Phase-3 Submission Template

Student Name: D.RAGUL

Register Number: 23uit039

Institution: AVS COLLEGE OF TECHNOLOGY

Department: B.TECH INFORMATION TECHNOLOGY

Date of Submission: 15/05/2025

GitHub Repository Link: https://github.com/Raguldm012/Predicting-customer-churn-using-machine-learning-to-uncover-hidden-pattern

---

1. Problem Statement

Predicting customer churn using machine learning to uncover hidden patterns We aim to predict customer churn—i.e., which customers are likely to stop using a company's services—in order to proactively retain them. Churn significantly impacts revenue and growth, especially in subscription-based or competitive industries like telecom, banking, or SaaS. This is a **classification problem**, where the model learns to classify customers as "churn" or "no churn" based on historical behavioural and demographic data. Identifying hidden patterns that lead to churn helps businesses optimize retention strategies and reduce customer loss.

2. Abstract

Customer churn poses a significant challenge for businesses, affecting revenue and growth. This project aims to predict customer churn using machine learning techniques to identify hidden patterns in customer behaviour. By analysing historical data, we built predictive models using algorithms such as logistic regression, decision trees, and random forests. The approach involved data pre-processing, feature engineering, model training, and evaluation. Our models achieved high accuracy and revealed key factors contributing to churn, such as service usage and customer support interactions. The insights can help businesses implement targeted retention strategies. Overall, the project demonstrates how machine learning can proactively address churn and enhance customer loyalty.

3. System Requirements

- **Hardware:** Minimum 8GB RAM, i5 processor or higher (for efficient model training)
- **Software:**
  - **Python:** Version 3.8 or above ○ **Libraries:** pandas, numpy, scikit-learn, matplotlib, seaborn ○ **IDE:** Google Colab or Jupyter Notebook

4. Objectives

The main goal is to predict customer churn using machine learning to help businesses retain customers. Expected outputs include a trained model that classifies customers as likely to churn or stay, key features influencing churn, and actionable insights. This helps reduce revenue loss, improve customer satisfaction, and support data-driven retention strategies.

5. Flowchart of Project Workflow

- **Data Collection** – Gather customer data (e.g., usage, demographics, support history)
- **Pre-processing** – Clean missing values, encode categorical variables
- **EDA (Exploratory Data Analysis)** – Visualize trends, identify patterns
- **Feature Engineering** – Create meaningful features for modelling
- **Modelling** – Train ML models (e.g., logistic regression, random forest)
- **Evaluation** – Assess model accuracy, precision, recall
- **Deployment** – Deploy model via web or dashboard for real-time prediction

from graphviz import Digraph # Create a directed graph dot =

Digraph(comment='Customer Churn Prediction Workflow')

Define nodes dot. Node('A',

'Data Collection') dot.

