AIU.
ALAMEIN
INTERNATIONAL UNIVERSITY
جامعـة العلميـن الدوليـة

Customer Churn Prediction Using AI: Project Report

Name: Mohamed Ahmed Mohamed Reda

ID: 21100914

Project Name: Customer Churn Prediction

Institution: National Bank of Egypt

Table of Contents:

Introduction:

In today's competitive banking industry, retaining customers is crucial for sustained growth and profitability. Customer churn, the phenomenon where customers terminate their relationship with a business, poses a significant challenge for banks. Churn can manifest in two ways:

Customer churn: when customers leave the bank entirely.

Product churn: when customers stop using one product or service while retaining others with the bank.

Our project aims to address this issue by developing a machine learning model to predict customer churn within the banking sector. Early identification of customers likely to churn allows for proactive measures, helping the bank retain customers.

The project uses 2 datasets from Kaggle titled “Customer-Churn-Records” and “Churn_Modelling”, containing various customer attributes, such as demographic details and account information. We analyze these features using data preprocessing, exploratory data analysis (EDA), and machine learning techniques to build a model that accurately predicts the likelihood of churn. By understanding key trends and patterns, the model can identify customers at risk, enabling the bank to focus on retention efforts effectively.

3. Requirements

3.1 Functional Requirements

Functional requirements specify what the system must do to meet its goals. In this project, the following functional requirements are essential:

1. **User Authentication:** The system must allow users to register, log in, and log out securely using unique credentials.
2. **Data Input:** Users should be able to input data through a user-friendly interface, including uploading customer records for churn prediction.
3. **Data Processing:** The system must process the input data to generate predictions on customer churn, aiming for at least 75% accuracy.
4. **Reporting:** The system should generate reports based on the processed data, including visualizations of churn trends, and can be exported in various formats, such as PDF or Excel.
5. **Notifications:** Users should receive notifications for key events, such as when a churn prediction is made or when a report is generated.
6. **User Management:** Admins must have the ability to manage user roles and permissions, ensuring secure access and control over the system.

3.2 Non-Functional Requirements

Non-functional requirements define the system's operational capabilities and constraints. For this project, they include:

1. **Performance:** The system should be capable of handling up to 10,000 concurrent users without performance degradation.
2. **Security:** All data must be encrypted both in transit and at rest. The system should comply with Egypt's Personal Data Protection Law (PDPL) to safeguard customer data and ensure proper data handling practices.
3. **Reliability:** The system should have an uptime of 99.9% and must be able to recover from failures within 5 minutes to ensure uninterrupted service.
4. **Scalability:** The system should be scalable to accommodate future increases in the number of users and data volume, particularly as more customer records are added.

5. **Usability:** The user interface must be intuitive and easy to navigate, allowing users to complete any task within a maximum of 3 clicks.
6. **Maintainability:** The system must be modular, with well-documented code, making it easier to maintain, debug, and update as needed.
7. **Compatibility:** The system should be compatible with major web browsers (e.g., Chrome, Firefox, Safari) and operating systems (e.g., Windows, macOS, Linux).

