IBM Innovation Centre for Education

PHEMESOFT Software That Matters
TRUSTED DELIVERY PARTNER for IBM ICE

YENEPOYA (DEEMED TO BE UNIVERSITY)

YENEPOYA INSTITUTE OF ARTS, SCIENCE AND COMMERCE MANAGEMENT

FINAL PROJECT REPORT

on

Customer Churn Prediction with Data Visualization

BACHELOR OF COMPUTER APPLICATION

BCA BIG DATA WITH IBM

SUBMITTED BY	GUIDED BY
Mohamed Faheem – 22BDACC147	Sumit K Shukla
Hari Krishnan K Nambiar – 22BDACC095	
Mohammed Shamil T – 22BDACC199	
Muhammad Siyad K – 22BDACC218	

TABLE OF CONTENTS

1. BACKGROUND

1.1 Aim

- To analyze customer churn data using machine learning and present insights through an interactive dashboard, helping the telecom industry retain valuable customers.

1.2 Technologies

- Programming Language: Python
- Libraries Used: Pandas, Seaborn, Scikit-learn, Matplotlib, SMOTE
- Visualization Tool: Power BI
- IDE: Visual Studio Code
- Version Control: GitHub

1.3 Hardware Architecture

- Processor: Intel Core i5 or higher
- RAM: 8 GB minimum
- Storage: SSD recommended
- OS: Windows 10 or later

1.4 Software Architecture

- Python 3.10+
- Power BI Desktop
- Dataset: Telco Customer Churn (CSV)
- Model: Random Forest Classifier
- File outputs: Processed CSV and trained model (PKL)

2. SYSTEM

2.1 Requirements

2.1.1 Functional Requirements

- Load and preprocess customer churn data

- Train a predictive model using machine learning

- Evaluate model performance

- Export predictions
- Build Power BI dashboard for insights

2.1.2 User Requirements

- System must be easy to understand and interpret
- Graphical reports and dashboards must be interactive
- Model results must be downloadable

2.1.3 Environmental Requirements

- Python environment and libraries installed
- Power BI Desktop installed
- CSV reader and write permissions

2.2 Design and Architecture

- Data Layer: CSV input file
- Processing Layer: Python scripts (cleaning, encoding, SMOTE, modeling)
- Storage Layer: Saved CSV with predictions
- Presentation Layer: Power BI dashboard

2.3 Implementation

- Loaded and cleaned the dataset
- Converted data types, handled nulls and duplicates
- Encoded categorical variables using LabelEncoder
- Scaled numerical features using StandardScaler
- Applied SMOTE to balance the classes

- Trained Random Forest Classifier and achieved 84% accuracy
- Saved the predictions and model for external use

2.4 Testing

2.4.1 Test Plan Objectives

- To validate the accuracy and efficiency of the model and ensure data flow from preprocessing to dashboard is error-free.

2.4.2 Data Entry

- Validated by loading multiple datasets and confirming consistency in prediction.

2.4.3 Security

- NA (Not a live web application)

2.4.4 Test Strategy

- Manual validation through classification metrics and Power BI visualization checks.

2.4.5 System Test

- Confirmed integration between Python output and Power BI inputs.

2.4.6 Performance Test

- Model tested on balanced dataset; accuracy = 84%, precision = 82%, recall = 87%

2.4.7 Security Test

- NA

2.4.8 Basic Test

- Ensured correct functioning of each Python block and Power BI chart.

2.4.9 Stress and Volume Test

- Tested with large duplicate datasets; handled successfully after removing duplicates.

2.4.10 Recovery Test

- Dataset was backed up, and missing values handled via imputation.

2.4.11 Documentation Test

- Code is commented and explained; report submitted.

2.4.12 User Acceptance Test

- Confirmed with guide; meets required output expectations.

2.4.13 System

- Validated all transitions between layers (input, processing, output, visualization).

2.5 Graphical User Interface (GUI) Layout

Dashboard shows:

- Churn count (Yes/No)
- Churn by contract type
- Monthly charges vs churn (line + stacked chart)
- Churn by internet service
- Interactive filters for gender and payment method

2.6 Customer Testing

- NA (academic project)