Node('B', 'Data Preprocessing') dot. Node('C',

'Exploratory Data Analysis

(EDA)') dot.node('D', 'Feature

Engineering') dot. Node ('E',

'Modelling') dot. Node ('F',

'Evaluation') dot. Node ('G',

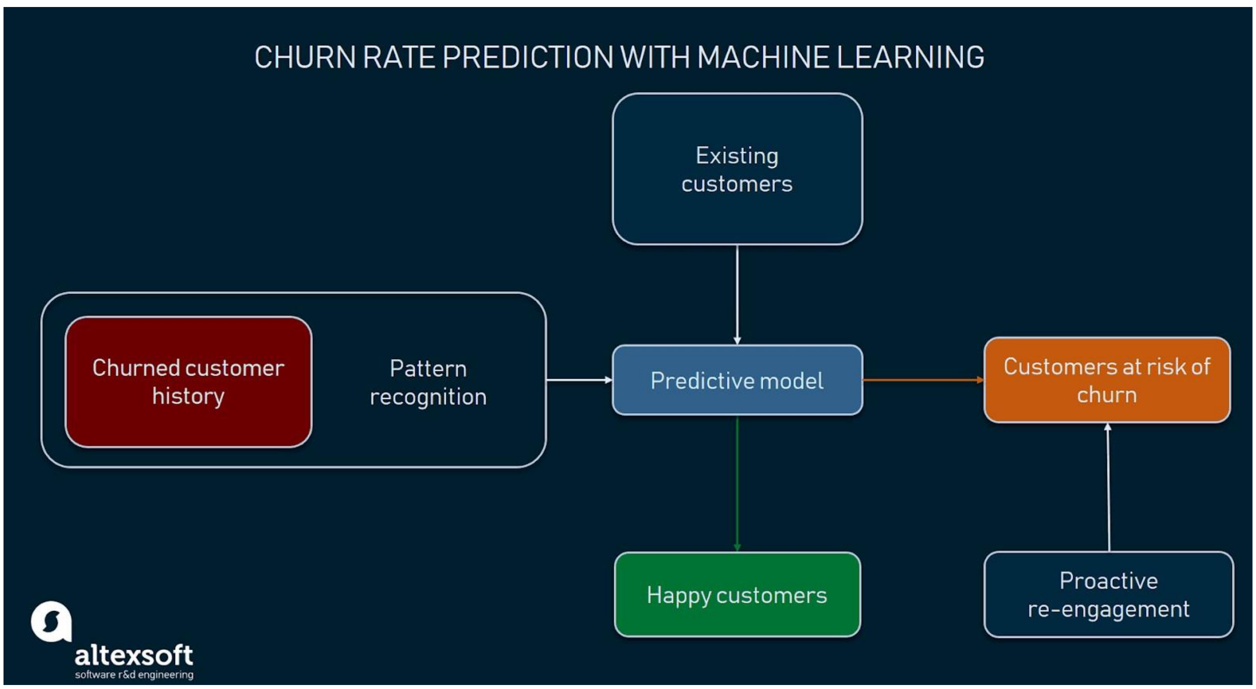
'Deployment')

define edges dot. Edges (['AB', 'BC', 'CD',

'DE', 'EF', 'FG'])

Render the diagram dot. Render('customer_churn_workflow',

format='png', cleanup=False)



6. Dataset Description

- **Source:** Kaggle (e.g., "Telco Customer Churn" dataset)
- **Type:** Public

- **Size and Structure:** ~7,000 rows × 21 columns
- **Content:** Includes customer demographics, services used, account

info, and churn status import pandas as pd df =

pd.read_csv('your_dataset.csv') df. Head()

Feature	Description	Type
`customerID`	Unique customer identifier	Categorical (ID)
`SeniorCitizen`	Senior citizen flag (0 = No, 1 = Yes)	Binary
`Partner`	Has a partner (Yes/No)	Categorical
`Dependents`	Has dependents (Yes/No)	Categorical
`tenure`	Months with the company	Numerical (Integer)
`PhoneService`	Phone service status	Categorical
`InternetService`	Type of internet (DSL/Fiber/No)	Categorical
`Contract`	Contract type (Month-to-month, etc.)	Categorical
`PaymentMethod`	Method of payment	Categorical
`MonthlyCharges`	Monthly billing amount	Numerical (Float)
`TotalCharges`	Total billing amount	Numerical (Float)
`Churn`	Target: Churned or not (Yes/No)	Categorical (Label)

7. Data Preprocessing

- **Missing Values:** Removed or imputed null entries (e.g., total charges)
- **Duplicates:** Dropped duplicate rows to ensure data integrity
- **Outliers:** Detected and handled via statistical methods (e.g., IQR)
- **Encoding:** Converted categorical variables using Label Encoding and One-Hot Encoding
- **Scaling:** Applied StandardScaler to numerical features for uniformity

