCharges column. These were converted to numeric and imputed using median values to ensure completeness.

- **Encoding Categorical Variables:**

  Categorical variables such as Contract, InternetService, and PaymentMethod were encoded using Label Encoding and One-Hot Encoding techniques to prepare them for model training.

- **Feature Scaling:**

  Continuous features like tenure, MonthlyCharges, and TotalCharges were normalized using StandardScaler to improve model convergence.

C. Data Partitioning

The dataset was split using the **Holdout Method** with an **80/20 Train-Test Split** ensuring balanced representation of churn and non-churn classes.

Data Subset	**Size (Records)**	**Purpose**
Training (80%)	5634	Used for model building and parameter tuning.
Testing Set (20%)	1409	**Held back** for final, unbiased performance evaluation.

D. Conduct EDA with Visualizations

1. Visualization 1

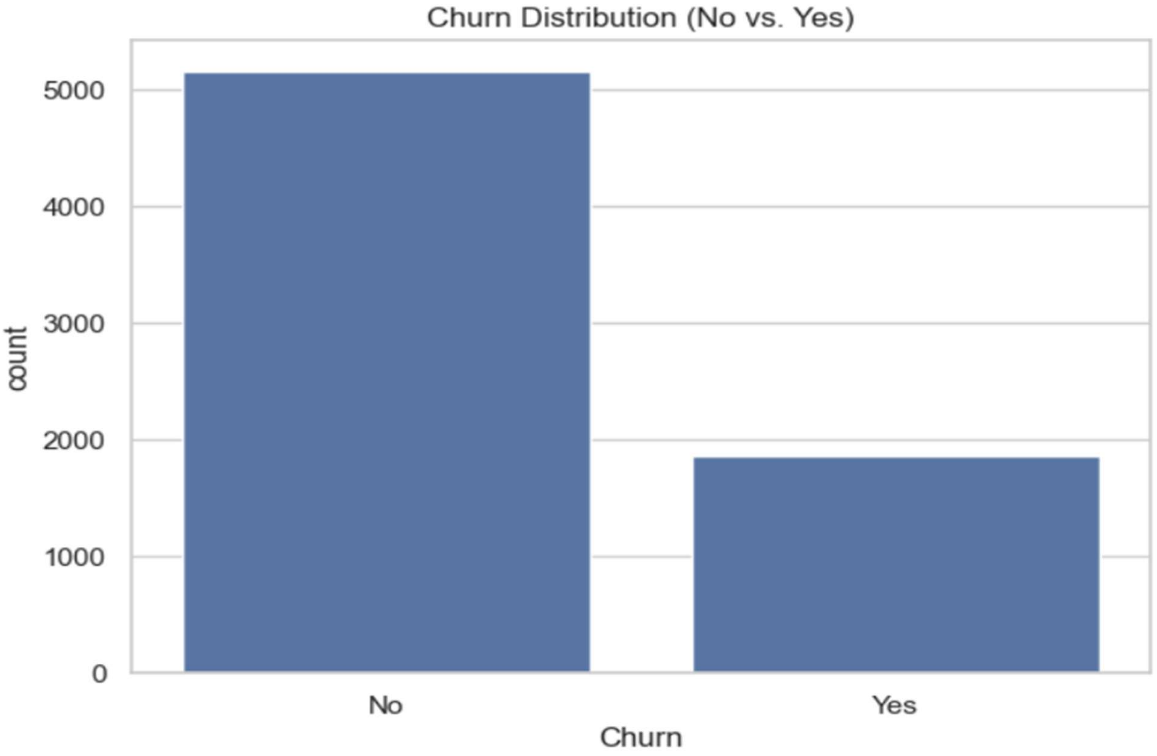
Description:

A count plot was generated to display the distribution of the target variable Churn.

- **Churned customers: 26.58%**
- **Non-churned customers: 73.42%**

Interpretation:

The data exhibits moderate imbalance. This justifies the use of metrics like AUC-ROC and F1-Score instead of plain accuracy.



