PROJECT REPORT

On

Competitor Influence On Customer Churn Using Machine Learning Algorithms

Submitted to

LOVELY PROFESSIONAL UNIVERSITY

in partial fulfillment of the requirements for the award of degree of

Bachelor of Computer Applications

Submitted By:-

[Sailesh Goel 12203461]

[AdityaShankar 12204270]

[Manveer Singh 12223006]

[Anushka Arora 12208629]

[Sourabh Kumar 12212794]

Supervised By:-

[Mrs. Satwinder Kaur 28958]

[Assistant Professor]

LOVELY FACULTY OF TECHNOLOGY & SCIENCES

LOVELY PROFESSIONAL UNIVERSITY

PUNJAB

[MAY 2024]

To whom so ever it may concern

We, **Sailesh Goel UID:12203461, Aditya Shankar UID:12204270, Manveer Singh UID:12223006 Anushka Arora UID:12208629 and Sourabh Kumar UID:12212794**, hereby declare that the work done by us on "**Competitor Influence On Customer Churn Using Machine Learning Algorithms** " under "**Mrs. Satwinder Kaur (UID:28958)**" from **January 2024** to **May 2024** at Lovely professional University, Phagwara, Punjab, is a record of original work for the partial fulfilment of the requirements for the award of the degree Bachelors Of Computer Applications. The results embodied in this report have not been submitted to any other subject or university.

Submitted By :-

Sailesh Goel	Aditya Shankar	Manveer Singh	Anushka Arora	Sourabh Kumar
12203461	12204270	12223006	12208629	12212794

CERTIFICATE

This is to certify that the project report titled "**Competitor Influence On Customer Churn Using Machine Learning Algorithms**" is being submitted by Sailesh Goel (UID:12203461), Aditya Shankar (UID:12204270), Manveer Singh (UID:12223006), Anushka Arora (UID:12208629) and Sourbh Kumar (UID:12212794), in **BCA 6th semester** is a record of bonafide work carried out by them. The results embodied in this report have not been submitted to any other University for the award of any degree.

Signature and Name of the Supervisor : Mrs. Satwinder Kaur (UID :28958)

Designation : Assitant Professor, SCA

Date :

Signature of AO:

Date :

ACKNOWLEDGEMENT

For all the efforts behind the paperwork & project, we first & for most would like to express our sincere appreciation to **Mrs.Satwinder Kaur (UID:)**, School of Computer Applications , for their extended help & suggestions at every stage of this project work. It is with a great sense of gratitude that we acknowledge the support, time to time suggestions and highly indebted to our guide. Finally, we pay sincere thanks to all those who indirectly and directly helped us towards the successful completion of the real time project **"Competitor Influence On Customer Churn Using Machine Learning Algorithms**" for End Term Practical.

With due regards

Sailesh Goel	Aditya Shankar	Manveer Singh	Anushka Arora	Sourabh Kumar
12203461	12204270	12223006	12208629	12212794

TOPIC APPROVAL PERFORMA

School of Computer Application (SCA)

Program : P124::BCA

COURSE CODE : CAP463 **REGULAR/BACKLOG :** Regular **GROUP NUMBER :** CARGC0048

Supervisor Name : Satwinder Kaur **UID :** 28958 **Designation :** Assistant Professor

Qualification : _____ **Research Experience :** _____

SR.NO.	NAME OF STUDENT	Prov. Regd. No.	BATCH	SECTION	CONTACT NUMBER
1	Manveer Singh	12223006	2022	D2228	7717779003
2	Sourav Kumar	12212794	2022	D2202	9872192060
3	Anushka Arora	12208629	2022	D2203	9465193062
4	Aditya Shankar	12204270	2022	D2201	9905893255
5	Sailesh Goel	12203461	2022	D2201	9877764427

SPECIALIZATION AREA : Databases **Supervisor Signature:** _____

PROPOSED TOPIC : Competitor Influence on Customer Churn

Qualitative Assessment of Proposed Topic by PAC		
Sr.No.	**Parameter**	**Rating (out of 10)**
1	Project Novelty: Potential of the project to create new knowledge	7.54
2	Project Feasibility: Project can be timely carried out in-house with low-cost and available resources in the University by the students.	7.51
3	Project Academic Inputs: Project topic is relevant and makes extensive use of academic inputs in UG program and serves as a culminating effort for core study area of the degree program.	7.54
4	Project Supervision: Project supervisor's is technically competent to guide students, resolve any issues, and impart necessary skills.	7.54
5	Social Applicability: Project work intends to solve a practical problem.	7.54
6	Future Scope: Project has potential to become basis of future research work, publication or patent.	7.54

PAC Committee Members		
PAC Member (HOD/Chairperson) Name: Dr. Sartaj Singh	UID: 11303	Recommended (Y/N): Yes
PAC Member (Allied) Name: Dr. Pawan Kumar	UID: 11522	Recommended (Y/N): Yes
PAC Member 3 Name: Jasleen Kaur Paintal	UID: 30857	Recommended (Y/N): Yes

Final Topic Approved by PAC: **Competitor Influence on Customer Churn**

Overall Remarks: Approved

PAC CHAIRPERSON Name: 32898::Dr. Anand Kumar Shukla **Approval Date:** 04 Apr 2025

4/23/2025 1:47:44 PM

ABSTRACT

Customer churn poses a significant challenge to telecom service providers, affecting revenue and business sustainability. This paper proposes a predictive model for churn detection using machine learning techniques applied to structured customer data. The dataset includes demographic, service usage and customer account information. Initial preprocessing steps involve handling missing values, transforming data types and categorical variables also normalizing numerical data. Three classifiers [Random Forest, Logistic Regression and XGBoost] are employed to identify potential churners. To analyze the effectiveness of model the data is distributed into training and testing also it's performance is measured using metrics like accuracy, confusion metrics and ROC AUC score. Among the tested models, XGBoost demonstrates superior performance in predicting churn behavior. To demonstrate the real-world applicability of the model, a web-based interface showcasing churn predictions was developed. The proposed methodology can help telecom providers tke proactive steps to retain high risk customers, thereby improving customer satisfaction and reducing churn rate.

Keywords:- Customer churn, Telecom industry, Predictive Modeling, Churn Detection, Customer Retention, Risk Prediction, Business Sustainability.

Table Of Content

S.No:-	Title	Page No.:-
1	CHAPTER-1 Introduction	7
2	1.1 Motivation to work	8
3	1.2 Problem Statement	9
4	CHAPTER-2 Implementation of Project	10
5	2.1 Methodology	11
6	2.2 Existing System	12
7	2.3 Proposed System	13
8	2.4 Machine Learning Techniques	14
9	2.5 Algorithms Used	16
10	2.6 Hardware/Software Requirement	17
11	2.7 Sample Code	18
12	2.8 Dataset Details	27
13	CHAPTER-3 Result And Discussion	28
14	3.1 Web-based Interface and Dashboard	37
15	3.2 Website Development	39
16	CHAPTER-4 Conclusion and Future Scope	43
17	CHAPTER-5 Appendix and References	45

CHAPTER-1
INTRODUCTION

In the contemporary environment of fierce competition, retaining customers is an immediate concern for all telecom companies. The ongoing battle to acquire and retain subscribers, coupled with the need to understand why customers churn, places enormous pressure on service providers. Churning is when customers stop using services with the provider, and it is directly correlated to revenue, profitability and sustainability. Acquiring a new customer is significantly more costly than retention, so churn management becomes a business strategy.

Of the many causes contributing to customer attrition, Competitive influence has taken prominence as the leading driver of churn. When competitors and the marketplace introduce aggressive marketing activity, a better offer, or even a better user experience, satisfied customers will leave. In the fast-moving telecommunications market, just having satisfied customers is not enough. Organizations need to analyze both customer behavior and factors in the marketplace, internally and externally, to understand any potential for churn earlier and to develop a retention action plan.

The project provides a machine learning based framework to assess competitive customer churn. We are able to blend traditional customer variables like service usage, billing and demographics, with market variables such as competitive offers, promotional triggers and bundles into the developed machine learning framework, to create strong predictive capabilities. An important feature of this approach is the use of analytics and AI framework to inform real time decisions. This framework can not only predict churn but identify which competitive factors matter most, so companies can focus their efforts on high risk customers with personalized interventions.

To increase usability and accessibility, a website was constructed. This website incorporates all the necessary features including churn prediction, swilling churn trends, model evaluation scores, and key feature information. Constructed using HTML, CSS JavaScript and visualization libraries, the website is an all-encompassing system where various stakeholders can titeractively examine customer churn situations, run churn predictions using Gradio, and analyze actionable insight through intuitive dashboards.

MOTIVATION FOR THE WORK

The driving force behind this project is the increasing difficulty that telecom providers are facing in consumer retention as competitive pressures in the marketplace grow. As digital access expands and customer expectations rise, consumers are more and more aware of other options. Even satisfied customers will switch providers if they are enticed by better pricing, superior features, or incentives from competitive providers. This increased incidence of competitive induced churn needs smarter ways to perform consumer retention through data.

Churn detection and retention methods currently used often focus on internal factors such as billing disputes, service complaints or product usage patterns. These methods miss the external pressures that can be generated by competitor pressure so if a service provider were to rely purely on internal indicators they would miss valuable early warning signs of consumer churn due to external pressures from competitors.

This research aims to bridge that gap through the construction of a machine-learning model that contemplates both internal customer behavior and external factors in the market to make reliable churn predictions. The proposed solution employs more sophisticated classifiers, such as Random Forests and XGBoost, which provide greater detail and enable our system to harness complex patterns in customer data and make real-time predictions. The use of a web-based interface will further enhance our solution, adding an interactive component that will allow managers to run scenario-based simulations and analyze current risk from competitors in a reliable time frame.

By combining all of the above, the proposed system provides telecom providers with the tools to better anticipate and minimize competitive intent in an increasingly volatile market, contributing to more informed decision making by managers and more effective customer retention in the market.

PROBLEM STATEMENT

Customer churn is a common concern for telecom service providers. In this hyper competitive environment where rival companies are actively launching new offers, price drops and value added services it is more difficult than ever for service providers to keep their customers. Traditional churn models identify characteristics that can indicate churn – diappointment with the service received or technical complaints and faults etc. However these approaches do not include one of the biggest drivers of churn – competitive influence.