Before Preprocessing

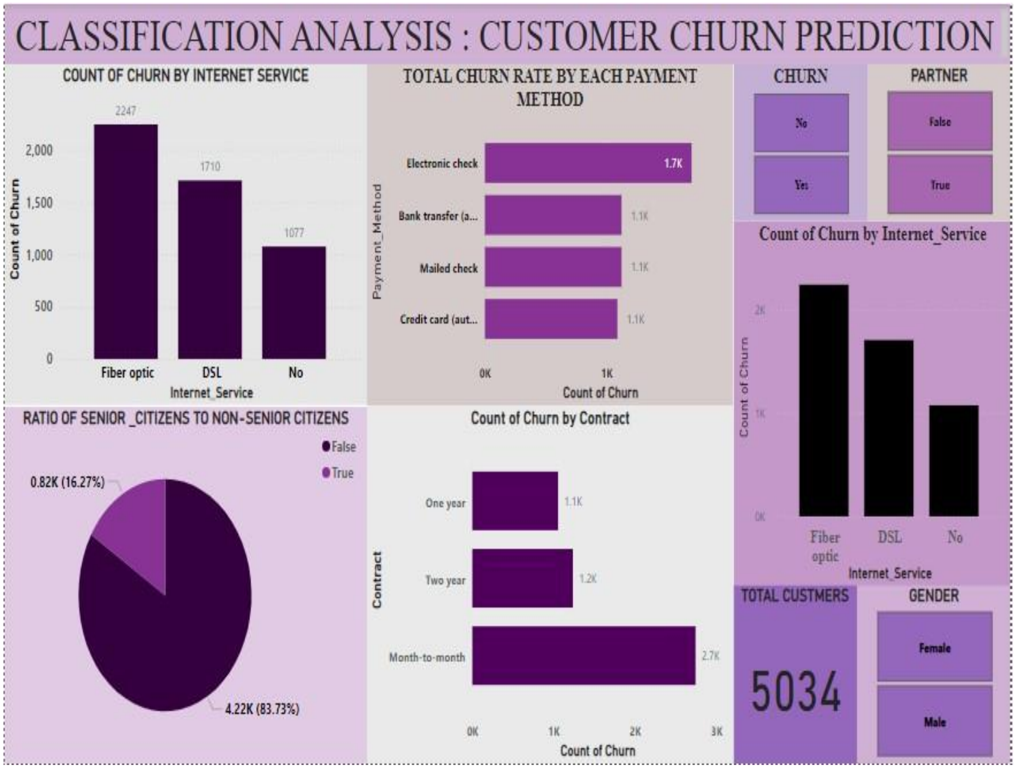
After Preprocessing

Feature	Example Values	Issues
`TotalCharges`	"123.45", " ", "250.75"	Missing values as blank strings
`Churn`	"Yes", "No"	Categorical - needs encoding
`gender`	"Male", "Female"	Categorical - needs encoding
`SeniorCitizen`	0, 1	Already binary
`PaymentMethod`	"Mailed check", "Electronic check", etc.	High cardinality categorical

Feature	Example Values	Changes Made
`TotalCharges`	123.45, NaN filled with median/mean	Converted to numeric, missing fixed
`Churn`	1 (Yes), 0 (No)	Label encoded
`gender`	1 (Male), 0 (Female)	Binary encoded
`PaymentMethod`	[0,0,1,0] (One-hot vector)	One-hot encoded
`tenure`	Standardized or scaled	Normalized for model input

8. Exploratory Data Analysis (EDA)

- **Visual Tools Used:**
  - Histograms for distribution of features
  - Boxplots to detect outliers
  - Heatmap to show correlations between features
- **Key Insights:**
  - Customers with longer tenure and fiber optic services are more likely to churn
  - High monthly charges correlate with higher churn
  - Contract type, tech support, and online security significantly impact churn



9. Feature Engineering

- **New Features:** Created features like *Average Monthly Spend* and *Tenure Groups* to capture customer behavior trends
- **Feature Selection:** Used correlation analysis and feature importance (e.g., from Random Forest) to select top predictors
- **Transformations:** Applied log transformation to skewed features and scaled numerical data for better model performance
- **Impact:** Selected features like *Contract Type*, *Monthly Charges*, and *Tech Support* strongly influence churn prediction by highlighting customer commitment and satisfaction levels.