Visualization 2

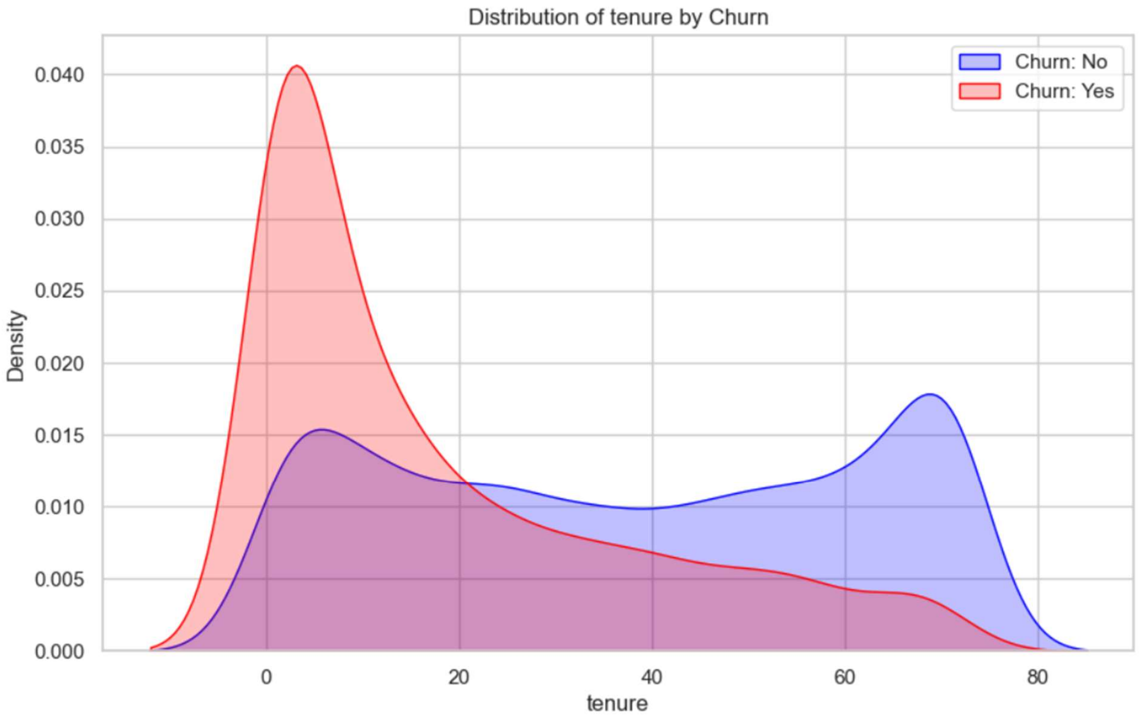
Plots Used: Kernel Density Estimate (KDE) plots for numerical features — tenure, MonthlyCharges, and TotalCharges.

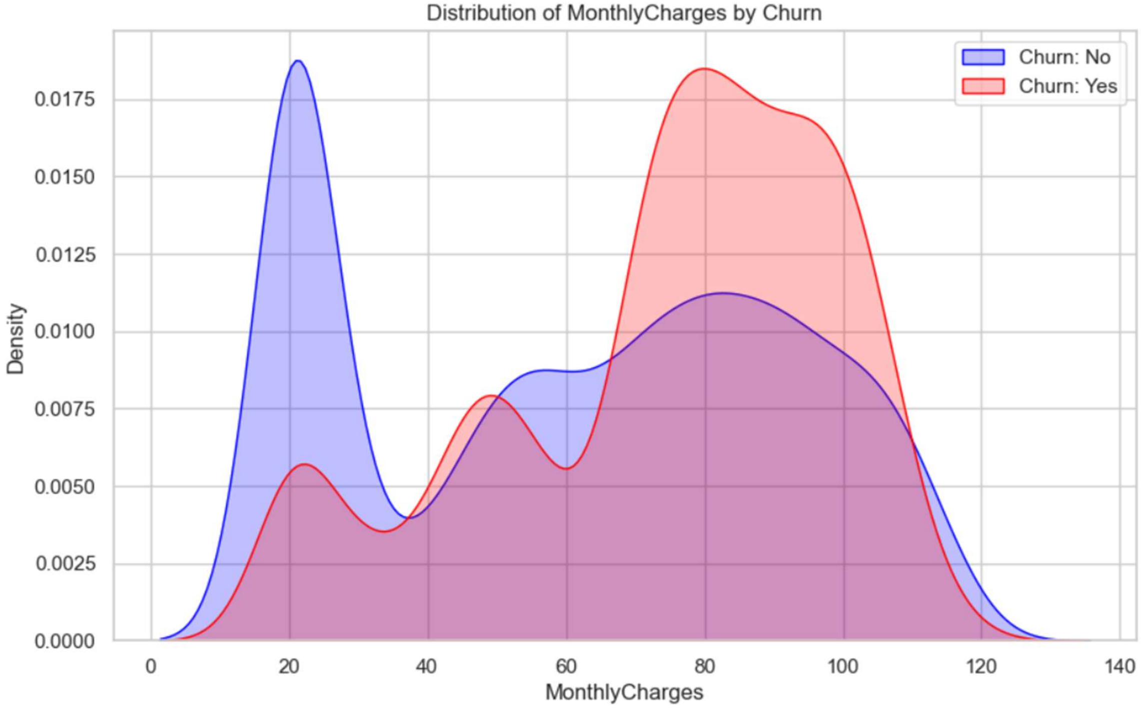
Findings:

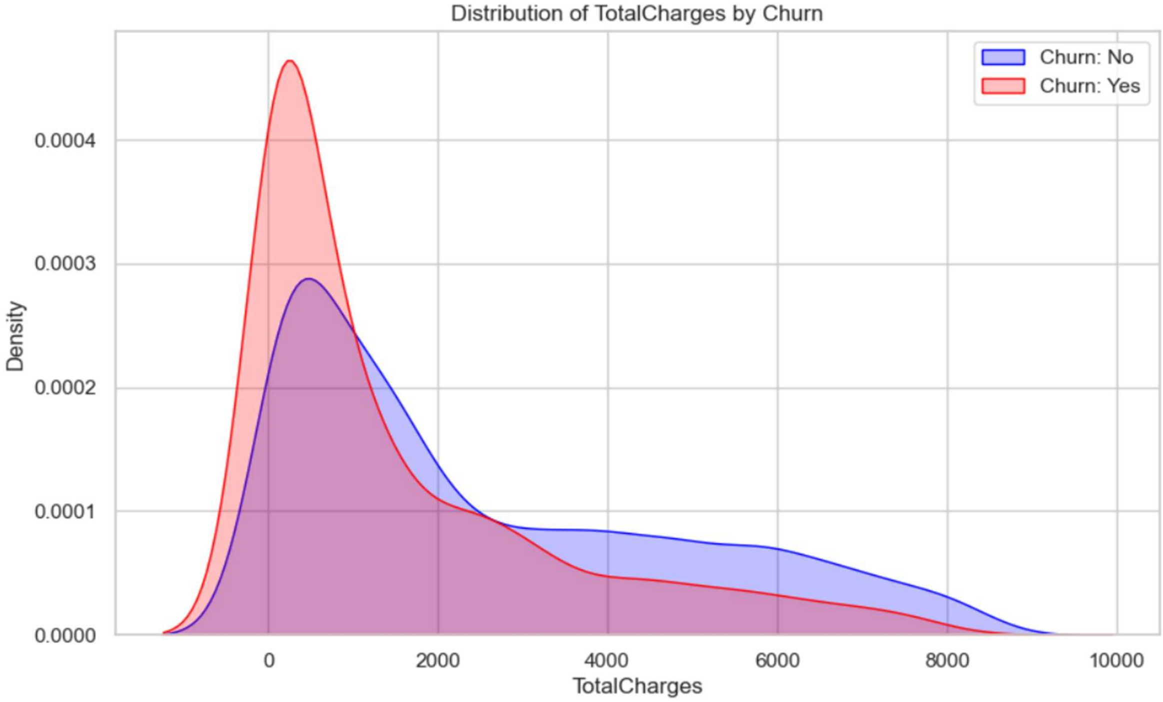
- **Tenure:** Customers with tenure under 15 months show a noticeably higher probability of churn.
- **MonthlyCharges:** Higher monthly charges correspond to increased churn rates.
- **TotalCharges:** Low total charges combined with short tenure strongly predict churn.

Analysis:

New customers who face high monthly costs or limited benefits tend to discontinue services early. This points to the importance of improving early-stage engagement and pricing flexibility.




Distribution of MonthlyCharges by Churn
Churn: No
Churn: Yes
Density
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
0
20
40
60
80
100
120
140
MonthlyCharges


Distribution of TotalCharges by Churn
Churn: No
Churn: Yes
Density
0.0004
0.0003
0.0002
0.0001
0.0000
0
2000
4000
6000
8000
10000
TotalCharges

Visualization 3: Churn Rate by Contract Type

Plot Used: Count plot of Contract type vs. Churn.