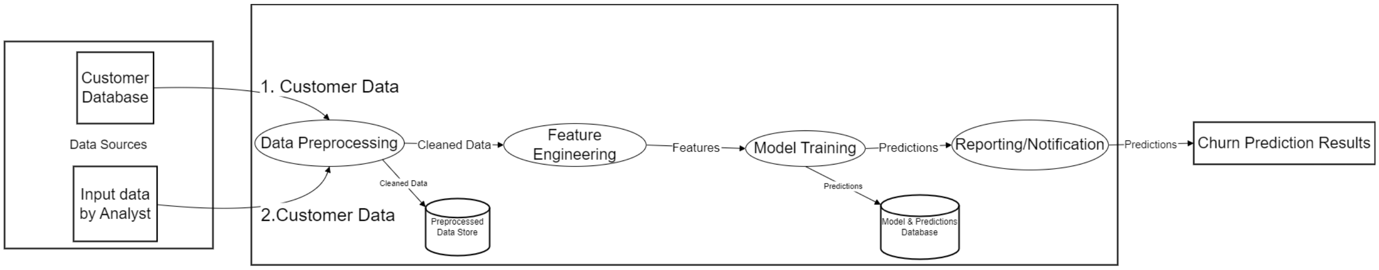
4. Design and Architecture

4.1 System Architecture

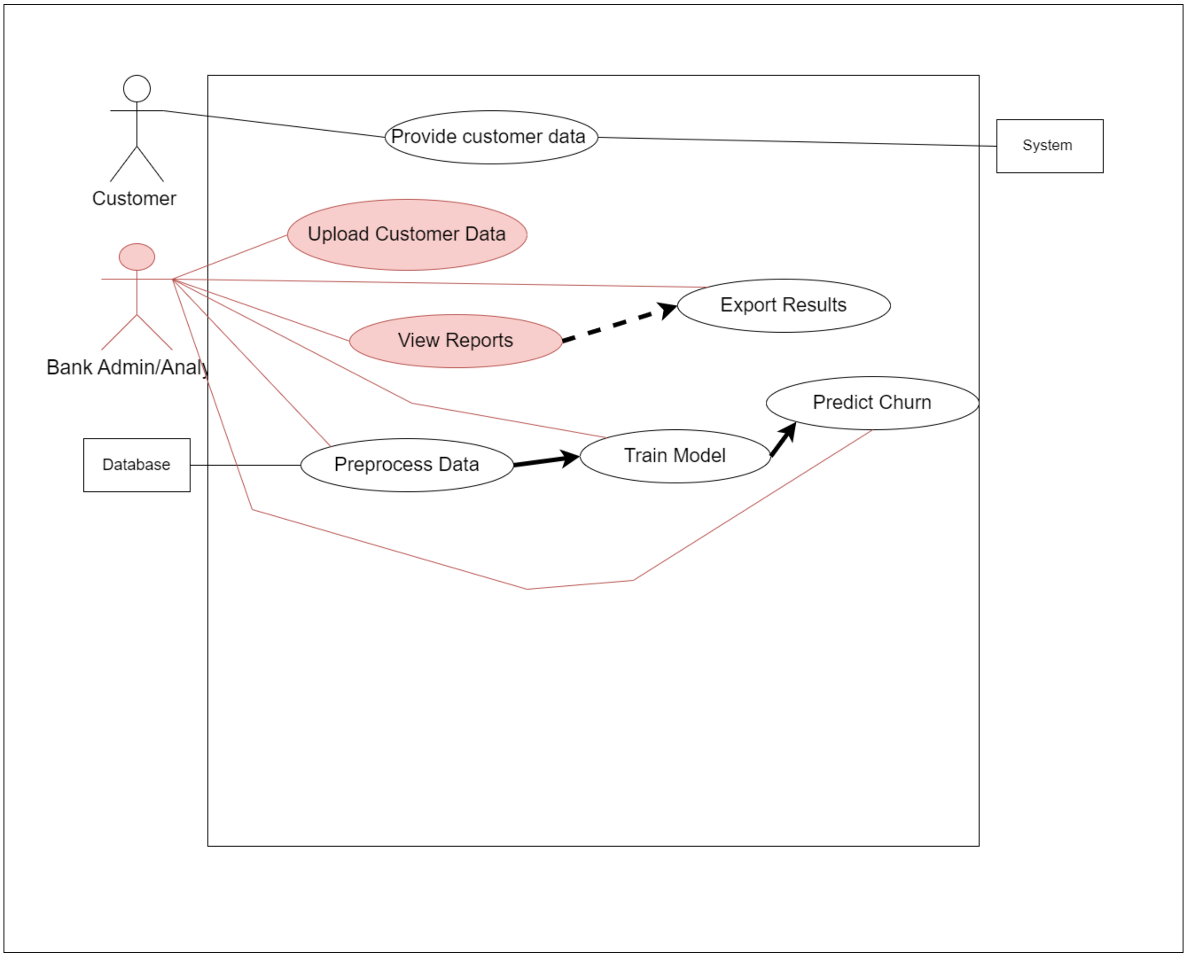
The architecture of the Customer Churn Prediction system consists of several components that work together to provide an efficient and user-friendly experience:

1. Data Input Layer:
   - **ETL Process:** An Extract, Transform, Load (ETL) process will be implemented to clean and preprocess the data before analysis.
2. Data Storage Layer:
   - **Database:** The processed data will be stored in a secure database that complies with Egypt's PDPL. Options could include relational databases (e.g., MySQL) or cloud storage solutions.
3. Data Processing Layer:
   - **Data Preprocessing:** This module will handle tasks such as data cleaning, normalization, and feature extraction to prepare the dataset for machine learning.
   - **Exploratory Data Analysis (EDA):** A set of tools for visualizing and analyzing data patterns will be implemented.
4. Modeling Layer:
   - **Machine Learning Models:** Various algorithms (e.g., logistic regression, decision trees, random forests) will be applied to train the predictive model on the processed data.
   - **Model Evaluation:** This component will evaluate the model's performance using metrics such as accuracy, precision, recall, and F1-score.
5. User Interface Layer:
   - **Web Application:** A user-friendly web interface will allow users to input data, view predictions, and generate reports.
   - **Reporting Tools:** Users will be able to generate and export reports in multiple formats, such as PDF and Excel.
6. Notification System:
   - **Alerts and Notifications:** Users will receive notifications for important events, such as predictions and report generation.

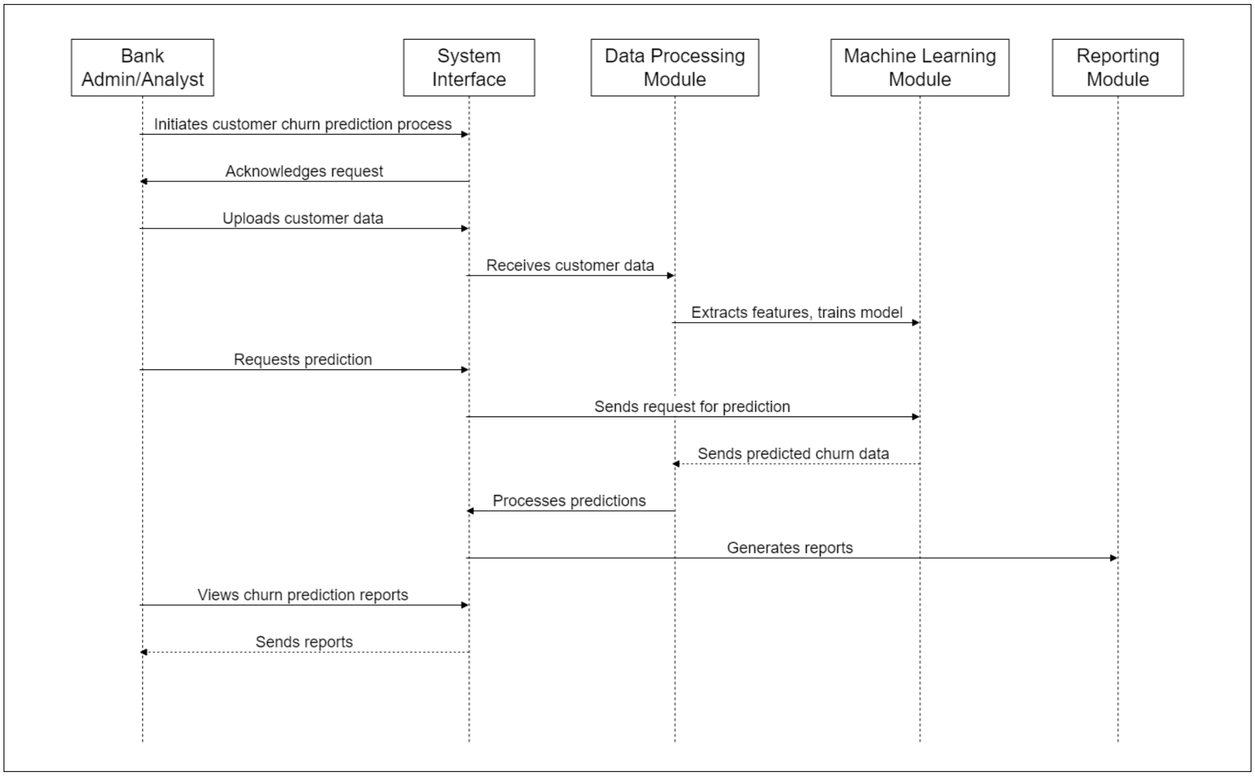
4.2 Data Flow Diagram



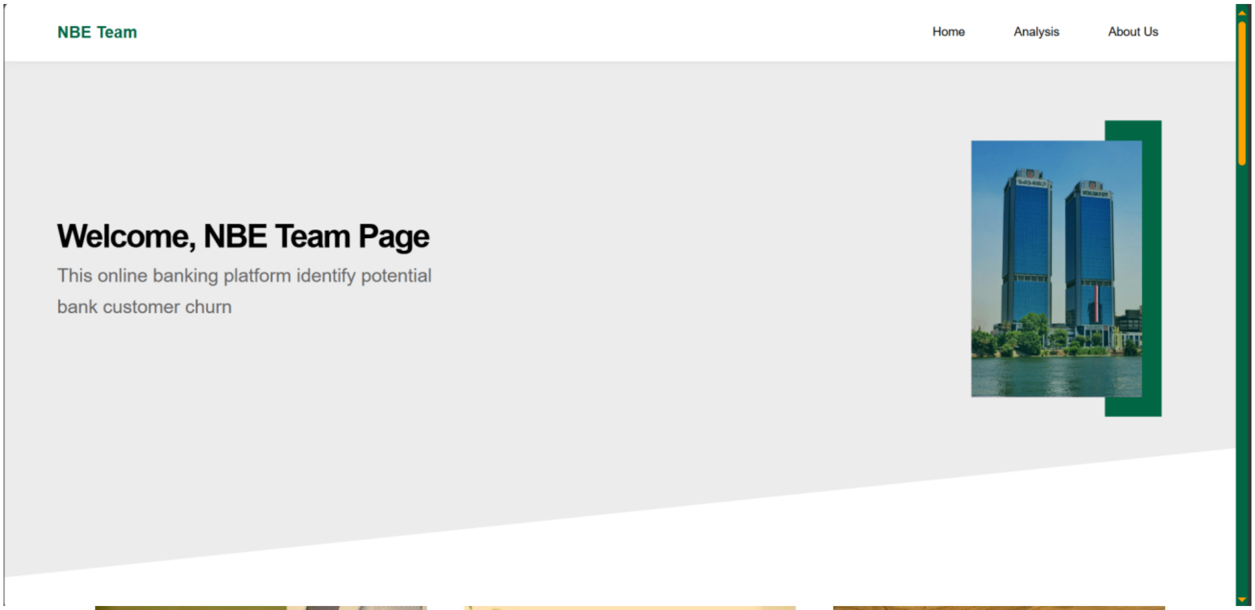
4.3 Use case Diagram



4.4 Sequence Diagram



4.5 Interfaces

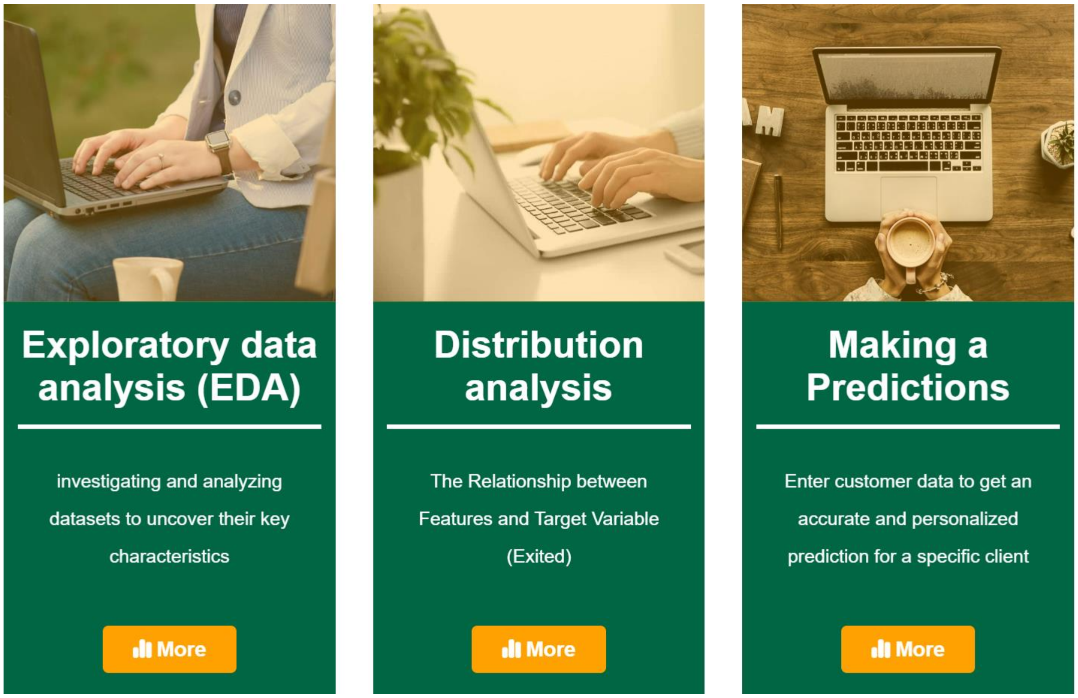


NBE Team

Home Analysis About Us

Welcome, NBE Team Page

This online banking platform identify potential bank customer churn



Exploratory data analysis (EDA)

investigating and analyzing datasets to uncover their key characteristics

More

Distribution analysis

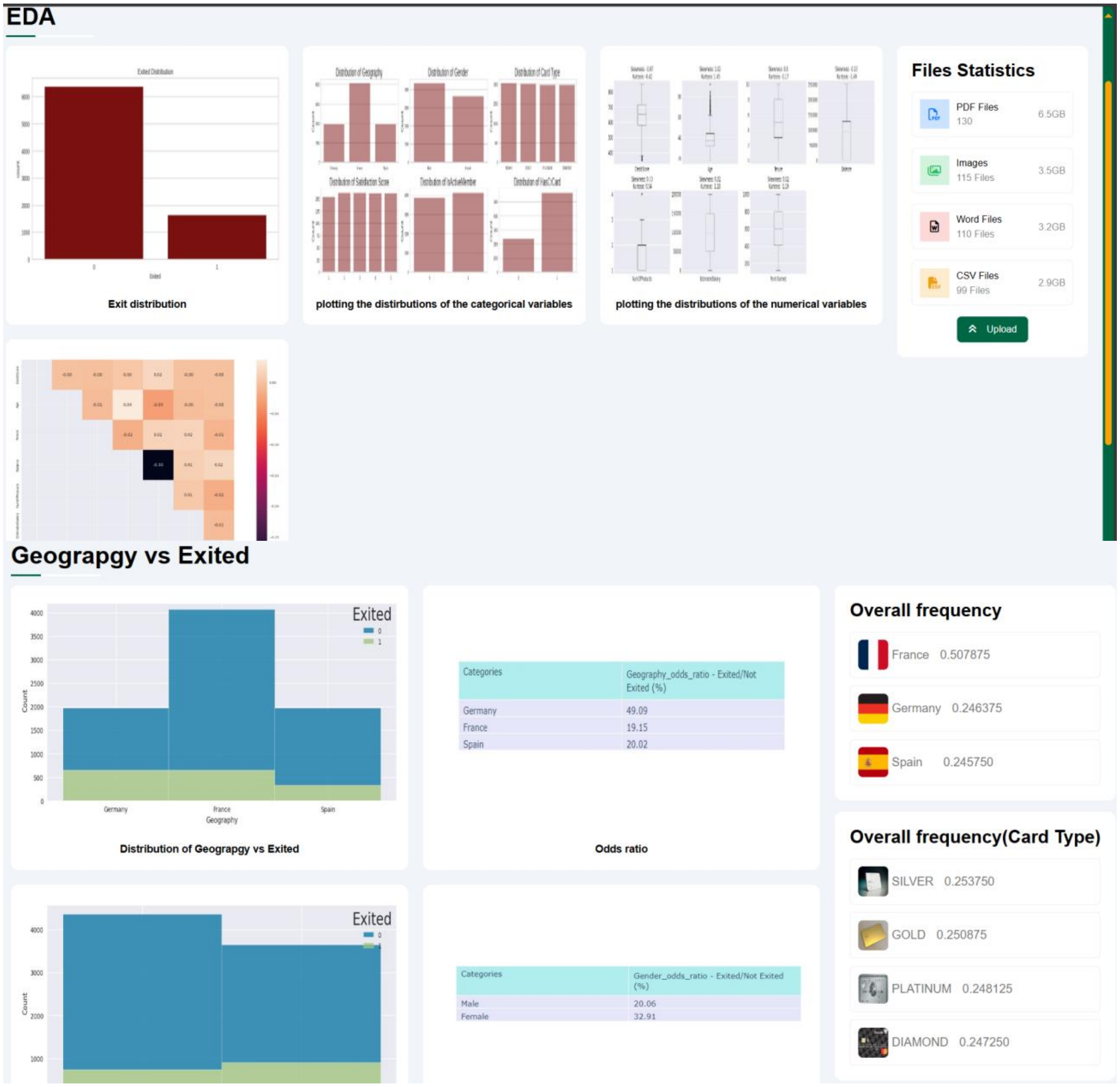
The Relationship between Features and Target Variable (Exited)