Many customers do not churn because they are dissatisfied, but they churn because their perception of value at a rival or a new decision they wish to make is perceived to be greater than their current provider. A competing advertisement or competing promotional offer, new pricing model, a new technology or service are all external triggers on a customer to churn. All of the traditional analytical tools do not satisfyably take into account these competitive triggers typically leading to predictions that are far too late or inaccurate.

In addition, the interplay of emotional and rational decision-making with respect to customers' decision, makes it hard to precisely pinpoint competitors' role to ultimately provide an agent in limiting churn so that it makes it difficult for telecoms to create preventative strategies and often they only react after the customer's missing.

We address this limitation by creating a predictive framework that integrates both customer behavior and competitors' relevant features using machine learning algorithms. By identifying at risk customers early, especially those influenced by the competitive market, the telecom will be able to run targeted retention campaigns, offer competitive counter incentives, and improve the customer experience. We provide an intelligent, proactive response to a long term systemic industry issue.

CHAPTER-2

IMPLEMENTATION OF PROJECT

OBJECTIVES

1. The main goal of this project is to study and identify the influence of competitive factors on customer churn among the telecom marketplace. Leveraging both internal behavioral characteristics and external market driven variables, the system seeks to provide a complete understanding of churn behavior.

2. To build machine learning based predictive models that classify customers as potential churners and non-churners. The operationalized machine learning models should adapt to continually changing competitive contexts and generalize across customer segments.

3. To assess and compare the performance of a number of supervised learning methodologies such as Logistic Regression, Random Forest and XGBoost based on accuracy, precision, recall, F1-score and AUC-ROC to identify the most appropriate model for real-world implementation.

4. To develop an interactive web application with Gradio that telecom managers can use by inputting a customer's parameters and receiving real-time churn predictions. This also includes the ability to run various "what-if" simulation scenarios by modifying competitive factors.

5. To create a responsive website with integrated dashboards and visualizations that allow both technical and non-technical users to investigate trends, understand model outputs and develop future strategic actions for customer retention..

METHODOLOGY

1. The project starts with data collection and cleanup. The dataset analyzed contains:' customer demographics, service usage patterns, contract and financial data. Competitive metrics such as 'churn score' and 'competitors bias index' indicant are also included in order to measure industry pressure on the organization. The dataset was cleaned, missing values were populated and categorical variables were encoded. The data was scaled to ensure that all the numerical variables would act as scale-neutral and not negatively impact model performance.

2. Descriptive analytics are performed using data visualization libraries in R (ggplot2) to identify an initial spark of patterns in terms of churn behavior. Histograms, bar charts, scatter plots, and heat maps are then created to show correlations, as well as trends of churn across tenure, monthly charges, and competitiveness.

3. The predictive engine consists of three essential machine learning models:

- We are using logistic regression as our baseline because of its interpretability.

- The random forest machine learning method can create a really large number of robust trees that can represent ensemble learning as well as complex non-linear relationships.

- The XGBoost machine learning method is being used because it contains superior performance and speed for gradient boosting.

4. The dataset will be divided in an 80:20 training and testing dataset ratio in order to train and test the model and thereby validate afterwards. Comparing the results of each classifier uses important metrics such as accuracy as well as confusion matrix and AUC- ROC.

5. We perform a feature importance analysis and identify churn score, monthly charges, tenure and competitive influence as high contributors to the churn prediction. We incorporate these insights into the model and use them to inform intervention strategies.

6. In order to enable real time decision making, a Gradio based module is built so that users enter their input variables and receive a churn probability score between 0 and 100. Users can also see how changes in contract length, competitive pressure, or service features affect this probability.

7. A front end website is built using HTML, CSS, JavaScript, and visualization libraries including Chart.js and D3.js. It contains the model outputs, comparison reports, and interactive visualizations that help to convey complex analytical results and reporting to stakeholders.

EXISTING SYSTEM

The vast majority of telecom companies today utilize traditional statistical models or simplistic rule based systems to monitor customer churn. Most churn systems utilize internal customer data like billing behaviors or customer support history to monitor for churn. Churn systems do not account for external pressure, like competitor pricing or promotions, that create voluntary churn. Some industries have started to explore automated churn prediction tools, however, in the telecom space, there are limited competitor aware models currently available. Standard churn models suffer high false negatives, as they fail to identify churners made during external market forces that create pressure for customers to churn. Also, the customer retention ability of churn systems is limited to real time capability and intuitive interface to move stakeholders across experiences. Existing customer retention platforms also do not simulate potential "What-if" scenarios that personalize the current experiences or churn predictions, and in turn will likely result in late or ineffective customer engagement activity.

PROPOSED SYSTEM

To address the shortcomings of current methods, the project will develop a machine learning driven churn prediction system that includes competitor dimension analysis. The churn prediction system uses a modular design that integrates predictive analytics, real time interactive analysis and visual storytelling for decision support.

1. Data:- The dataset provided from IBM has 7000 customer records, with variables falling under demographics, service history, bill payment and churn status. Competitive influence data is simulated and added to develop a complete view of customer motivation.

2. Feature Engineering:- The churn score and one competitive influence index are new variables engineered to illustrate competitive intensity. Other variables that represent churn potential include contract type, tenure, monthly charges, and tech support use.

3. Preprocessing:- There are multiple steps like label encoding, missing value treatment and standardization used in R and Python libraries to prepare a training dataset.

4. Model Building: The following models have been built.

• Logistic Regression: Accuracy= 87.15%, AUC-ROC=0.84

• Random Forest: Accuracy= 94.67%, AUC-ROC= 0.93

• XGBoost: Accuracy= 93.68%, AUC-ROC=0.91

From the performance metric it is clear that Random Forest performs best, with XGBoost next in performance.

5. Model Deployment: In order to serve the best performing model, AGradio-based module is built. Users can vary inputs (contract type, charges and competitor pressure) to develop instant churn predictins.

6. Website Development:-

- Constructed using HTML5, CSS3, and JavaScript
- The Dashbords created using Chart.js and D3.js
- User Login, interactive charts, real time updates, and a feature explanation section

7. Real Time Evaluation:- The system is evaluated under mock conditions for responsiveness, accuracy and use experience. Evaluation measures are prediction accuracy, runtime efficiency and user experience input from telecom stakeholders.

MACHINE LEARNING TECHNIQUES

To build a comprehensive and intelligent churn prediction model, the project will utilize a number of basic machine learning methods. These methods will collectively ground the modeling methods and enhance the model capability to be accurate and vulnerable for real world competitive environments.

1. Supervised Learning:- This technique is a major component of the churn prediction strategy. Supervised Learnings, processes models to use historical customer data, which included two two fratures, an input feature set = (tenant, monthly charges, competitor's influence) and a an output labeled target variable (churn yes or no). The model becomes bewteen the input feature set and target variable training to enable the program a prediction for new polymer based upon the relationship shared to the knew target vector.Supervised learning is of utmost importance in this data science project - systems can show identify relationship between complex factors in the environment - customer behavior, competition, even pricing.

2. Classification Modeling:- Since our target variable is categorical (churn or not) we will use classification algorithms - these procedures return a proability score for each instance and then classify it into one of the predefined class. and validate our models using classficiation metrics (accuracy, precision, recall, AUC-ROC).

3. Feature Engineering:- To enhance model learning, engineered features such as churn score and competitive influence index are included. Engineered features are derived variables that transform multiple data points into a single measurement and have more predictive power than the source data alone. Engaging in adequate feature engineering ultimately improves model learning as it increases its ability to recognize nuanced but relevant trends within the source data.

4. Model Evaluation Metrics:- The performance of each algorithm is assessed using a variety of metrics.

Accuracy: Gives the overall number of correct predictions.

Precision and Recall: Assessed how well the model dealt with imbalanced class sizes

F1-Score: The harmonic mean of precision and recall.

AUC - ROC Score: Evaluates how well the model discriminates across threshold levels.

5. Cross Validation:- To mitigate overfitting and assess generalizability, the model has undergone k-fold cross validation through adhering to the protocol outlined above. K-fold cross validation divides the dataset into 'k' parts and trains/evaluates the model across various combinations of the dataset to lends credence to the models reliability and stability in making predictions.

ALGORITHMS USED

This project used three machine learning classification algorithms to analyze the applications performance in predicting customer churn. Each algorithm was selected for its strengths in classification and recognizing success in churn prediction research.

I. Logistic Regression: a model designed for binary classification problems. It calculates the probability that a sample belongs to a particular class using a logistic (sigmoid) function.

Purpose: Served as a baseline model for churn prediction.

Strengths: Easy to implement, easy to interpret, and fast when the data is linearly separable.

Results: Accuracy: 87.15% AUC – ROC Score: 0.84

Weaknesses: Had difficulty correctly predicting churners, generating many false negatives.

II. Random Forest Classifier: an ensemble learning method that uses multiple decision trees during training and outputs the class that is the mode of the classes

Purpose: Identify non-linear relationships and interactions in the dependent variables.

Strengths: High accuracy, not easily overfit, and ability to investigate feature importance.

Results: Accuracy:94.67% AUC - ROC Score: 0.93

Highlights: Outperformed all models, and produced high precision and recall for each class.

III. XGBoost (Extreme Gradient Boosting): It is an optimized implementation of gradient boosting for speed and performance; it builds trees sequentially; each new tree corrects the errors made from the previous iterations.

Purpose: To provide a competitive model with a balance of accuracy and computational efficiency.

Strengths: It can internally handle missing values, regularization is built in to prevent overfitting,, it is scalable to large datasets.

Results: Accuracy:93.68% AUC – ROC Score: 0.91

- Highlights: Both libraries are competitive and show slight differences in accuracy; Random Forest had better accuracy than XGBoost, but that result was due to the state of the machine learning experiment and XGBoost had greater generalization ability, and had the fastest training time compared to the Random Forest machine learning model.

