Bank Customer Churn Predictor using ML

1. Executive Summary

This project predicts whether a bank customer is likely to leave the bank using machine learning. The Bank Churn dataset was analyzed to understand how factors such as credit score, age, tenure, account balance, number of products, and customer activity influence churn behavior. After data preprocessing, exploratory data analysis (EDA), and feature engineering, a **Random Forest classification model** was trained and evaluated. The final model provides reliable churn predictions and demonstrates a complete end-to-end machine learning workflow for customer retention analysis.

🔗 **Project Links:**

- GitHub Repository
- Live Application
- Presentation

---

2. Problem Statement

Customer churn poses a significant challenge for banks, as retaining existing customers is more cost-effective than acquiring new ones. This project aims to build a classification model that predicts whether a customer will exit the bank based on demographic, financial, and behavioral attributes. The model helps identify high-risk customers early, enabling banks to implement targeted retention strategies and make data-driven business decisions.

---

3. Dataset Description

Column Name	Description
RowNumber	Sequential index number of each record (not a customer feature).
CustomerId	Unique identification number assigned to each customer.
Surname	Customer's last name (used only for identification, not modeling).
CreditScore	Customer's credit score indicating financial reliability.
Geography	Country where the customer is located (e.g., France, Spain).
Gender	Customer's gender (Male/Female).
Age	Age of the customer in years.

Tenure	Number of years the customer has been with the bank.
Balance	Account balance maintained by the customer.
NumOfProducts	Number of bank products the customer is using.
HasCrCard	Indicates whether the customer owns a credit card (1 = Yes, 0 = No).
IsActiveMember	Indicates if the customer is an active bank member (1 = Active, 0 = Inactive).
EstimatedSalary	Estimated annual salary of the customer.
Exited	Target variable indicating customer churn (1 = Exited, 0 = Retained).
Geography	Germany .

4. Project Objectives

The project followed a structured machine learning pipeline consisting of :

- **Data Exploration and Preprocessing**
- **Exploratory Data Analysis (EDA)**
- **Feature Engineering**
- **Model Training**
- **Model Evaluation and Deployment**
- **Key Insights from Data Analysis**

4.1 Tech Stack

- **Programming Language:** Python
- **Data Analysis & Visualization:** Pandas, NumPy, Matplotlib, Seaborn
- **Machine Learning:** Scikit-learn
- **Model Deployment:** Streamlit
- **Model Serialization:** Joblib

4.2 Data Exploration

The dataset was first examined to understand its structure, check for missing values, and review basic statistics. This step ensures a clear understanding of the variables, their types, and any potential issues before modeling.

```
: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   RowNumber        10000 non-null  int64
 1   CustomerId       10000 non-null  int64
 2   Surname          10000 non-null  object
 3   CreditScore      10000 non-null  int64
 4   Geography        10000 non-null  object
 5   Gender           10000 non-null  object
 6   Age              10000 non-null  int64
 7   Tenure           10000 non-null  int64
 8   Balance          10000 non-null  float64
 9   NumOfProducts    10000 non-null  int64
 10  HasCrCard        10000 non-null  int64
 11  IsActiveMember   10000 non-null  int64
 12  EstimatedSalary  10000 non-null  float64
 13  Exited           10000 non-null  int64
dtypes: float64(2), int64(9), object(3)
memory usage: 1.1+ MB
```

4.3 Data Preprocessing

Data preprocessing involved cleaning and preparing the dataset for machine learning. Irrelevant columns like name were removed, missing values were handled, and duplicates were checked. This ensures that the data is accurate and suitable for modeling.

Categorical features were converted into numerical format to make them suitable for machine learning models. The Gender feature was label encoded by mapping Female to 0 and Male to 1.

```
: df["Gender"] = df["Gender"].map({'Female': 0, 'Male': 1})
```

The Geography feature was transformed using one-hot encoding with the first category dropped to avoid multicollinearity.

```
df = pd.get_dummies(df, columns=['Geography'], drop_first=True)
```

4.4 Exploratory Data Analysis (EDA)

EDA was performed to uncover patterns and relationships between features and fuel efficiency (mpg). Visualizations such as histograms, correlation heatmaps, and boxplots were used to identify trends, detect outliers, and highlight features most related to the target variable.

Finding the Correlation:

Before proceeding with **feature selection**, it is important to examine the **correlation between feature** This analysis helps provide an initial understanding of how strongly features are related to each other and to the target variable.

What is Correlation:

Correlation is a measure that shows how strongly two variables are related to each other.

