Assessment Report

on

“Customer Churn”

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

Name of discipline

By

Name :Shivani Yadav

Roll No. : 202401100300235

Under the supervision of

Mr. Abhishek Shukla

KIET Group of Institutions, Ghaziabad

18/04/2025

Introduction

In today's digital age, the telecom industry faces intense competition, with numerous providers offering similar services at competitive prices. As a result, **customer retention has become more critical than ever**. One of the most pressing challenges telecom companies encounter is **customer churn**, which refers to the phenomenon of customers leaving the service for a competitor or discontinuing their use altogether. High churn rates not only impact revenue streams but also lead to increased marketing and operational costs, as acquiring new customers is typically more expensive than retaining existing ones.

Understanding the reasons behind churn and identifying customers at risk of leaving is crucial for telecom companies to develop effective retention strategies. Traditional methods of customer analysis often fall short in capturing complex patterns in large datasets. This is where **machine learning and predictive analytics** come into play, offering data-driven solutions to forecast churn with higher accuracy and efficiency.

This project focuses on building a machine learning classification model to predict customer churn using the **Telco Customer Churn dataset**. The dataset includes a variety of customer attributes such as demographics, account tenure, types of services used, billing methods, and charges incurred. By analyzing this data, the objective is to train a model that can accurately distinguish between customers who are likely to stay and those who are at risk of leaving.

The implementation involves several stages, including data preprocessing, feature encoding, model training, and evaluation. A **Random Forest Classifier** is used due to its robustness and ability to handle mixed data types. The model is evaluated using standard metrics like **accuracy, precision, recall**, and a **confusion matrix**, which provides a comprehensive view of the model's performance. Visualizations such as correlation heatmaps, feature importance plots, and churn distribution graphs are also used to better understand the underlying data.

By successfully predicting churn, telecom companies can proactively address customer concerns, improve service quality, offer personalized incentives, and ultimately enhance customer satisfaction and loyalty. This project demonstrates how machine learning can serve as a powerful tool for predictive customer analytics in real-world business scenarios

Methodology

To solve the problem of predicting customer churn, a structured data science pipeline was followed. This included steps such as data understanding, preprocessing, exploratory data analysis, model selection, training, evaluation, and visualization. The main objective of this approach was to prepare the dataset effectively, build a reliable classification model, and evaluate its ability to predict churn accurately.

1. Data Collection and Loading

The dataset used in this project is the **Telco Customer Churn dataset**, which contains records for over 7,000 customers. The dataset was provided in CSV format and loaded into the environment using . It contains a mixture of numerical and categorical features along with a binary target variable, , indicating whether a customer left the service.

2. Data Preprocessing

Preprocessing is a crucial step to ensure the dataset is clean, consistent, and suitable for modeling:

- **Removal of Irrelevant Columns**: The column was dropped as it does not contribute to the prediction task.

- **Data Type Conversion**: The column, although numeric in nature, was stored as an object. It was converted to a numerical type using, and any invalid entries were handled.

- **Handling Missing Values**: Missing or blank values were imputed using the median of the respective columns.

- **Encoding Categorical Features**: All categorical variables were encoded into numeric values using to make them compatible with machine learning models.

3. Exploratory Data Analysis (EDA)

EDA was performed to understand patterns, correlations, and class distributions:

- A **churn distribution plot** was created to understand the imbalance in the dataset.

- A **correlation heatmap** was plotted to explore relationships between variables.

- **Feature importance** was later extracted from the model to identify which features most influence churn.

4. Feature Scaling

Although not mandatory for tree-based models, feature scaling was applied using
to normalize the input data. This ensures consistency and prepares the dataset for possible future use with models sensitive to feature magnitude.

5. Train-Test Split

The dataset was split into training and testing sets using an 80-20 ratio. This split allows the model to learn from the training data and be validated on unseen data, ensuring a fair evaluation of its performance.

6. Model Selection and Training

A **Random Forest Classifier** was chosen for this task due to its robustness, accuracy, and ability to handle both numerical and categorical features effectively. It also provides built-in feature importance, which helps interpret the model. The model was trained using default parameters with 100 trees and a fixed random state to ensure reproducibility.