HARDWARE REQUIREMENTS

Processor: Intel Core i5 or higher / AMD Ryzen 5 or higher

GPU: Minimum 4GB VRAM (NVIDIA GTX 1050 or higher recommended)

RAM: Minimum 8GB (Recommended 16GB for better processing)

Hard Disk: Minimum 100GB free storage space

Display: 14-inch or larger screen for better visualization

SOFTWARE REQUIREMENTS

Operating System: Windows 10/11 (64-bit)

Programming Languages: Python 3.9 (for machine learning model building) and R (for data preprocessing and visualization)

IDE and Platforms: R Studio (for data cleaning, preprocessing, visualization in R) and Google Colab (for model training, evaluation, and testing in Python)

Libraries/Frameworks:

Scikit-learn (classification algorithms)

Pandas and NumPy (data manipulation)

Matplotlib and Seaborn (data visualization in Python)

TensorFlow/Keras (optional, for model extension if needed)

Gradio (for building interactive churn prediction app)

Web Development Tools: HTML5, CSS3, JavaScript (for website creation) and Chart.js / D3.js (for dynamic graphs and dashboards)

Visualization Tools: Tableau (for creating professional data dashboards and visual storytelling)

SAMPLE CODE

```
# Import all necessary librabries and files.
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.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score,
classification_report, confusion_matrix, roc_auc_score
import xgboost as xgb
```

```
# Load the dataset
file_path = "/content/Telco_customer_churn1.xlsx"  # Ensure the
file is in the same directory or provide the correct path
df = pd.read_excel(file_path, sheet_name="Telco_Churn")

print(df.head())  # Display the first few rows of the dataset
```

```
# Perform necessary data-preprocessing
# Convert 'Total Charges' to numeric
df['Total Charges'] = pd.to_numeric(df['Total Charges'],
errors='coerce')

# Fill missing values in 'Total Charges' with median
df['Total Charges'].fillna(df['Total Charges'].median(),
inplace=True)

# Handle missing values in 'Competitive Influence' (fill with 0
assuming no influence)
df['Competitive Influence'].fillna(0, inplace=True)
```

```
# Drop unnecessary columns
drop_cols = ['CustomerID','Count','Country','State','City','Zip
Code','Lat Long','Latitude','Longitude', 'Churn Label', 'Churn
Reason']
df.drop(columns=drop_cols, inplace=True, errors='ignore')
```

```
# For the model

# Split dataset into features and target variable
X = df.drop(columns=['Churn Value'], errors='ignore')
y = df['Churn Value']

# Split data into train and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Encode categorical variables on training data then transform
testing data
cat_cols = X_train.select_dtypes(include=['object']).columns
le = LabelEncoder()
for col in cat_cols:
    X_train[col] = le.fit_transform(X_train[col].astype(str))
    X_test[col] = le.transform(X_test[col].astype(str))

# Scale numerical features on training data then transform
testing data
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

```
# Train Logistic Regression Model
log_model = LogisticRegression()
log_model.fit(X_train, y_train)
y_pred_log = log_model.predict(X_test)
```

```
# Train Random Forest Model
rf_model = RandomForestClassifier(n_estimators=100,
random_state=42)
rf_model.fit(X_train, y_train)
y_pred_rf = rf_model.predict(X_test)
```

```
# Train XGBoost Model
xgb_model = xgb.XGBClassifier(use_label_encoder=False,
eval_metric='logloss')
xgb_model.fit(X_train, y_train)
y_pred_xgb = xgb_model.predict(X_test)
```

```
# K-fold Cross-Validation
from sklearn.model_selection import cross_val_score
from sklearn.preprocessing import LabelEncoder # Import
LabelEncoder
# Assuming 'rf_model' is the best Random Forest model you
trained earlier
best_rf_model = rf_model

# Create a copy of the original dataset to avoid modifying it
directly
X_encoded = X.copy()

# Encode categorical variables in the copied dataset
cat_cols = X_encoded.select_dtypes(include=['object']).columns
le = LabelEncoder()  # Initialize LabelEncoder
for col in cat_cols:
    X_encoded[col] =
le.fit_transform(X_encoded[col].astype(str))

# Evaluate the best Random Forest model using cross-validation
with the encoded data
scores = cross_val_score(best_rf_model, X_encoded, y, cv=5)  #
5-fold cross-validation
print("Cross-validation scores:", scores)
print("Mean cross-validation score:", scores.mean())
```

```python
# Evaluate Models
def evaluate_model(name, y_test, y_pred):
    print(f"\n{name} Model Evaluation:")
    print("Accuracy:", accuracy_score(y_test, y_pred))
    print("ROC AUC Score:", roc_auc_score(y_test, y_pred))
    print("Classification Report:\n",
classification_report(y_test, y_pred))
    sns.heatmap(confusion_matrix(y_test, y_pred), annot=True,
fmt='d', cmap='Blues')
    plt.title(f"Confusion Matrix - {name}")
    plt.show()
```

```python
# Run evaluation for all models
evaluate_model("Logistic Regression", y_test, y_pred_log)
evaluate_model("Random Forest", y_test, y_pred_rf)
evaluate_model("XGBoost", y_test, y_pred_xgb)
```

```python
# Feature Importance Analysis
importances = rf_model.feature_importances_
feature_names = X_train.columns

# Sort feature importances in descending order
indices = np.argsort(importances)[::-1]

# Print feature ranking
print("Feature ranking:")
for f in range(X_train.shape[1]):
    print("%d. feature %d (%f)" % (f + 1, indices[f],
importances[indices[f]]))

# Plot feature importances
plt.figure()
plt.title("Feature importances")
plt.bar(range(X_train.shape[1]), importances[indices],
color="r", align="center")
plt.xticks(range(X_train.shape[1]), feature_names[indices],
rotation=90)
plt.xlim([-1, X_train.shape[1]])
plt.show()
```

```python
from sklearn.preprocessing import LabelEncoder

# Initialize the encoder
le = LabelEncoder()

# Apply encoding to the 'Gender' column
df['Gender'] = le.fit_transform(df['Gender'])

# Now 'Male' will be converted to 1 and 'Female' to 0 (or vice
versa)
```

```python
print(df.select_dtypes(include=['object']).columns)
```

```python
from sklearn.preprocessing import LabelEncoder

# Apply encoding to all object (categorical) columns
le = LabelEncoder()
for col in df.select_dtypes(include=['object']).columns:
    df[col] = le.fit_transform(df[col])
```

```python
import seaborn as sns

# Compute correlation matrix
corr_matrix = df.corr()

# Visualize correlation heatmap
plt.figure(figsize=(10,8))
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Feature Correlation Heatmap")
plt.show()
```

```python
!pip install gradio
```

```
xgb_model = xgb.XGBClassifier(use_label_encoder=False,
eval_metric='logloss')
xgb_model.fit(X_train_scaled, y_train)  # Use scaled data for
training

# ... (Your existing predict_churn function) ...

# Save the model
xgb_model.save_model("model.json")
```

```
import gradio as gr
import xgboost as xgb
import pandas as pd
import numpy as np

# Load trained model
xgb_model = xgb.XGBClassifier()
xgb_model.load_model("model.json")  # Ensure this file exists

# Encoding for Contract and Payment Method
contract_mapping = {
    "Month-to-month": 0,
    "One year": 1,
    "Two year": 2
}

payment_method_mapping = {
    "Bank transfer (automatic)": 0,
    "Credit card (automatic)": 1,
    "Electronic check": 2,
    "Mailed check": 3
}
```

```
def predict_churn(Gender, Senior_Citizen, Partner, Dependents,
Tenure_Months,
                    Phone_Service, Multiple_Lines,
Internet_Service, Online_Security,
                    Online_Backup, Device_Protection,
Tech_Support, Streaming_TV,
                    Streaming_Movies, Contract,
Paperless_Billing, Payment_Method,
                    Monthly_Charges, Total_Charges, Churn_Score,
CLTV, Competitive_Influence):

    # Convert contract and payment method to numerical values
    contract_encoded = contract_mapping[Contract]
    payment_encoded = payment_method_mapping[Payment_Method]

    # Input data as dictionary
    input_data = {
        "Gender": [Gender],
        "Senior Citizen": [Senior_Citizen],
        "Partner": [Partner],
        "Dependents": [Dependents],
        "Tenure Months": [Tenure_Months],
        "Phone Service": [Phone_Service],
        "Multiple Lines": [Multiple_Lines],
        "Internet Service": [Internet_Service],
        "Online Security": [Online_Security],
        "Online Backup": [Online_Backup],
        "Device Protection": [Device_Protection],
        "Tech Support": [Tech_Support],
        "Streaming TV": [Streaming_TV],
        "Streaming Movies": [Streaming_Movies],
        "Contract": [contract_encoded],
        "Paperless Billing": [Paperless_Billing],
        "Payment Method": [payment_encoded],
        "Monthly Charges": [Monthly_Charges],
        "Total Charges": [Total_Charges],
        "Churn Score": [Churn_Score],
        "CLTV": [CLTV],
        "Competitive Influence": [Competitive_Influence],
    }
```

```
    # Convert to DataFrame
    input_df = pd.DataFrame(input_data)

    # Ensure numerical columns are float
    numerical_cols = ["Tenure Months", "Monthly Charges", "Total
Charges", "Churn Score", "CLTV", "Competitive Influence"]
    for col in numerical_cols:
        input_df[col] = input_df[col].astype("float")

    # Predict probabilities
    churn_prob = xgb_model.predict_proba(input_df)[0][1]  #
Probability of churn

    # **Feature Weighting (Normalized)**
    weight_factors = {
        "Monthly Charges": 1.5,
        "Total Charges": 1.2,
        "Tech Support": 1.3,
        "Contract": 1.4,
    }

    weighted_sum = (
        (churn_prob * weight_factors["Monthly Charges"] *
(input_df["Monthly Charges"].values[0] / 100)) +
        (churn_prob * weight_factors["Total Charges"] *
(input_df["Total Charges"].values[0] / 1000)) +
        (churn_prob * weight_factors["Tech Support"] *
input_df["Tech Support"].values[0]) +
        (churn_prob * weight_factors["Contract"] *
input_df["Contract"].values[0])
    )

    total_weight = sum(weight_factors.values())