Note:

However, this is **for reference purposes only** and should not be considered a definitive method for selecting features to train the model.

```
]: corr = dfinclude = df.select_dtypes(include = "number").corr()
   corr
```

[9]:

	RowNumber	CustomerId	CreditScore	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary
RowNumber	1.000000	0.004202	0.005840	0.018196	0.000783	-0.006495	-0.009067	0.007246	0.000599	0.012044	-0.005988
CustomerId	0.004202	1.000000	0.005308	-0.002641	0.009497	-0.014883	-0.012419	0.016972	-0.014025	0.001665	0.015271
CreditScore	0.005840	0.005308	1.000000	-0.002857	-0.003965	0.000842	0.006268	0.012238	-0.005458	0.025651	-0.001384
Gender	0.018196	-0.002641	-0.002857	1.000000	-0.027544	0.014733	0.012087	-0.021859	0.005766	0.022544	-0.008112
Age	0.000783	0.009497	-0.003965	-0.027544	1.000000	-0.009997	0.028308	-0.030680	-0.011721	0.085472	-0.007201
Tenure	-0.006495	-0.014883	0.000842	0.014733	-0.009997	1.000000	-0.012254	0.013444	0.022583	-0.028362	0.007784
Balance	-0.009067	-0.012419	0.006268	0.012087	0.028308	-0.012254	1.000000	-0.304180	-0.014858	-0.010084	0.012797
NumOfProducts	0.007246	0.016972	0.012238	-0.021859	-0.030680	0.013444	-0.304180	1.000000	0.003183	0.009612	0.014204
HasCrCard	0.000599	-0.014025	-0.005458	0.005766	-0.011721	0.022583	-0.014858	0.003183	1.000000	-0.011866	-0.009933
IsActiveMember	0.012044	0.001665	0.025651	0.022544	0.085472	-0.028362	-0.010084	0.009612	-0.011866	1.000000	-0.011421
EstimatedSalary	-0.005988	0.015271	-0.001384	-0.008112	-0.007201	0.007784	0.012797	0.014204	-0.009933	-0.011421	1.000000
Exited	-0.016571	-0.006248	-0.027094	-0.106512	0.285323	-0.014001	0.118533	-0.047820	-0.007138	-0.156128	0.012097

- **Age ↔ Exited = 0.29**
- **Balance ↔ NumOfProducts = −0.30**
- **IsActiveMember ↔ Exited = −0.16**
- **Balance ↔ Exited = 0.12**
- **Gender ↔ Exited = −0.11**

📌 **These features show the most noticeable correlations in the dataset,**

4.5 Key Insights from Data Analysis:

Geography & Products Drive Churn: German customers with 4 bank products show the highest churn risk due to potential dissatisfaction or better competitor offers.

Inactivity is Critical: Inactive members are far more likely to leave, regardless of balance — highlighting the need for engagement strategies.

High Balance + Low Activity = Danger: Customers with large balances but inactive accounts are prime churn candidates, indicating lost loyalty.

Demographics Matter Slightly: Younger customers, those with short tenure, and females exhibit moderately higher churn rates.

4.6 Feature Engineering

Relevant features were selected based on correlation analysis and domain knowledge.

```
]: x = df.drop(columns = ["RowNumber","CustomerId","Exited","Surname"])
```

```
y  =df["Exited"]
```

Train_Test_Split:

Finally, the dataset was split into training (80%) and testing (20%) sets for model evaluation.

```
: x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=42)
```

Scaling Techniques:

Numerical features were standardized using **StandardScaler** to ensure consistent feature ranges and improve model performance. Scaling was applied only to models that are sensitive to feature magnitude.

- **Scaling Applied:** Logistic Regression, KNN, SVC
- **Scaling Not Required:** Decision Tree, Random Forest, XGBoost
- **Optional Scaling:** Gaussian Naive Bayes

Tree-based models do not rely on distance calculations and therefore do not require feature scaling.

```
: sc = StandardScaler()
x_train_sc = sc.fit_transform(x_train)
x_test_sc = sc.fit(x_test)
```

5. Model Development

Multiple machine learning models were developed to predict customer churn, including **Logistic Regression, KNN, SVC, Decision Tree, Random Forest, XGBoost, and Naive Bayes**. Each model was trained using preprocessed and feature-engineered data to learn patterns associated with customer churn.

6. Model Evaluation

The trained models were evaluated using standard classification metrics such as Accuracy, Precision, Recall, and F1-score. This evaluation ensured a fair comparison of model performance, particularly in identifying customers at risk of churning.

.

✅ **Random Forest is the best-performing model overall**

Here's why 👇

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.198	0.197	**0.997**	0.328
SVC	0.197	0.197	**1.000**	0.328
Decision Tree	0.838	**0.662**	0.359	0.465
Random Forest	**0.855**	0.632	**0.626**	**0.629**
XGBoost	0.843	0.593	0.631	0.612

7. Model Selection & Interpretation

Based on evaluation results, the **Random Forest model** was selected as the best-performing model due to its strong balance between precision and recall. Feature analysis and model behavior were reviewed to understand the key factors influencing churn and to ensure reliable predictions.

8. Save The Model and Deployment

The final model was saved using serialization techniques and deployed as an interactive **Streamlit web application**. This allows users to input customer details and receive real-time churn predictions through a user-friendly interface.

```
jb.dump(rf,"Churn_Prediction.joblib")
jb.dump(sc,"scaler.joblib")
```

9. Conclusion

This project demonstrates a complete end-to-end machine learning workflow, from data analysis to deployment. By combining a robust predictive model with an interactive application, the solution provides practical value for proactive customer retention and data-driven decision-making in the banking domain.