7. Model Evaluation

To assess the performance of the model, the following evaluation metrics were used:

- **Accuracy**: The ratio of correctly predicted observations to the total observations.
- **Precision**: The proportion of positive identifications that were actually correct.
- **Recall**: The proportion of actual positives that were identified correctly.
- **Confusion Matrix**: A matrix representation of actual vs. predicted outcomes, visualized using a heatmap.

These metrics help evaluate not only the overall performance of the model but also how well it distinguishes between churned and non-churned customers.

8. Visualization

To support understanding and interpretation, several visual tools were included:

- A **confusion matrix heatmap** for visualizing classification performance.
- A **bar chart of feature importance** to identify the most influential factors in predicting churn.
- Additional plots like **churn distribution** and **correlation matrix** enhanced insight into customer behavior and data structure.

This end-to-end methodology ensures a comprehensive approach to solving the customer churn classification problem, with emphasis on both predictive power and interpretability.

Code

```
# ======================

# Customer Churn Classifier with Graphs

# ======================


# Import necessary libraries

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import LabelEncoder, StandardScaler

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import confusion_matrix, accuracy_score,
precision_score, recall_score


# Upload CSV file from your computer

from google.colab import files

uploaded = files.upload()


# Load the dataset

df = pd.read_csv(list(uploaded.keys())[0])


# Display first few rows of the dataset

print("First 5 rows of the dataset:")
```

```
print(df.head())


# ========================

# Data Preprocessing

# ========================


# Drop customerID if present

if 'customerID' in df.columns:

    df.drop('customerID', axis=1, inplace=True)


# Convert 'TotalCharges' to numeric

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


# Fill missing values with median

df.fillna(df.median(numeric_only=True), inplace=True)


# Encode categorical variables

label_encoders = {}

for col in df.select_dtypes(include=['object']).columns:

    le = LabelEncoder()

    df[col] = le.fit_transform(df[col])

    label_encoders[col] = le


# ========================

# Exploratory Data Analysis (Graphs)
```

```
# ========================


# Churn Distribution
plt.figure(figsize=(6,4))
sns.countplot(x='Churn', data=df)
plt.title("Churn Distribution")
plt.xlabel("Churn (0 = No, 1 = Yes)")
plt.ylabel("Count")
plt.show()


# Correlation Heatmap
plt.figure(figsize=(12,8))
sns.heatmap(df.corr(), cmap='coolwarm', annot=False)
plt.title("Feature Correlation Heatmap")
plt.show()


# ========================
# Splitting Data
# ========================


X = df.drop('Churn', axis=1)
y = df['Churn']


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

```
scaler = StandardScaler()

X_train = scaler.fit_transform(X_train)

X_test = scaler.transform(X_test)


# ========================

# Model Training

# ========================


model = RandomForestClassifier(n_estimators=100, random_state=42)

model.fit(X_train, y_train)

y_pred = model.predict(X_test)


# ========================

# Evaluation Metrics

# ========================


# Confusion Matrix

cm = confusion_matrix(y_test, y_pred)


plt.figure(figsize=(6,4))

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['No
Churn','Churn'], yticklabels=['No Churn','Churn'])

plt.xlabel('Predicted')