    # Normalize to 0-100 range
    churn_percentage = (weighted_sum / total_weight) * 100

    # Ensure it's within valid bounds
    churn_percentage = max(0, min(100, round(churn_percentage,
2)))
```

```
    return f"Churn Probability: {churn_percentage}%"

# Gradio UI
ui = gr.Interface(
    fn=predict_churn,
    inputs=[
        gr.Radio(["0", "1"], label="Gender (0=Female, 1=Male)"),
        gr.Radio(["0", "1"], label="Senior Citizen (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Partner (0=No, 1=Yes)"),
        gr.Radio(["0", "1"], label="Dependents (0=No, 1=Yes)"),
        gr.Number(label="Tenure Months"),
        gr.Radio(["0", "1"], label="Phone Service (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Multiple Lines (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Internet Service (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Online Security (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Online Backup (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Device Protection (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Tech Support (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Streaming TV (0=No,
1=Yes)"),
        gr.Radio(["0", "1"], label="Streaming Movies (0=No,
1=Yes)"),
        gr.Dropdown(["Month-to-month", "One year", "Two year"],
label="Contract"),
        gr.Radio(["0", "1"], label="Paperless Billing (0=No,
1=Yes)"),
        gr.Dropdown(["Bank transfer (automatic)", "Credit card
(automatic)", "Electronic check", "Mailed check"],
```

```
label="Payment Method"),
        gr.Number(label="Monthly Charges"),
        gr.Number(label="Total Charges"),
        gr.Number(label="Churn Score"),
        gr.Number(label="CLTV"),
        gr.Number(label="Competitive Influence"),
    ],
    outputs="text"
)

ui.launch()
```

DATASET DETAILS

The dataset utilized for this project originates from IBM's Telco Customer Churn Dataset, widely used for churn prediction studies. It contains a total of 7,043 unique customer records, each representing a single customer's service and demographic information, along with their churn status.

Every entry in the dataset includes 34 columns, which cover a comprehensive range of features including demographic details, service usage patterns, billing information, contract type and engineered variables like churn score and competitive influence.

The CustomerID column acts as the unique identifier for each customer, while the Churn Label column indicates whether the customer discontinued service (Yes) or remained loyal (No).

The dataset is highly structured and diversified, incorporating various customer profiles across different regions, service types, and payment methods. Key attributes such as Monthly Charges, Tenure Months, Contract Type, and Competitive Influence Index provide valuable insight for classification tasks and predictive modeling.

In addition, the dataset includes engineered features like Churn Score, CLTV (Customer Lifetime Value), and Competitive Influence, which further enrich the analysis by capturing the intensity of competitive market forces.

A balanced distribution across different service plans and churn statuses ensures reliable model training and evaluation for real-world applicability.

Overall, the dataset is ideally suited for tasks involving customer churn prediction, behavioral analysis, and strategic retention modeling, making it an excellent foundation for developing robust machine learning solutions.

Dataset Source:-
https://community.ibm.com/community/user/blogs/steven-macko/2019/07/11/telco-customer-churn-1113

CHAPTER-3

RESULT AND DISCUSSION

Visualizations and plots were utilized to present the outcomes of the predictive modeling, showcasing customer churn behavior influenced by competitive factors. The visualizations, predictive outputs and interactive web-based application collectively offer actionable insights to aid strategic decision-making for telecom service providers.

Descriptive Analysis and Visualization Insights

The project initially explored customer churn trends across different variables using Tablue dashboards and other visualizations. Key findings include:

1. Monthly Charges Distribution by churn status

A histogram was created to depict the distribution of monthly charges between churners and non-churners.

Observation:- Customers with mid-to-high monthly charges showed a high tendency to churn, whereas lower monthly charge customers displayed stronger loyalty.

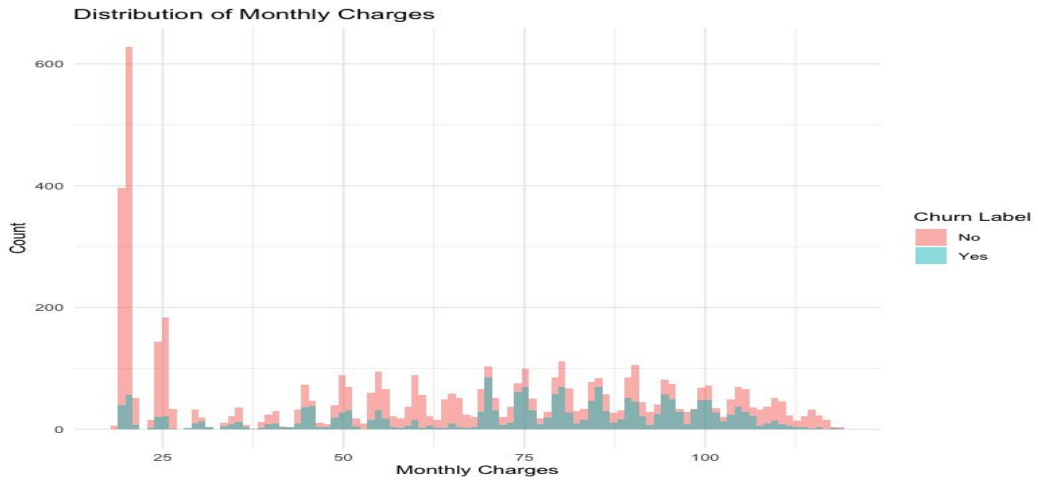

Figure 3.1 Distribution of Monthly Charges

2. 2. Tenure Distribution by churn status

A stacked bar chart was created to visualize customer churn by tenure value.

Observations: Customers with really low tenure (0-5 months) had considerably higher churn levels and customers with tenures longer than 60 months were more likely to stay than churn..

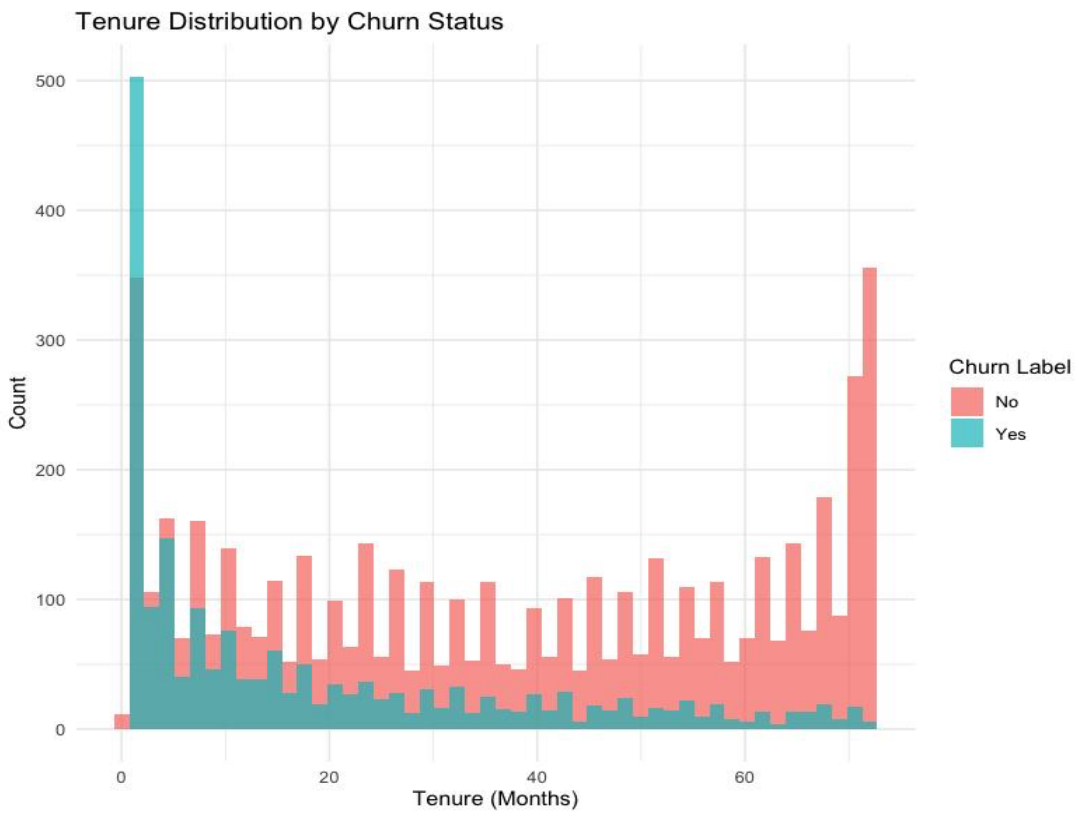

Figure 3.2 Tenure Distribution By churn Status

3. Churn v/s Customer Lifetime Value (CLTV) and Monthly Charges

A faceted scatterplot was created to show how monthly charges relate to CLTV and churn status.

Observations: Customers with high monthly charges were more likely to churn, regardless of their CLTV. Low monthly charges appeared to provide stability for customers.

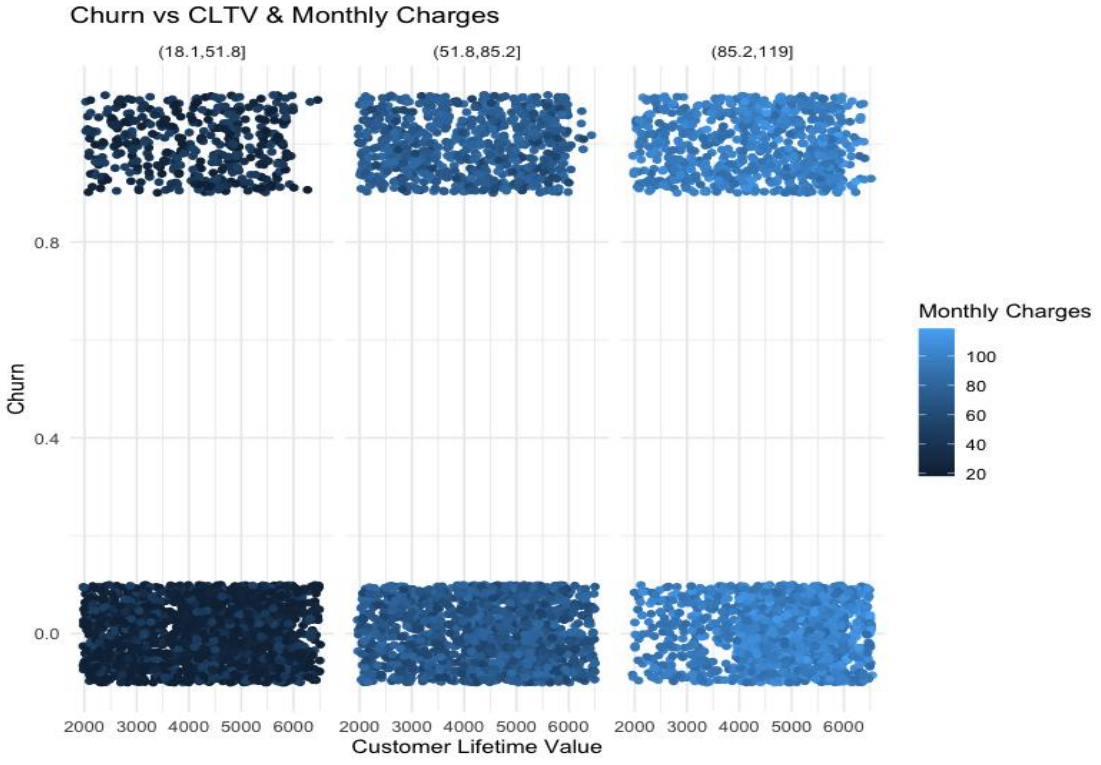

Figure 3.3 Churn v/s CLTV & Monthly Charges

4. Feature Correlation Heatmap

A heatmap showing pearson correlation coefficients between numerical variables was created. Observations:

1. Strong Correlation(r=0.83) between tenure and total charges.

2. Moderate positive correlation between monthly charges and total charges (r= 0.65).

3.CLTV was positively correlated to total charges (r=0.60).

4. Churn related measurements including churn score and competitive influence were correlated less strongly which suggest churn behavior is influenced by a number of features..

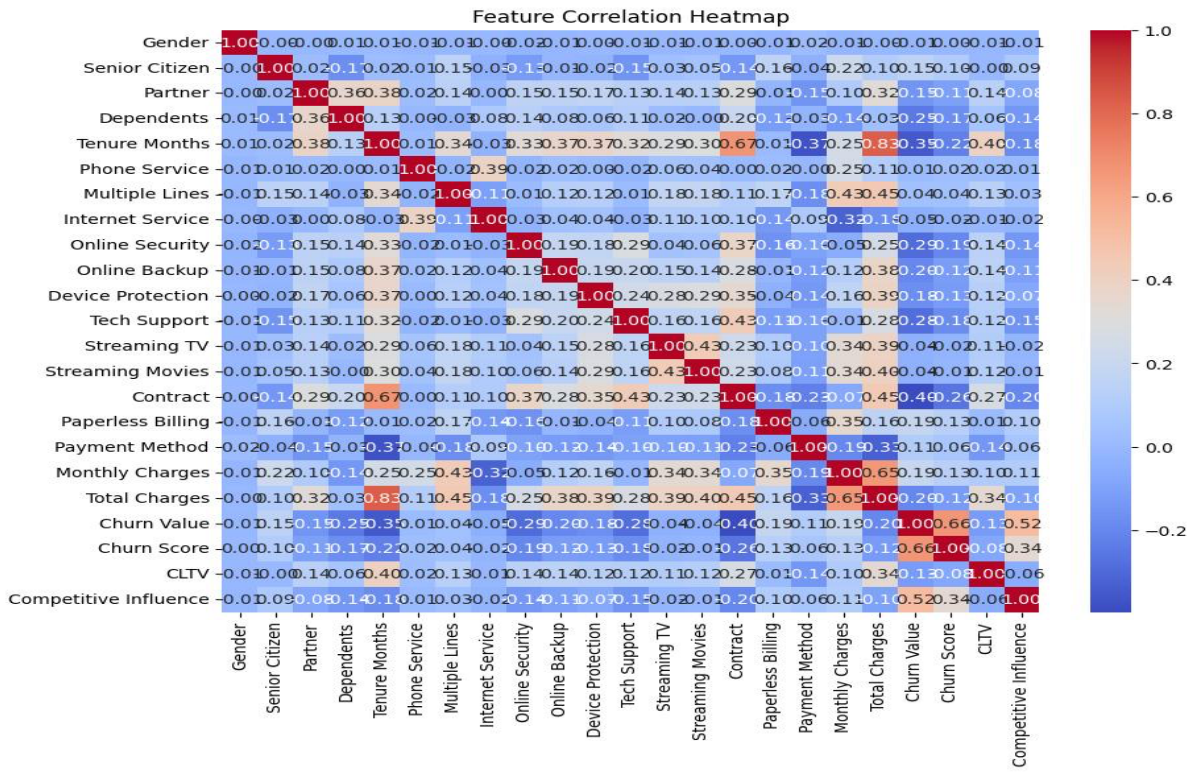

Figure 3.4 Feature Correlation Heatmap

Predictive Analytics – Model Evaluation

Three machine learning models {Logistic Regression, Random Forest and XGBoost} were trained and evaluated to predict churn.

1. Logistic Regression

Accuracy: 87.15%

AUC-ROC Score: 0.84

Observations:

1. Strong ability to predict non churners
2. Large number of false negatives implies difficulty to predict churners