More

Making a Predictions

Enter customer data to get an accurate and personalized prediction for a specific client

More



EDA

Exit distribution

plotting the distirbutions of the categorical variables

plotting the distributions of the numerical variables

Files Statistics

PDF Files	130	6.5GB
Images	115 Files	3.5GB
Word Files	110 Files	3.2GB
CSV Files	99 Files	2.9GB

Geograpgy vs Exited

Distribution of Geograpgy vs Exited

Categories	Geography_odds_ratio - Exited/Not Exited (%)
Germany	49.09
France	19.15
Spain	20.02

Odds ratio

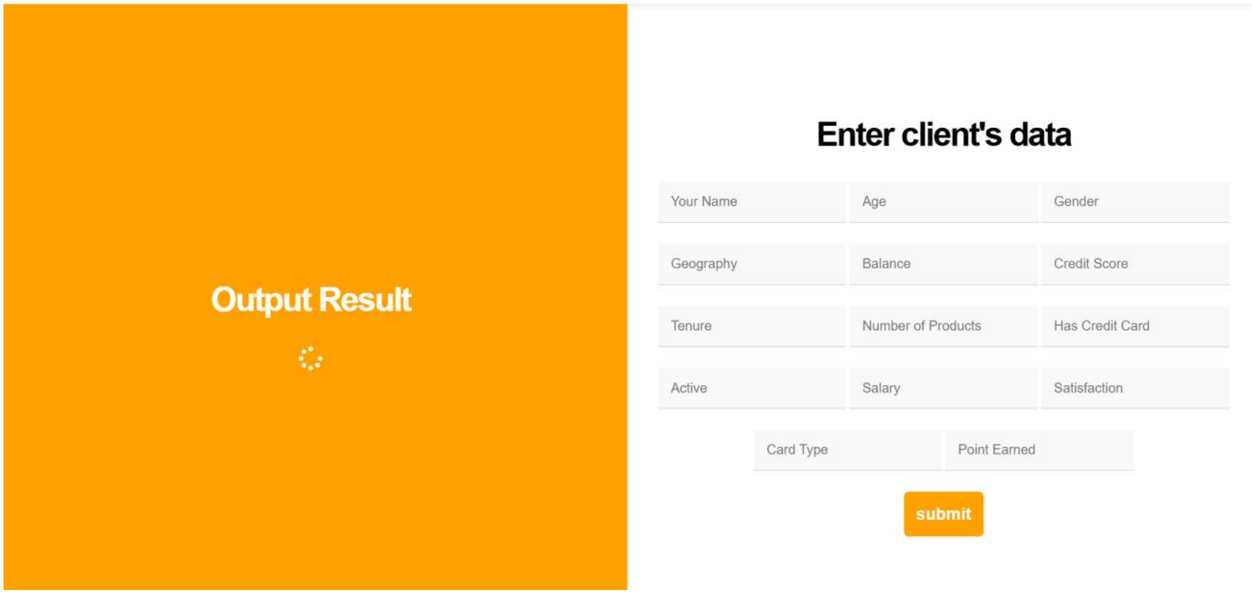
Overall frequency

France	0.507875
Germany	0.246375
Spain	0.245750

Overall frequency(Card Type)

SILVER	0.253750
GOLD	0.250875
PLATINUM	0.248125
DIAMOND	0.247250

Categories	Gender_odds_ratio - Exited/Not Exited (%)
Male	20.06
Female	32.91



5. Implementation

5.1. Tools and Libraries

Python: Used as the primary programming language.

Libraries:

1. Pandas & NumPy: For data manipulation and numerical operations.
2. Matplotlib & Seaborn: For visualization and exploratory data analysis (EDA).
3. Scikit-learn: For machine learning algorithms and preprocessing.
4. Imbalanced-learn (SMOTE): For handling data imbalance in the target variable.
5. Yellowbrick: For model evaluation and visualization.