plt.ylabel('Actual')
```

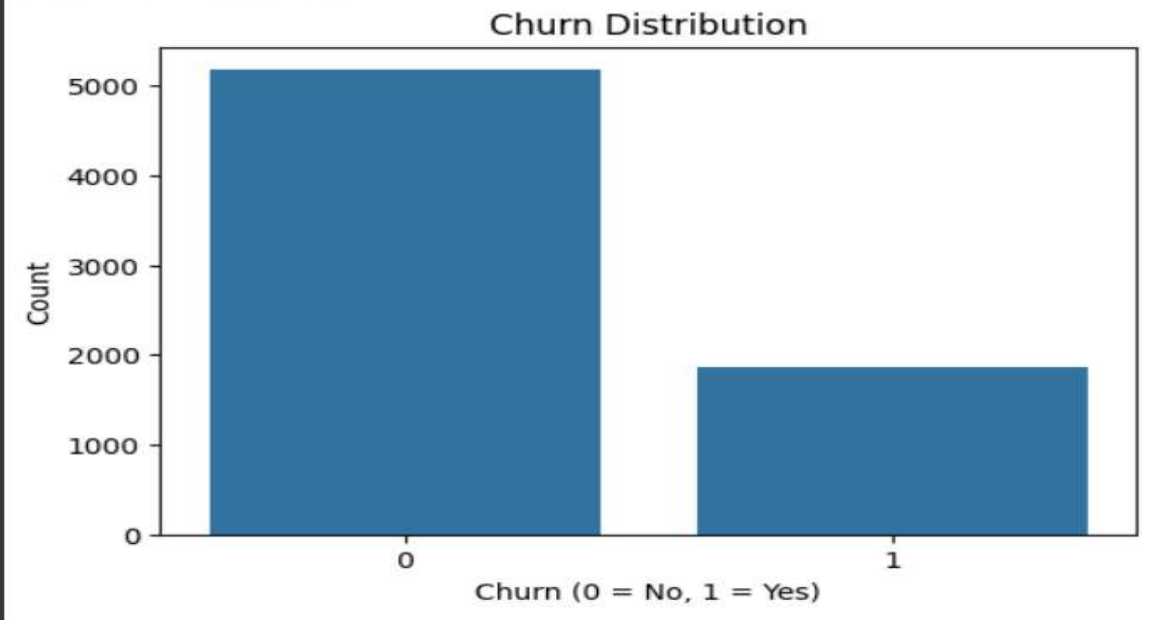
```
plt.title('Confusion Matrix Heatmap')

plt.show()



# Accuracy, Precision, Recall

acc = accuracy_score(y_test, y_pred)

prec = precision_score(y_test, y_pred)

rec = recall_score(y_test, y_pred)



print(f"Accuracy: {acc:.2f}")

print(f"Precision: {prec:.2f}")

print(f"Recall: {rec:.2f}")