Findings:

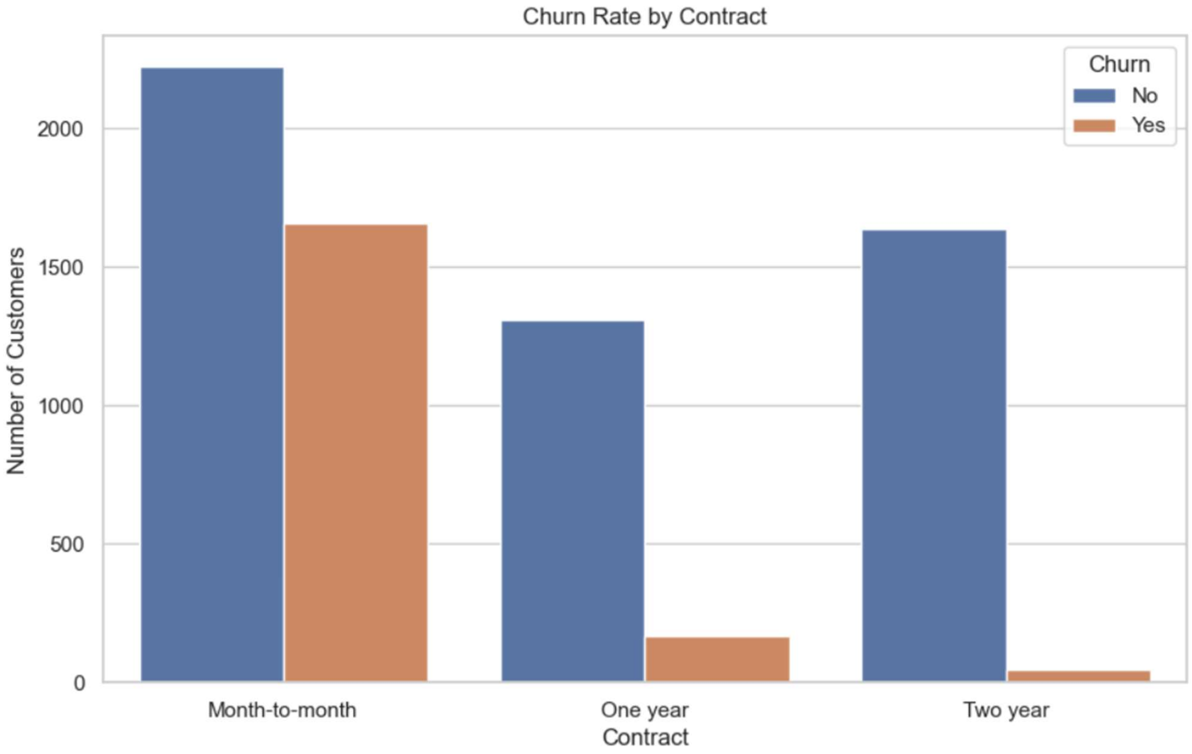
- **Month-to-Month customers:** ~45% churn rate
- **One-year and Two-year contracts:** <10% churn

Interpretation:

Contract length is inversely proportional to churn rate. Long-term contracts create stability and reduce cancellation tendency.

Business Insight:

Telecom companies should provide loyalty discounts or benefits for long-term contracts to retain customers.