5.3 Data Preprocessing

- Data Cleaning: Removed irrelevant features such as Complain, CustomerId, RowNumber, and Surname.
- Handling Categorical Variables:

1. Used OneHotEncoder for nominal features (Geography, Gender).
2. Applied OrdinalEncoder to Card Type to encode the card categories hierarchically.

- **Normalization:** Scaled numerical features using MinMaxScaler to normalize them for improved model performance.
- **Oversampling with SMOTE:** Applied the Synthetic Minority Over-sampling Technique (SMOTE) to balance the classes (Exited/Not Exited) in the dataset.

5.4 Model Development

- Model Selection: We tried Random Forest and Logistic regression, and we selected Random Forest Classifier due to its high performance on classification tasks.
- Training and Test Splitting: The dataset was split into an 80/20 ratio for training and testing respectively, ensuring stratified sampling to balance the target variable.
- Model Training: The Random Forest model was trained on oversampled data.
- Evaluation: Performance was evaluated using metrics such as F1-score, precision, and recall.

6. Testing

6.1 Unit Testing:

Train-Test Split:

- The data was split into training (80%) and test (20%) sets using stratified sampling to maintain the proportion of the target class (Exited vs. Not Exited).
- The train-test split was validated to ensure that the split maintained the same class distribution as the original data.

Cross-Validation:

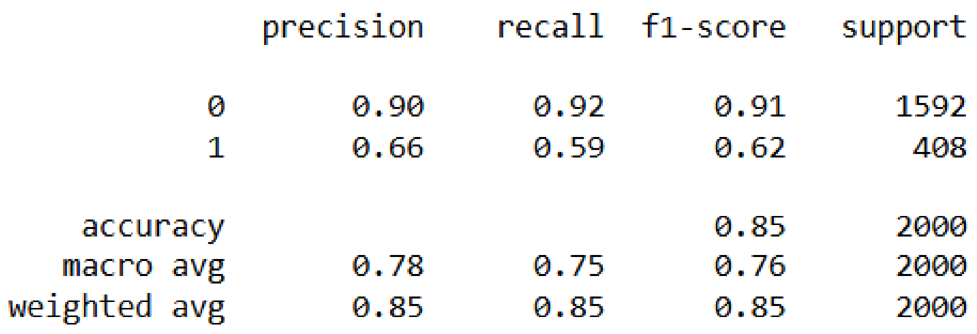
- 5-fold Stratified Cross-Validation was applied to ensure that the model performs consistently across different subsets of the training data. This technique ensures each fold maintains the proportion of classes as in the original dataset, avoiding class imbalance within each fold.
- The F1-score was used as the evaluation metric for cross-validation, focusing on the balance between precision and recall.

Cross-Validation Results:

- Cross-Validation F1 Scores:
- Fold 1: 0.913
- Fold 2: 0.903
- Fold 3: 0.904

- Fold 4: 0.906
- Fold 5: 0.907
- Mean F1 Score: 0.907
- This performance indicates that the model is robust, with minimal variance across different folds.

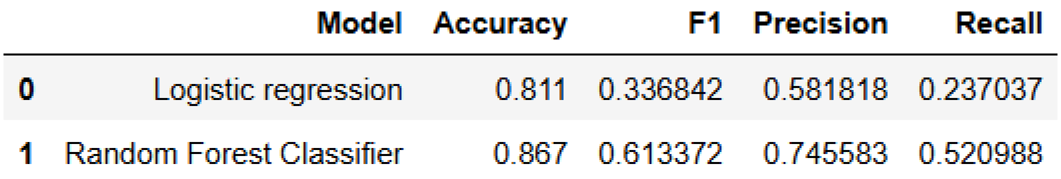
7. Results



```
              precision    recall  f1-score   support

           0       0.90      0.92      0.91      1592
           1       0.66      0.59      0.62       408

    accuracy                           0.85      2000
   macro avg       0.78      0.75      0.76      2000
weighted avg       0.85      0.85      0.85      2000
```



	Model	Accuracy	F1	Precision	Recall
0	Logistic regression	0.811	0.336842	0.581818	0.237037
1	Random Forest Classifier	0.867	0.613372	0.745583	0.520988