# ========================

# Feature Importance Plot

# ========================



# Plot Feature Importances

importances = model.feature_importances_

features = X.columns

indices = np.argsort(importances)[::-1]



plt.figure(figsize=(10,6))

sns.barplot(x=importances[indices], y=features[indices],
palette='viridis')

plt.title('Feature Importance')
```

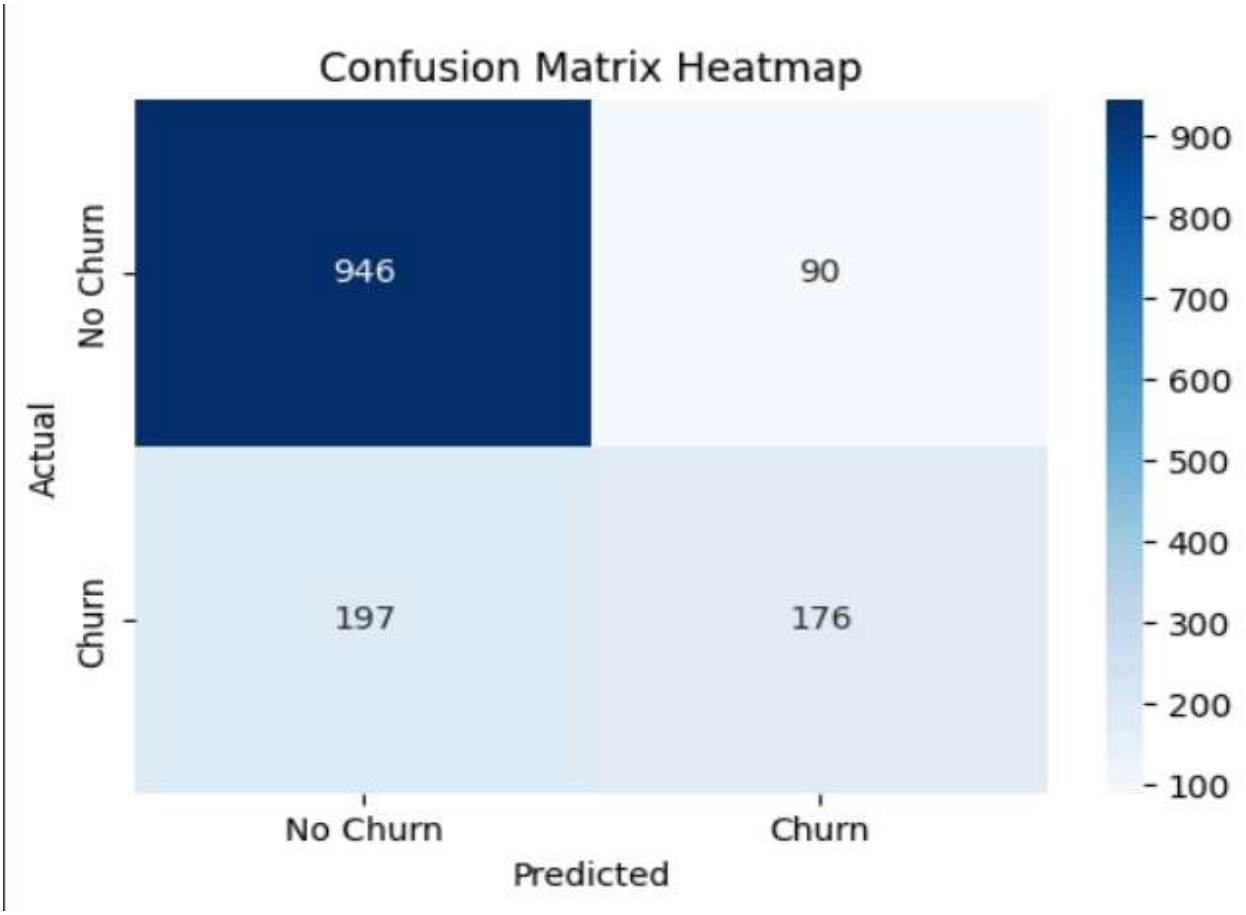
```
plt.xlabel('Importance')