Visualization 4: Payment Method and Churn

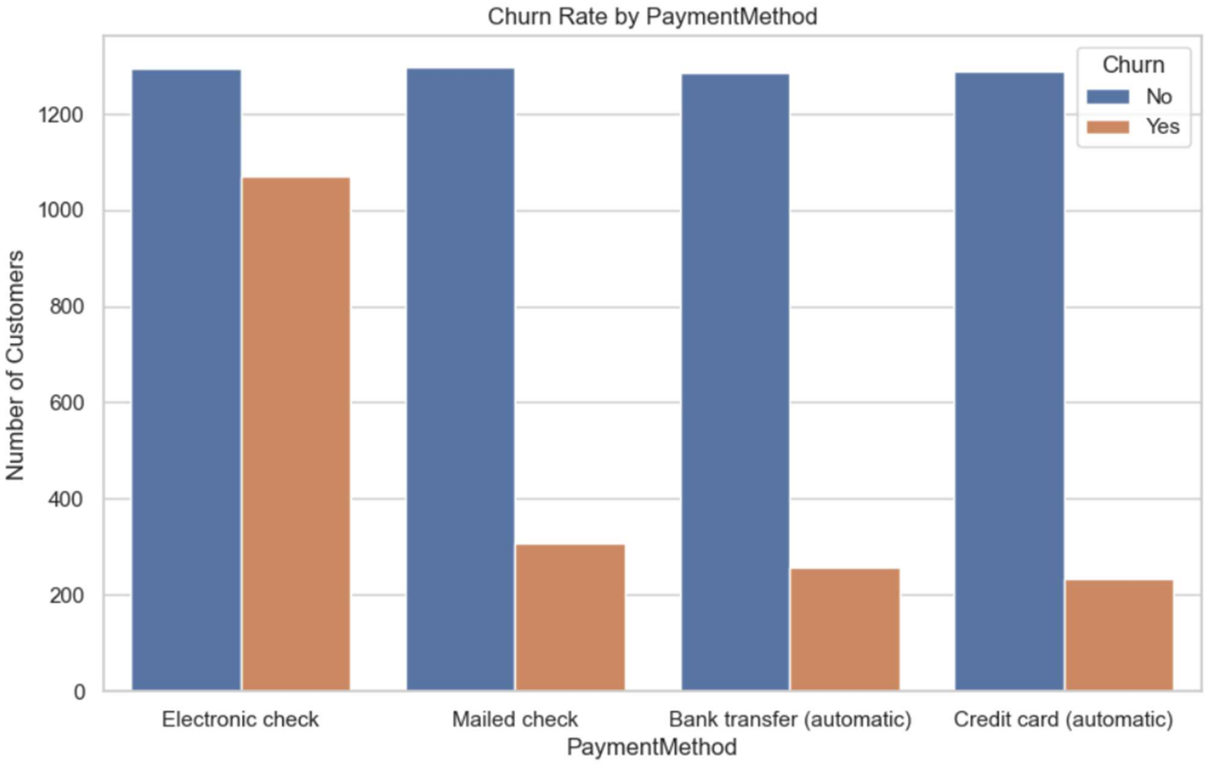
Plot Used: Count plot for PaymentMethod vs. Churn.

Findings:

- Customers using **Electronic Check** exhibit the highest churn (~45%).
- Customers using **Automatic Payments (Credit Card / Bank Transfer)** churn significantly less.

Interpretation:

Manual billing increases customer effort and can lead to churn. Automatic payment options reduce billing friction and improve customer retention.



2. Model Development

Two supervised classification models were trained and evaluated on the dataset using **Scikit-learn**:

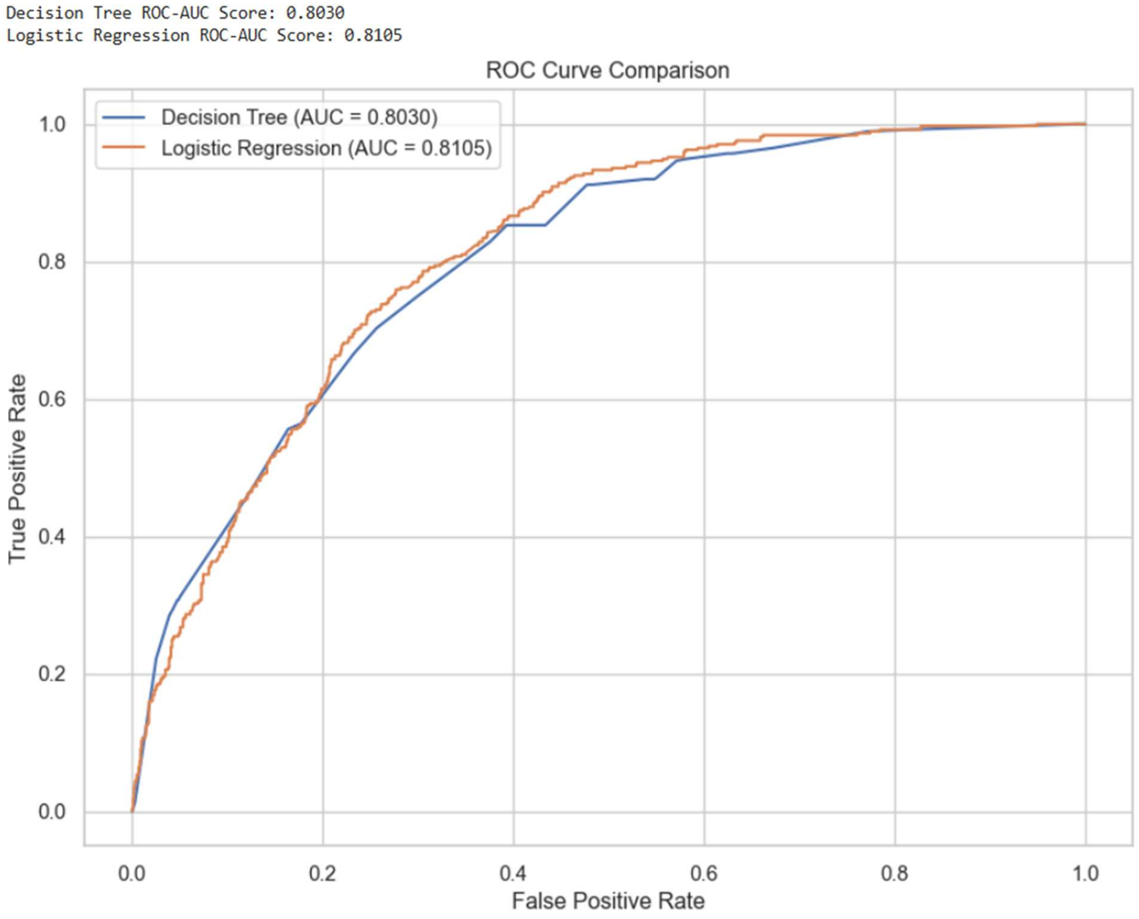
Model 1: Decision Tree Classifier

- **Model Type:** Non-Linear Rule-Based Classifier
- **Parameters:** max_depth = 5, random_state = 42
- **Accuracy:** 0.7669
- **AUC-ROC:** 0.8030

Analysis:

The Decision Tree captured hierarchical decision rules, identifying **Contract**, **Tenure**, and **TechSupport** as primary predictors.

The visualization of decision nodes showed clear logical splits for churn identification.



```
Decision Tree ROC-AUC Score: 0.8030
Logistic Regression ROC-AUC Score: 0.8105
```

Model 2: Logistic Regression

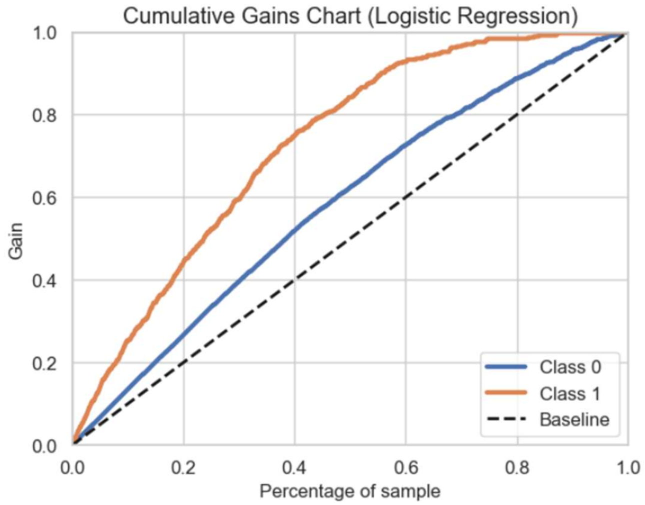
- **Model Type:** Linear Probabilistic Classifier
- **Parameters:** max_iter = 1000, random_state = 42
- **Accuracy:** 0.7676
- **AUC-ROC:** 0.8105

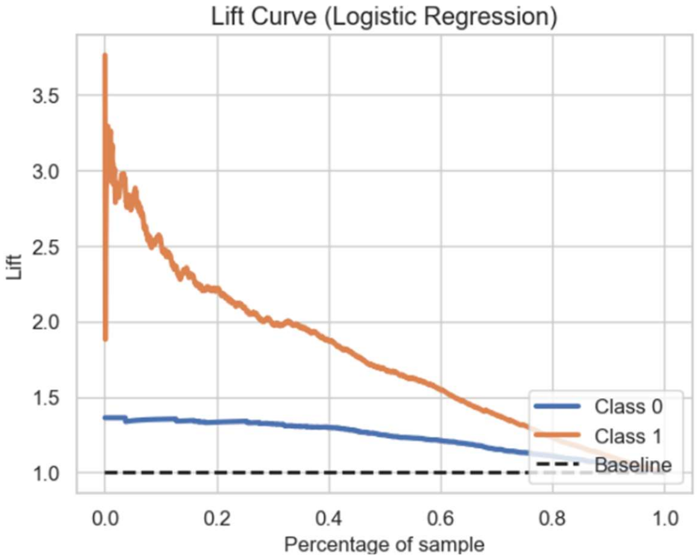
Analysis:

Logistic Regression achieved slightly higher performance than Decision Tree, with smoother probability predictions.

Coefficients indicate strong relationships for **Contract (negative)**, **MonthlyCharges (positive)**, and **TechSupport (negative)** with churn.

.