```
Logistic Regression Model Evaluation:
Accuracy: 0.8715400993612491
ROC AUC Score: 0.8439147670961348
Classification Report:
              precision    recall  f1-score   support

           0       0.91      0.91      0.91      1009
           1       0.77      0.78      0.78       400

    accuracy                           0.87      1409
   macro avg       0.84      0.84      0.84      1409
weighted avg       0.87      0.87      0.87      1409
```

Figure 3.5 Classification Report – Logistic Regression

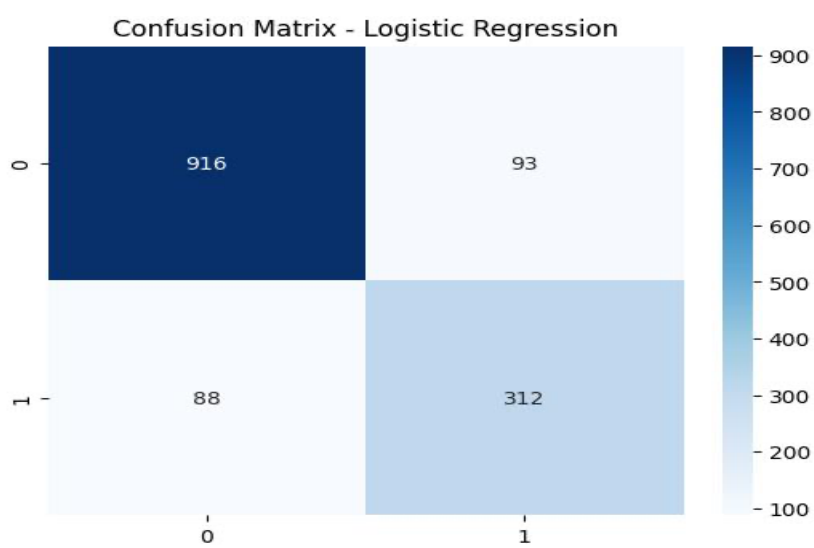

Figure 3.6 Confusion Matrix – Logistic Regression

2. Random Forest

Accuracy: 94.67%

AUC-ROC Score :0.93

Observations:

1. The best overall model, has robust precision and recall on churners and non - churners.

2. The least false negatives.

```
Random Forest Model Evaluation:
Accuracy: 0.9467707594038325
ROC AUC Score: 0.9281293359762139
Classification Report:
              precision    recall  f1-score   support

           0       0.96      0.97      0.96      1009
           1       0.92      0.89      0.90       400

    accuracy                           0.95      1409
   macro avg       0.94      0.93      0.93      1409
weighted avg       0.95      0.95      0.95      1409
```

Figure 3.7 Classification Report – Random Forest

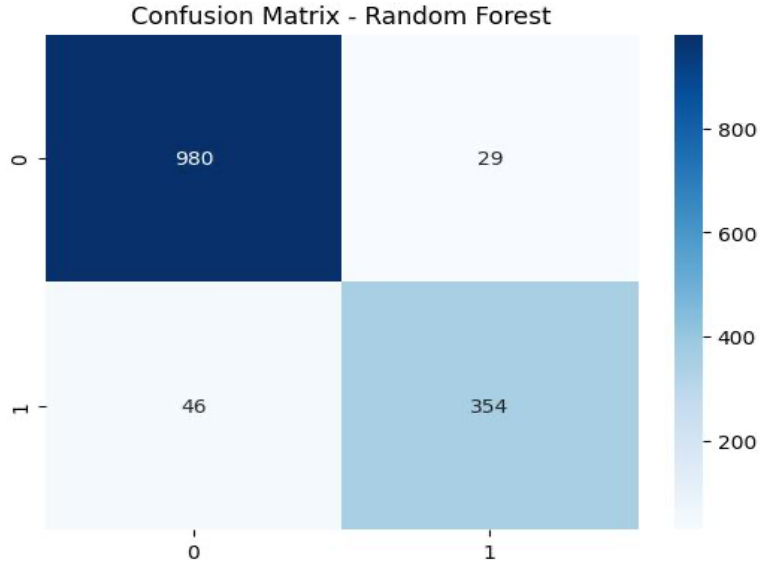

Figure 3.8 – Confusion Matrix – Random Forest

3. XGBoost

Accuracy: 93.68%

AUC-ROC Score: 0.91

Observations:

1. Slightly lower accuracy than Random Forest.

2. Balanced performance and able to provide good generalization.