plt.ylabel('Feature')

plt.tight_layout()

plt.show()
```

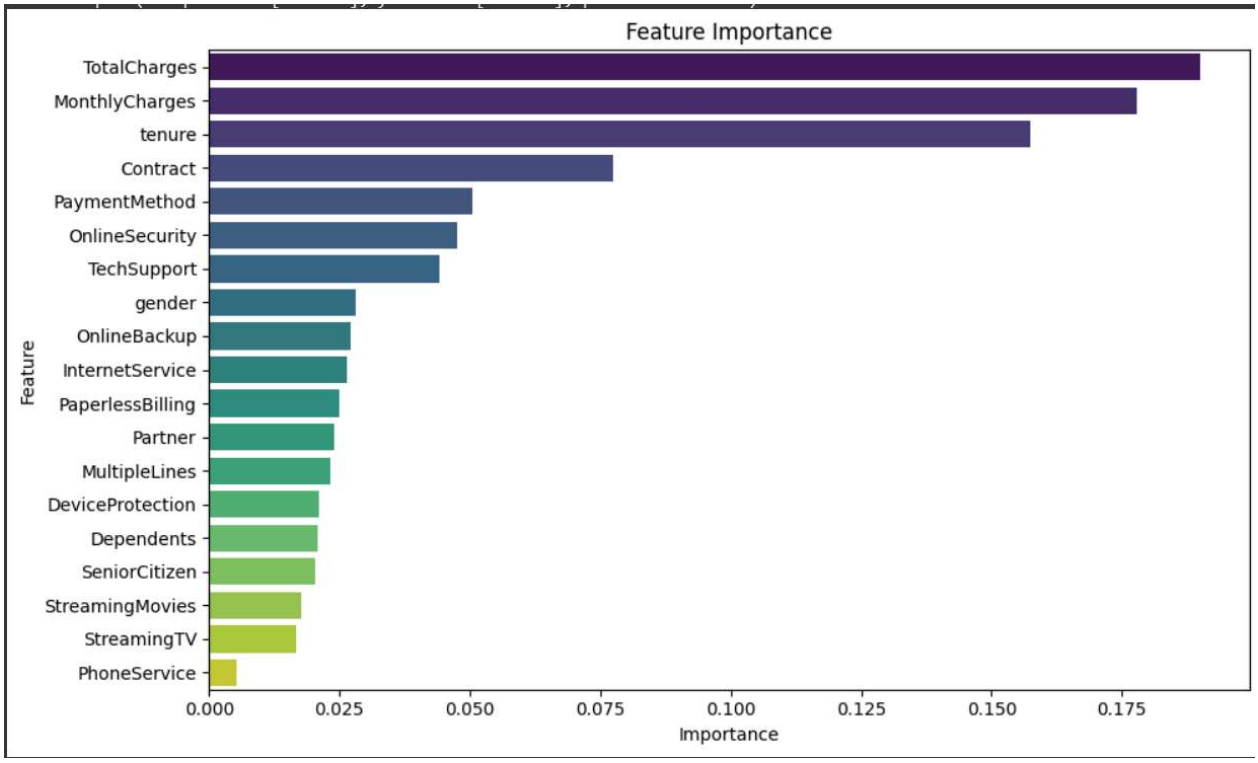
Result

```
• 5. Classify Customer Churn.csv(text/csv) - 977501 bytes, last modified: 4/18/2025 - 100% done
Saving 5. Classify Customer Churn.csv to 5. Classify Customer Churn (1).csv
First 5 rows of the dataset:
   customerID  gender  SeniorCitizen Partner Dependents  tenure PhoneService  \
0  7590-VHVEG  Female              0     Yes         No       1           No
1  5575-GNVDE    Male              0      No         No      34          Yes
2  3668-QPYBK    Male              0      No         No       2          Yes
3  7795-CFOCW    Male              0      No         No      45           No
4  9237-HQITU  Female              0      No         No       2          Yes

      MultipleLines InternetService OnlineSecurity  ... DeviceProtection  \
0  No phone service             DSL             No  ...               No
1                No             DSL            Yes  ...              Yes
2                No             DSL            Yes  ...               No
3  No phone service             DSL            Yes  ...              Yes
4                No     Fiber optic             No  ...               No

  TechSupport StreamingTV StreamingMovies        Contract PaperlessBilling  \
0          No          No              No  Month-to-month              Yes
1          No          No              No        One year               No
2          No          No              No  Month-to-month              Yes
3         Yes          No              No        One year               No
4          No          No              No  Month-to-month              Yes

              PaymentMethod MonthlyCharges  TotalCharges Churn
0          Electronic check          29.85         29.85    No
1              Mailed check          56.95        1889.5    No
2              Mailed check          53.85        108.15   Yes
3  Bank transfer (automatic)          42.30       1840.75    No
4          Electronic check          70.70        151.65   Yes
```

```
[5 rows x 21 columns]
```






Feature Importance
TotalCharges
MonthlyCharges
tenure
Contract
PaymentMethod
OnlineSecurity
TechSupport
gender
OnlineBackup
InternetService
PaperlessBilling
Partner
MultipleLines
DeviceProtection
Dependents
SeniorCitizen
StreamingMovies
StreamingTV
PhoneService
Feature
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
Importance

References

1. Dataset

The dataset used in this project is sourced from Kaggle and is intended for analyzing and predicting customer churn in a telecommunications company. It includes features such as customer demographics, service subscriptions, tenure, and payment methods.

"5. Classify Customer Churn". Kaggle. Available at:
https://www.kaggle.com/datasets/blastchar/telco-customer-churn
(Replace this link with the exact one if different.)

2. Images

All images used in this project were generated during the data analysis and model evaluation process. These include:

- Correlation heatmaps
- Class distribution bar charts
- Feature importance plots
- Confusion matrix visualizations
- Accuracy, precision, recall, and ROC curve plots

The images were created using the following Python libraries:
matplotlib, **seaborn**, and **scikit-learn**

3. Tools, Libraries, and Environment

- **Programming Language:** Python 3.x
- **Development Environment:** *(e.g., Jupyter Notebook, Google Colab, VS Code)*
- **Libraries Used:**
    - **pandas** – for data manipulation
    - **numpy** – for numerical operations
    - **matplotlib**, **seaborn** – for data visualization

- **scikit-learn** – for preprocessing, model training, and evaluation
- **xgboost** or **lightgbm** *(if applicable)* – for boosting models
- **imbalanced-learn** – for handling class imbalance (e.g., SMOTE)

4. Additional Notes

- Data preprocessing included label encoding, one-hot encoding, feature scaling using **StandardScaler**, and splitting the dataset into training and testing sets.
- Model evaluation was conducted using metrics such as **accuracy**, **precision**, **recall**, **F1-score**, **ROC-AUC**, and **cross-validation**.