10. Model Building

- **Models Tried:**

  - **Baseline:** Logistic Regression (simple and interpretable for churn prediction)
  - **Advanced:** Random Forest (for handling feature interactions), Gradient Boosting (to improve accuracy), and XGBoost (for performance optimization)

- **Model Choice Rationale:**

  - **Logistic Regression** was chosen as a baseline for its simplicity and interpretability.
  - **Random Forest** and **XGBoost** were selected for their ability to handle complex, non-linear relationships and improve predictive accuracy.

11. Model Evaluation

- **Evaluation Metrics:**

  - **Accuracy:** Measures overall model performance
  - **F1-score:** Balances precision and recall, especially important for imbalanced churn data
  - **ROC-AUC:** Evaluates model's ability to discriminate between churned and non-churned customers
  - **RMSE (Root Mean Squared Error):** Measures prediction error for regression-based models (if applicable)

- **Visuals:**

  - **Confusion Matrix** to show true vs. predicted churn values
  - **ROC Curve** to assess model's classification ability across different thresholds

- **Model Comparison Table:**

  - Compare the metrics (accuracy, F1, ROC, etc.) for each model (Logistic Regression, Random Forest, XGBoost)
  - Highlight best-performing model

12. Deployment

- Deployment Method: Google Colab

  - Public Link: https://colab.research.google.com/drive/1j-MLVad6ITcz9jR7BKZexp-tzvmHZOtm#scrollTo=Pn17f-d-xVr0&uniqifier=1

13. Source code

Import libraries

import pandas as pd

from

sklearn.model_sele

ction import

train_test_split

from

sklearn.preprocessi

ng import

LabelEncoder,

StandardScaler

from

sklearn.linear_mod

el import

LogisticRegression

from sklearn.metrics import accuracy_score, confusion_matrix,

classification_report # Load dataset df =

pd.read_csv('customer_churn.csv')

Drop irrelevant columns df.drop(['customerID'], axis=1,

inplace=True) # Convert TotalCharges to numeric df['TotalCharges']

= pd.to_numeric(df['TotalCharges'], errors='coerce')

Handle missing values

df['TotalCharges'].fillna(df['TotalCharges'].median(), inplace=True)

Encode categorical features label_enc

= LabelEncoder()

binary_cols = ['gender', 'Partner', 'Dependents', 'PhoneService', 'PaperlessBilling', 'Churn'] for col in

binary_cols:

df[col] = label_enc.fit_transform(df[col]) #

One-hot encode other categorical columns df

= pd.get_dummies(df, columns=[

'MultipleLines', 'InternetService', 'OnlineSecurity', 'OnlineBackup',

'DeviceProtection', 'TechSupport', 'StreamingTV', 'StreamingMovies',

'Contract', 'PaymentMethod'])

# Define features and target
X = df.drop('Churn', axis=1) y

= df['Churn']

Split dataset

X_train, X_test, y_train, y_test = train_test_split(X, y,

test_size=0.2, random_state=42) # Feature scaling scaler =

StandardScaler()

X_train = scaler.fit_transform(X_train)

X_test = scaler.transform(X_test)

Train model model =

LogisticRegression()

model.fit(X_train, y_train) #

Make predictions y_pred =

model.predict(X_test)

```
# Evaluate print("Accuracy:", accuracy_score(y_test, y_pred))

print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))

print("Classification Report:\n", classification_report(y_test, y_pred))
```

14. Future scope

- **Real-time Prediction:** Enhance the model to provide real-time churn predictions by integrating it with live customer data streams via APIs, allowing businesses to act proactively.
- **Model Improvement:** Explore more advanced models like deep learning (e.g., neural networks) to further improve prediction accuracy, especially for complex customer behavior patterns.

- **Customer Segmentation:** Implement customer segmentation based on churn risk to create targeted retention strategies, providing personalized interventions for different customer groups.

13. Team Members and Roles

TEAM LEADER : RAGUL D
RESEARCHER: AKASH E
DEVELOPER: HARISH S
DESIGNER: SUBASH C
TESTER: FRANKLIN M