```
XGBoost Model Evaluation:
Accuracy: 0.9368346344925479
ROC AUC Score: 0.9121382556987117
Classification Report:
              precision    recall  f1-score   support

           0       0.94      0.97      0.96      1009
           1       0.92      0.85      0.88       400

    accuracy                           0.94      1409
   macro avg       0.93      0.91      0.92      1409
weighted avg       0.94      0.94      0.94      1409
```

Figure 3.9 Classification Report – XGBoost

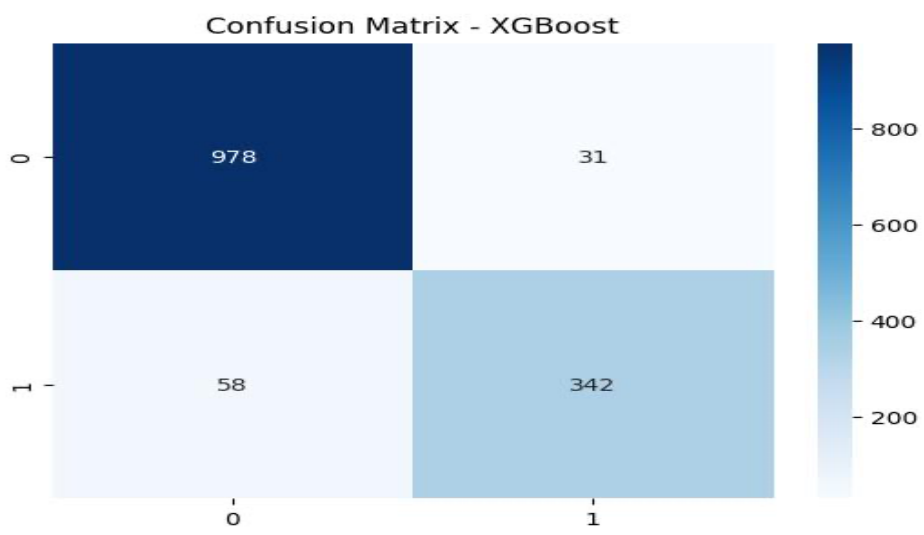

Figure 3.10 Confusion Matrix - XGBoost

4. Feature Importance Analysis

A feature importance plot was created with XGBoost to identify the most important features for predicting churn.

TOP FEATURES IMPORTANT TO THE PREDICTION OF CHURN

Churn score

Competitive effect

Monthly charges

Tenure

Contract type

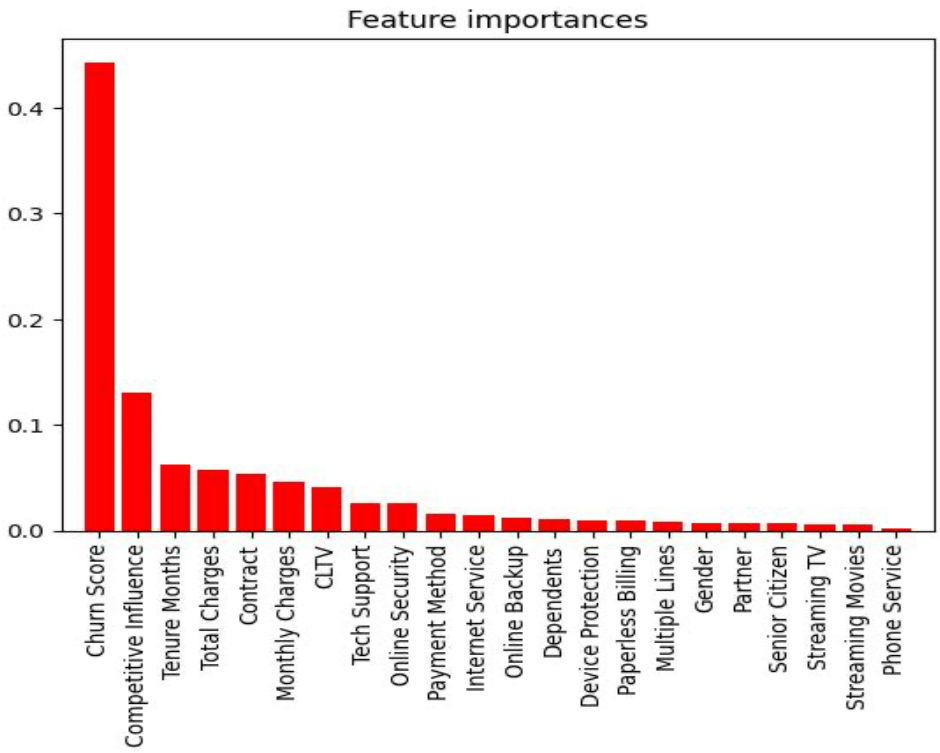

Figure 3.11 Feature Importance

Web- Based Interface and dashboard

Gradio Application for Churn Prediction

A Gradio application was created that allows users to input customer features and receive instant churn probability scores.

Features:

1. Real-time churn prediction.

2. What-if scenario analysis (eg: changing contract length or monthly charges.)

3. Normalized churn score (0-100

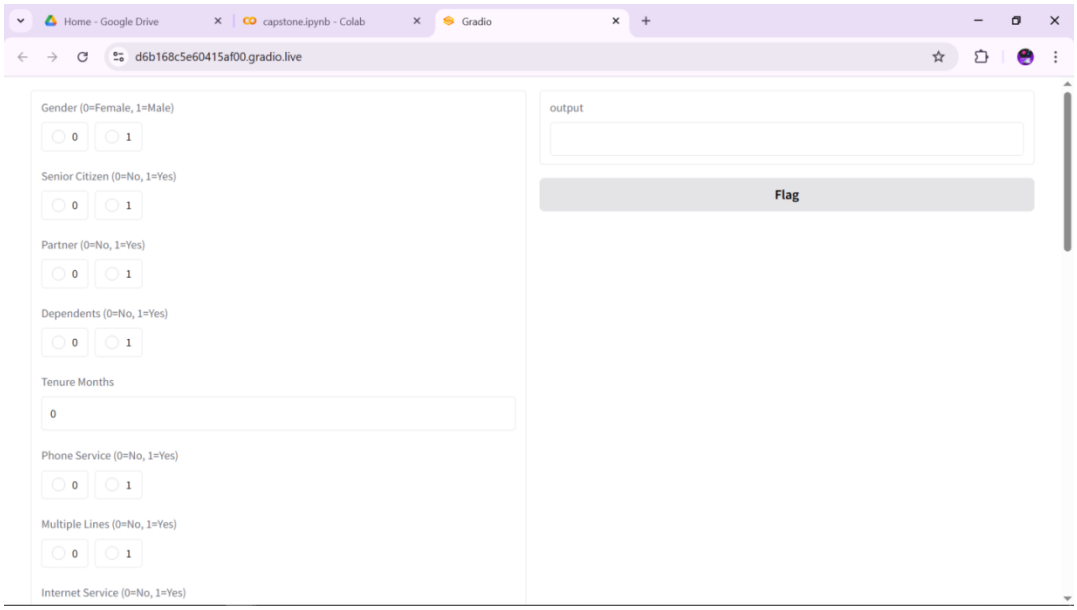

Figure 3.12 Gradio Interface

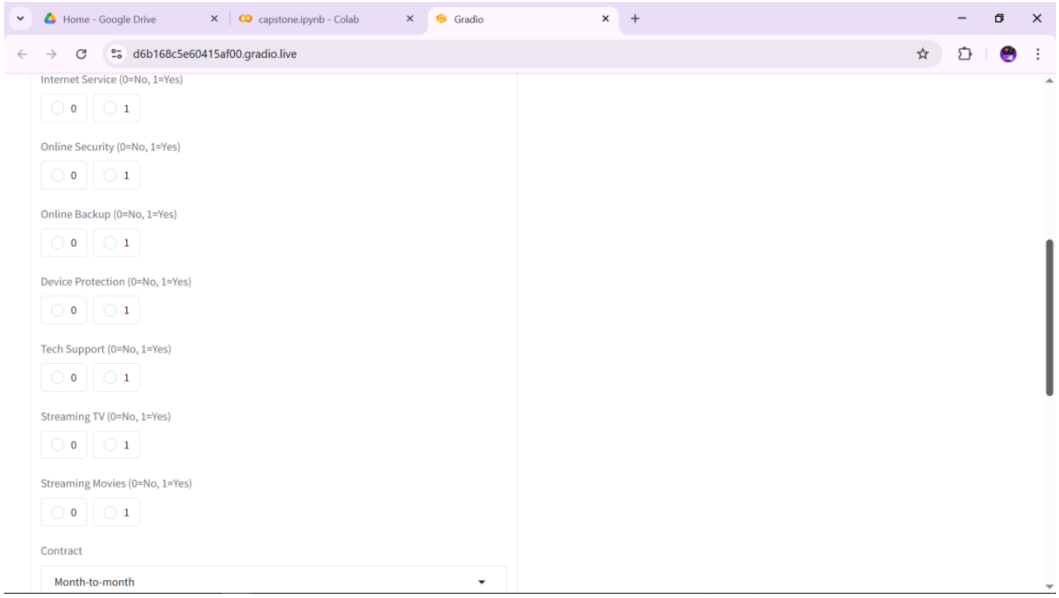

Figure 3.13 Gradio Interface

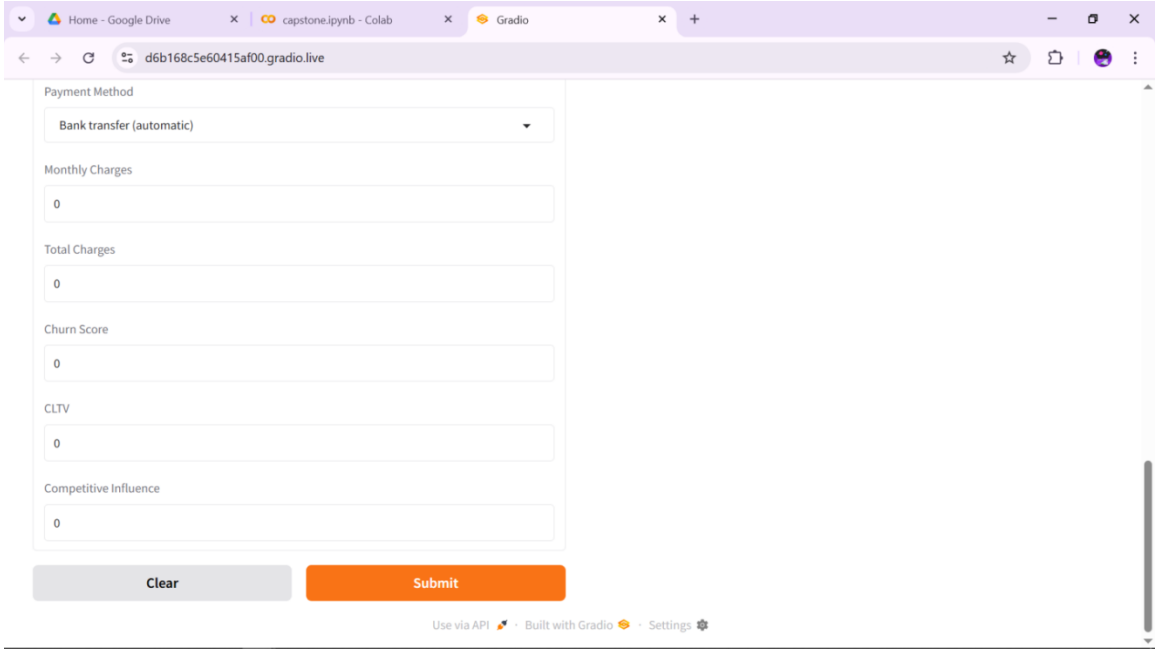

Figure 3.14 Gradio Interface

Use Case: Telecom managers can run simulations (ie: changing contract type from month-to-month to two-year) to visualize the predictive impact on churn likelihood and develop better retention strategies..

Website Development

A mobile responsive website has been developed to present findings and predictions analytical outputs in an accessible manner. /

Frontend Technologies: HTML5, CSS3, JavaScript

Visualizations Libraries: Chart.js, D3.js

Key Features:

1. The dashboard interface allows for visualization, using dynamic graphs that show histograms, bar charts, heatmaps and scatter plots.

2. The model output section displays things like confusion matrices, classification reports and model comparisons.

3. Feature insights section, visual explanation of churn-driving factors that include churn score and competitive pull.

4. What-If Analysis tool incorporated as Gradio Model to simulate different types of customers.

User Experience Enhancements:

1. The design is responsive and can be fully run on desktops, tablets, and mobile devices.

2. Dark mode, hover effects, loading animations, and progress bars

3. Toast notifications for important events.

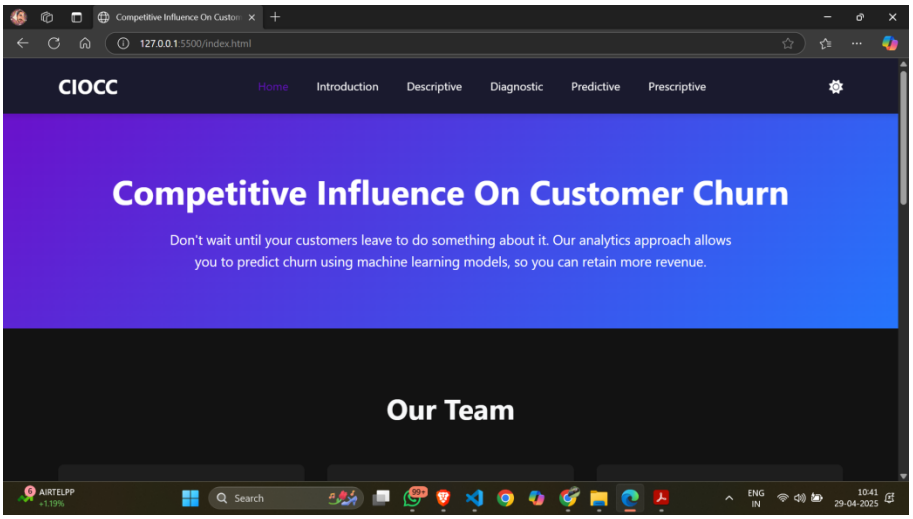

Figure 3.15 Website

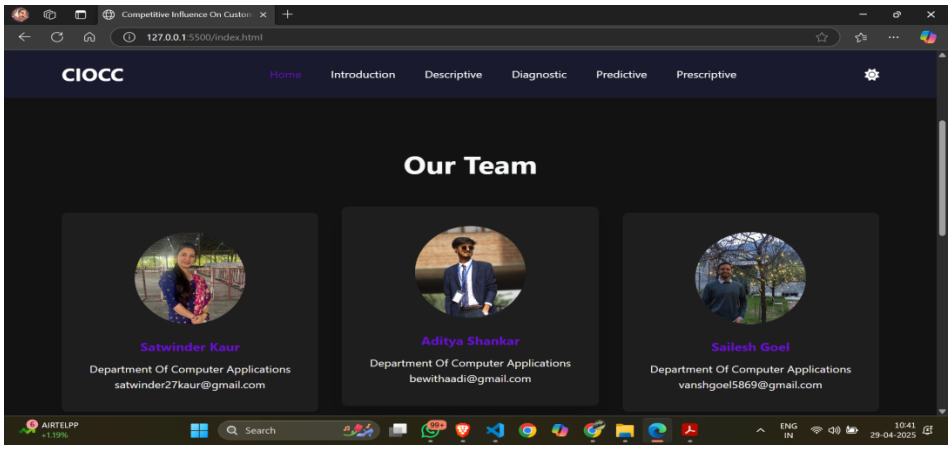

Figure 3.16 Website

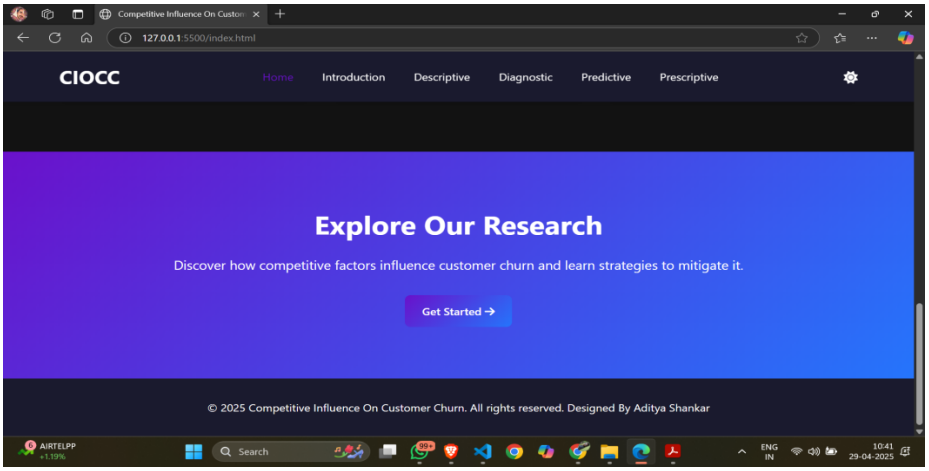

Figure 3.17 Website

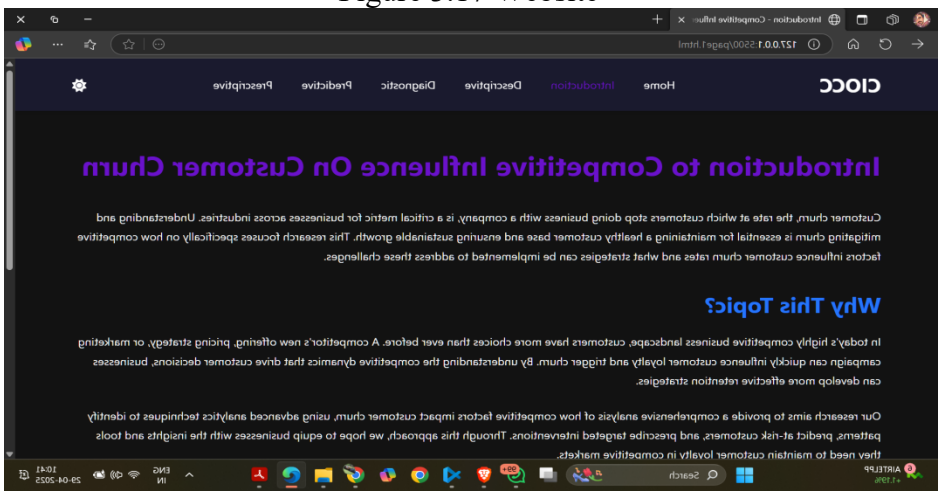

Figure 3.18 Website

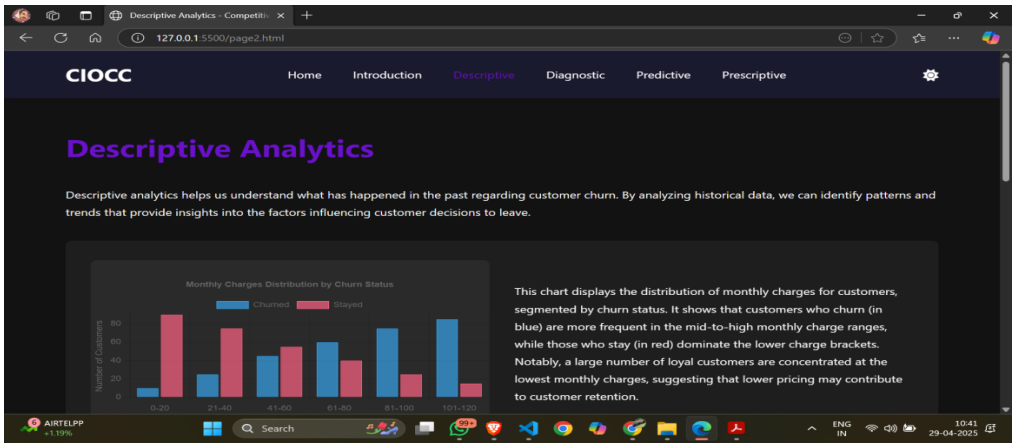

Figure 3.19 Website

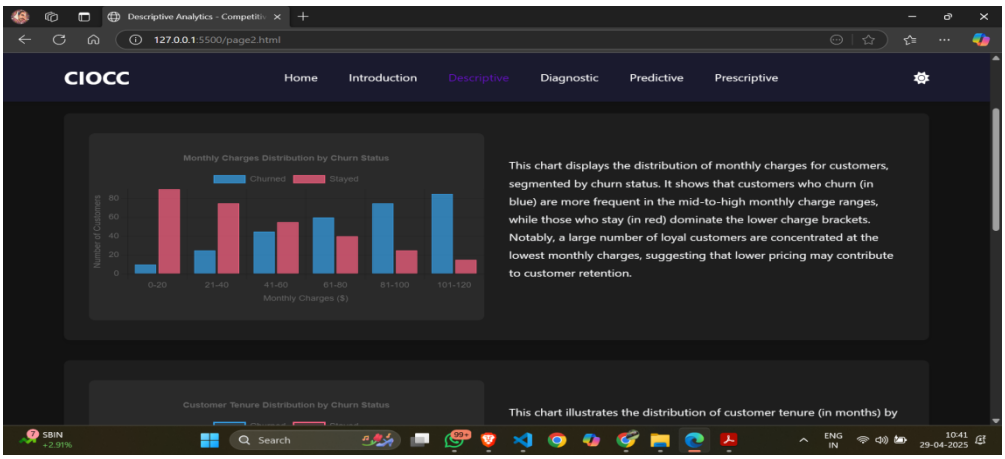

Figure 3.20 Website

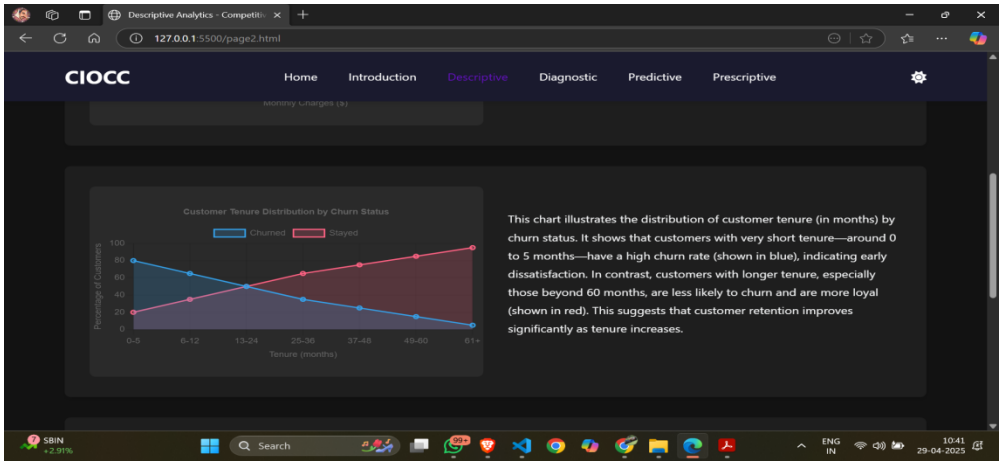