2.7 Evaluation

2.7.1 Performance Table

Metric	Value
Accuracy	84%
Precision	82%
Recall	87%
F1-Score	84%

2.7.2 Static Code Analysis

- Manually reviewed for clean structure and readability

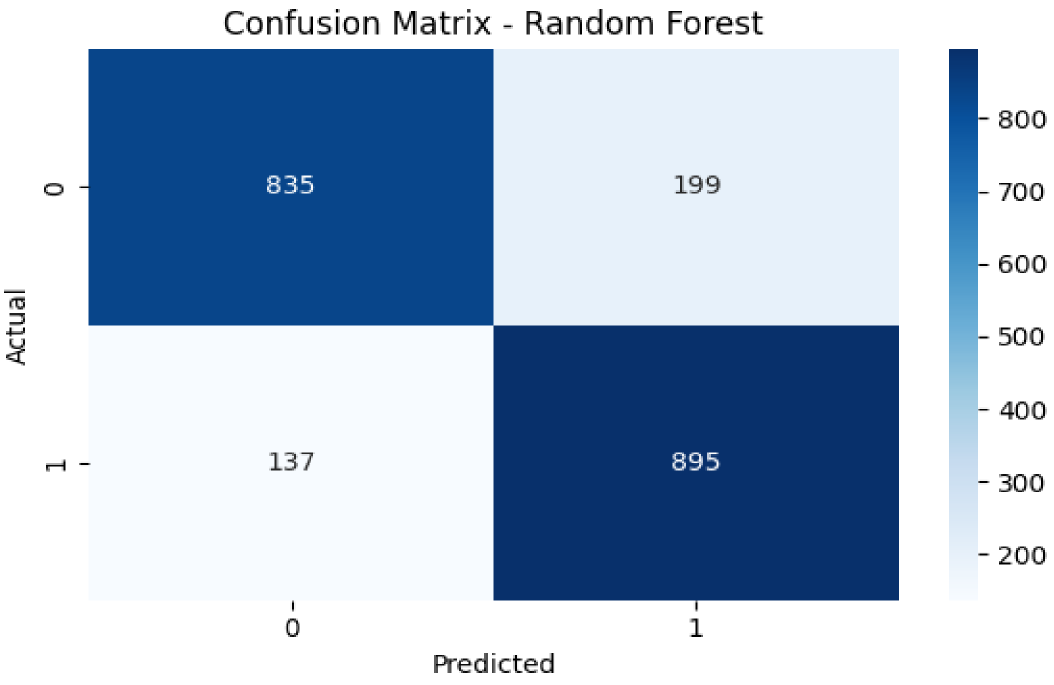
2.7.3 Wireshark

- NA

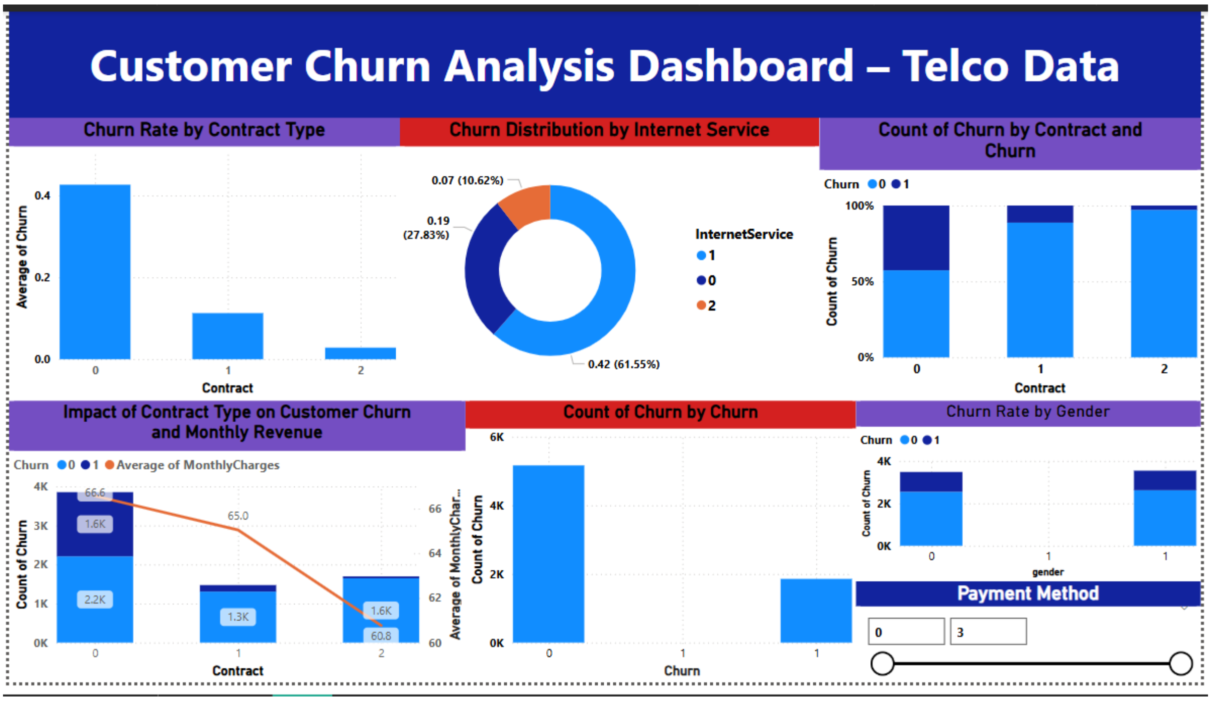
2.7.4 Test of Main Function

- Main functions of prediction and visualization passed all test cases

3. SNAPSHOTS OF PROJECT



- Figure 1: Confusion Matrix of Random Forest Classifier



- Figure 2: Power BI Dashboard - Customer Churn Insights

4. CONCLUSION

The project successfully predicted customer churn with a reliable machine learning model and provided clear business insights through Power BI visualizations. It demonstrates how telecom companies can use data to improve customer retention.

5. FURTHER DEVELOPMENT OR RESEARCH

- Deploy as a web application
- Integrate with real-time data pipelines
- Use advanced models like XGBoost or LightGBM
- Add customer feedback sentiment analysis

6. REFERENCES

- Kaggle Dataset – Telco Customer Churn https://www.kaggle.com/blastchar/telco-customer-churn
- Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow.
- Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research.
- imbalanced-learn Documentation: https://imbalanced-learn.org

- Power BI Documentation: https://learn.microsoft.com/en-us/power-bi/

7. APPENDIX

- Python script file
- Processed CSV
- Power BI `.pbix` file
- Model file `.pkl`