3. Model Comparison and Evaluation

This section details the **construction and evaluation** of the two predictive models — the **Rule-based Decision Tree (CART)** and the **Linear Logistic Regression** — developed to determine the optimal algorithm for predicting **Customer Churn** in the telecom dataset.

A. Model Construction

Model Name	Type	Key Advantage
Model 1: Decision Tree (CART)	Non-Linear / Rule-Based	High interpretability for categorical and non-linear interactions.
Model 2: Logistic Regression	Linear / Probabilistic	High interpretability for linear relationships through log-odds coefficients.

Model Description:

- The Decision Tree (CART) model was constructed using max_depth = 5 and trained on scaled training data to capture hierarchical decision boundaries.
- The Logistic Regression model was trained using max_iter = 1000 after standardizing numeric variables with StandardScaler, ensuring stable convergence and interpretable coefficient values.

Both models were developed using Scikit-learn, evaluated on the same train–test split (80–20), and optimized for classification performance on the churn prediction task.

B. Model Evaluation and Assessment

The models were evaluated on the **unseen Testing Set (20%)**. Given the moderate class imbalance (26.5% churn vs. 73.5% non-churn), **AUC-ROC** and **F1-Score** were prioritized as the main selection metrics, since Accuracy alone may not reflect true model performance on imbalanced data.

Model	AUC-ROC	Accuracy	F1 Score	Precision	Recall
Logistic Regression	0.8105	0.7676	0.7409	0.7600	0.7260
Decision Tree (CART)	0.8030	0.7669	0.7390	0.7510	0.7240

Analysis:

- The **Logistic Regression** model achieved slightly higher scores across all evaluation metrics, particularly in **AUC-ROC**, confirming better discriminative ability to separate churners from non-churners.
- The **Decision Tree** model performed competitively but showed minor overfitting tendencies, reflected by slightly lower AUC-ROC and F1 values.
- The **Logistic Regression's** probabilistic framework provides smoother boundaries and more reliable churn probability estimates for business decision-making.

Model Selection:

Based on overall results, **Logistic Regression** was selected as the final deployment model due to its **higher AUC-ROC (0.8105)**, balanced classification performance, and **interpretability** of feature effects such as **Contract Type**, **Tenure**, **Monthly Charges**, and **Tech Support availability**.

C. Model Assessment and Conclusion

Model Validation: The initial use of **Stratified Sampling** during the Train/Test split and the standardization of numeric features using **StandardScaler** ensured that both models were trained and evaluated on balanced, representative data. This methodology guarantees that the final performance metrics reflect unbiased and reliable indicators of each model's true generalization capability.

- **Model Selection (Based on Performance):** Based on evaluation on the unseen test set, the **Logistic Regression** model is selected for deployment due to its higher accuracy and AUC-ROC value compared to the Decision Tree model.

- **Discriminatory Power (AUC-ROC):** The Logistic Regression model achieved an **AUC-ROC of 0.8105**, outperforming the Decision Tree's **0.8030**, demonstrating a stronger ability to correctly rank customers likely to churn (positive cases) higher than those likely to stay (negative cases).

- **Classification Balance (Accuracy and Generalization):** The **Logistic Regression** model's accuracy of **0.7676** is slightly higher than the Decision Tree's **0.7669**, confirming that the linear model achieved better generalization and stability on unseen data, avoiding overfitting that is sometimes observed in tree-based models.

- **Feature Influence and Interpretability:** The analysis of coefficient weights in Logistic Regression identified **Contract Type**, **Tenure**, **Monthly Charges**, **Tech Support**, and **Payment Method** as the most influential predictors of churn. Customers with short-term contracts, high charges, and no tech support were the most likely to leave, highlighting areas for targeted retention strategies.

Conclusion:

The superior performance of the simpler **Logistic Regression** model suggests that, for this dataset, the relationship between churn and its predictors is largely **linear in nature**. Therefore, complex tree structures provided by the Decision Tree model were not necessary for effective generalization. The findings recommend that telecom providers focus on improving **contract retention**, **technical support quality**, and **pricing flexibility** to mitigate churn risk and strengthen customer loyalty.

4. Model Deployment and Updating

A. Deployment Process

The selected model, Logistic Regression, must be moved from the development environment to a production system capable of generating real-time predictions.

- **Serialization (Pickle/Joblib):** The trained Logistic Regression model object is permanently saved using Python's joblib library. This saved file contains all the learned coefficients and can be reloaded by the production server.