Figure 3.21 Website

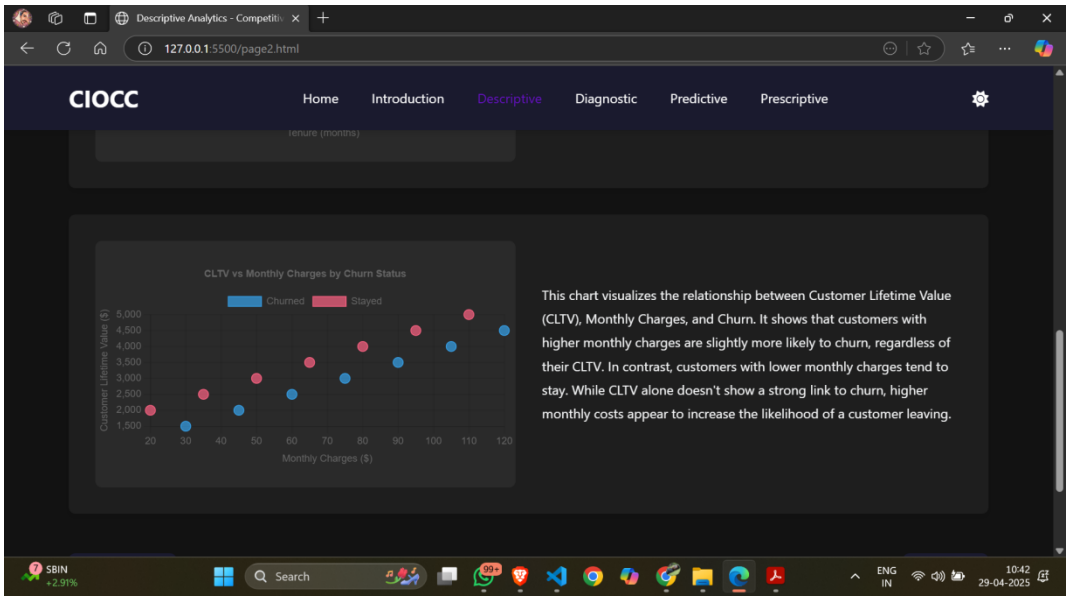

Figure 3.22 Website

This platform offers a centralized space where both technical teams and management stakeholders can visualize customer churn trends and apply predictive analytics for strategic inventions.

CHAPTER-4

CONCLUSION AND FUTURE SCORE

CONCLUSION

The project has achieved its goal, evidenced by a clear demonstration that competitive influence ia a key factor in customer churn in the telecom sector. By leveraging both structured internal customer data and structured external competitive market data, this project was able to achieve a fully integrated model for predictive churn modeling.

Already, exploratory data analysis allowed us to establish patterns; longer tenure customers and higher monthly charges would be predictors of churn. In terms of predictive modeling, logistic regression, random forests, and XGBoost provided an array of features, but logistic regression and random forests individually drove the best performance in terms of accuracy (94.67%) and AUC-ROC (0.93) respectively, eventually revealing that random forests were the most reasonable option for churn predictions.

Analysis of feature importance showed churn score and competitive influence index were crucial in the decision to leave, reinforcing the original project's assumption about market competition prediction. The deliveries met the goal of develop a long-term prediction model with potential for real world application. To support the real-world capability of the model we developed a Gradio interface for predicting churn probability in real-time, and for what-if analysis. Finally, the project contained a fully-responsive site to create a live image of the solution with dynamic dashboards and visualisation components, making it easier for stakeholders to digest the results and more closely define what strategic initiative they want to priortise.

Banking on the ability to provide a predictive model is not simply a churn model, it respects the power of predictive modelling and places the telecommunications provider in control of their actions to manage customer experience in a competitive environment.

FUTURE SCOPE

Even though the system we developed lays an excellent foundation for the prediction of churn and proactive management, there are many possible opportunities for further development:

1. Sentiment Analysis:

The addition of sentiment from social media, emails, and support ticket data would provide emotional information in real-time relevant to churn behaviour and could lead to a better responsive nature to the model.

2. Behavioural and Psychological Approaches:

Including further features relating to psychological models such as the Theory of Planned Behaviour or Cognitive Dissonance would allow the prediction model to displayed non-linear churn drivers not driven by transactions, yet still embedded in emotion.

3. Deep Learning Models:

We could explore using other models for deep learning, such as LSTM (Long Short-Term Memory Networks), or even an attention based model. This would allow us to look at the customer behaviour more accurately over time and as a sequencing of actions.