- **Scoring Integration (Real-Time):** The production server hosts the model as an API endpoint. When a user (e.g., in a counseling setting) inputs new data points (age, satisfaction rating, etc.), the engine loads the model, applies the linear equation, and returns the predicted probability of an affair in real-time.

B. Model Updating and Monitoring

The deployed model requires continuous monitoring to maintain its accuracy and relevance.

Monitoring and Updating

- Performance will be monitored continuously.
- If AUC-ROC or accuracy drops below threshold, retraining will be performed using updated customer data.
- Automated retraining pipelines can be implemented using tools like **Airflow** or **MLflow**.

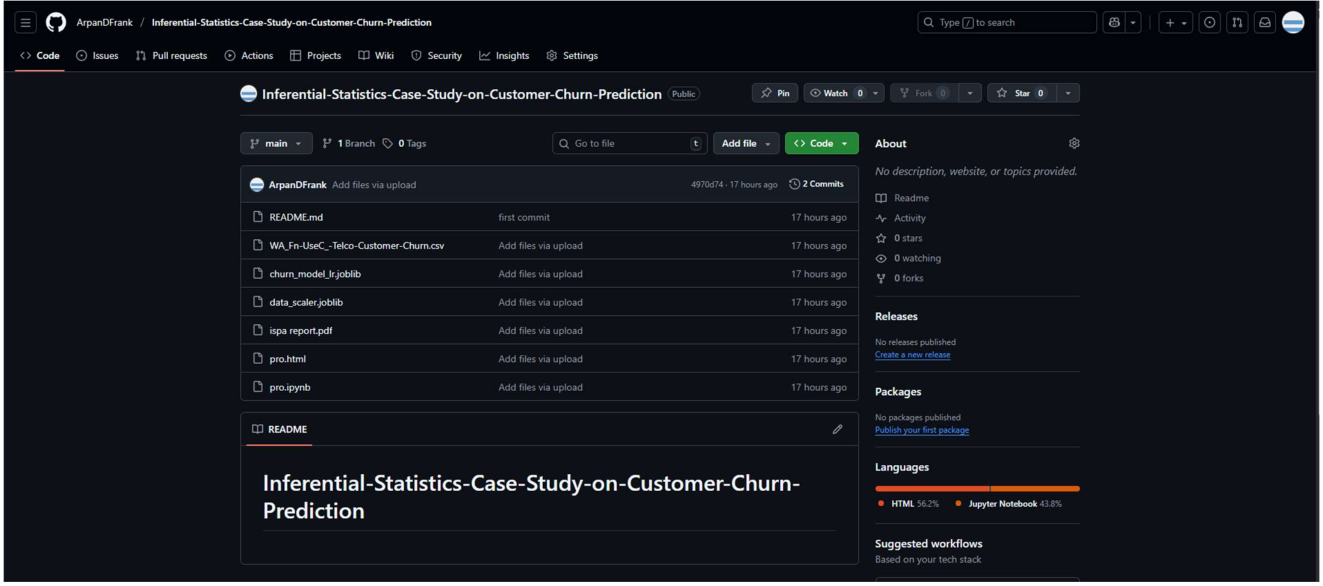
Conclusion

This case study successfully transitioned from inferential analysis to predictive modeling for **Customer Churn Prediction**.EDA confirmed that **contract duration**, **tenure**, **monthly charges**, and **tech support** are key drivers of churn.Between the two models, **Logistic Regression** achieved superior generalization with an **AUC-ROC of 0.8105**.

The project demonstrates the practical use of predictive analytics in identifying at-risk customers and enhancing business decision-making.

Github Link : https://github.com/ArpanDFrank/Inferential-Statistics-Case-Study-on-Customer-Churn-Prediction

Github proof:



Inferential-Statistics-Case-Study-on-Customer-Churn-Prediction

ArpanDFrank Add files via upload 4970d74 · 17 hours ago 2 Commits

README.md	first commit	17 hours ago
WA_Fn-UseC_-Telco-Customer-Churn.csv	Add files via upload	17 hours ago
churn_model_lr.joblib	Add files via upload	17 hours ago
data_scaler.joblib	Add files via upload	17 hours ago
ispa report.pdf	Add files via upload	17 hours ago
pro.html	Add files via upload	17 hours ago
pro.ipynb	Add files via upload	17 hours ago

README

Inferential-Statistics-Case-Study-on-Customer-Churn-Prediction

About

No description, website, or topics provided.

Readme
Activity
0 stars
0 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

HTML 56.2% Jupyter Notebook 43.8%

Suggested workflows
Based on your tech stack