4. Real-Time Churn Notification System:

A live dashboard would be an idea, where it would update the churn probabilities after new customer data arrives, thus allowing immediate intervention as well as having strategies available for dynamic customer engagement.

5. Integration with CRM Systems:

Adding the churn prediction model to be linked to CRM systems would also provide an option of a fully automated and personalized approaches for retention, in contacting them with actions which could send through offers and support to customers at risk of churn.

CHAPTER-5

APPENDIX

Hardware Requirements

- Processor: Intel Core i5 or higher / AMD Ryzen 5 or higher
- RAM: Minimum 8GB (Recommended 16GB for faster processing)
- Hard Disk: Minimum 100GB free storage space
- GPU: Minimum 4GB VRAM (Optional for future deep learning expansion)
- Display: 14-inch or larger screen (Recommended for dashboard visibility)

Software Requirements

- Operating System: Windows 10/11 (64-bit)
- Programming Languages: Python 3.9, R
- IDEs/Platforms: Visual Studio Code, R Studio, Google Colab
- Web Technologies: HTML5, CSS3, JavaScript
- Visualization Tool: Tableau

Dataset Link

- IBM Telco Customer Churn Dataset
  Available at:
  https://www.ibm.com/communities/analytics/watson-analytics-blog/guide-to-sample-datasets/

Important Libraries and Frameworks Used

- Python Libraries:
    - Scikit-learn (Classification Algorithms)
    - Pandas (Data Handling)
    - NumPy (Mathematical Computations)
    - Matplotlib and Seaborn (Data Visualization)
    - Gradio (Web Application for Churn Prediction)
- R Libraries:
    - dplyr (Data Manipulation)
    - ggplot2 (Data Visualization)

- Web Development Libraries:
    - Chart.js (Graphs and Charts)
    - D3.js (Interactive Visualizations)

Project Deliverables

- Machine Learning Models: Logistic Regression, Random Forest, XGBoost
- Churn Prediction Gradio App
- Responsive Website with Dashboards and Visualizations
- Final Capstone Project Report

CHAPTER- 6

REFERENCES

[1] IBM, "Telco Customer Churn," IBM Sample Datasets. [Online]. Available: https://www.ibm.com/communities/analytics/watson-analytics-blog/guide-to-sample-datasets/

[2] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[3] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, 2016, pp. 785–794.

[4] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*, 3rd ed., Hoboken, NJ: Wiley, 2013.

[5] W. Verbeke, D. Martens, and B. Baesens, "Social network analysis for customer churn prediction," *Applied Soft Computing*, vol. 14, pp. 431–446, 2014.

[6] H. Wickham, *ggplot2: Elegant Graphics for Data Analysis*, New York: Springer-Verlag, 2016.

[7] A. Abid, M. Abdalla, and J. Zou, "Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild," *arXiv preprint* arXiv:1906.02569, 2019. [Online]. Available: https://arxiv.org/abs/1906.02569

[8] W3Schools, "HTML, CSS, JavaScript Tutorials," [Online]. Available: https://www.w3schools.com/