PREDICTIVE ANALYTICS ON RETAIL BANKING

PRESENTED BY:

C.MAHESWARI-MSC(AI&DS)

REG.NO:2023557016

DEPARTMENT OF COMPUTER APPLICATIONS

TABLE OF CONTENTS

- ABSTRACT
- INTRODUCTION
- SYSTEM SPECIFICATION
- DATASET DESCRIPTION
- EXISTING SYSTEM
- PROPOSED SYSTEM
- SYSTEM IMPLEMENTATION
- MACHINE LEARNING ALGORITHMS USED
- HYPOTHESIS TESTING
- SCREENSHOTS
- CONCLUSION

ABSTRACT

- In the dynamic banking industry, analytics plays a pivotal role in addressing various challenges such as identifying the most suitable product to recommend to a customer, determining the optimal timing for marketing the product, and selecting the most effective communication channel.
- This study focuses on a dataset derived from direct marketing campaigns by a banking institution, conducted primarily through phone calls. The aim is to predict whether a client will subscribe to a term deposit based on multiple contacts made with the same client.
- By leveraging analytics, the goal is to enhance the efficiency of marketing campaigns and improve customer engagement.

INTRODUCTION

- **Retail Banking** offers financial services like savings accounts, loans, and credit cards directly to individuals. With increasing competition, optimizing marketing efforts is crucial.
- **Predictive Analytics** uses data analysis and machine learning to predict customer behavior. In banking, it helps improve marketing strategies and customer engagement.
- This project aims to use predictive analytics to enhance **direct marketing campaigns** in retail banking.
- The goal is to predict if a customer will **subscribe to a term deposit** based on historical interactions.
- The **objective** is to optimize marketing strategies and increase customer engagement by predicting customer responses.

SYSTEM SPECIFICATION

HARDWARE SPECIFICATION

PROCESSOR	INTEL CORE i7
CPU CLOCK SPEED	2.80 GHz
RAM	16 GB
HARD DISC CAPACITY	500 GB
DISPLAY TYPE	INTEGRATED GPU
INTERNET CONNECTION	STABLE INTERNET CONNECTION

SOFTWARE SPECIFICATION

PROGRAMMING LANGUAGE	PYTHON
OPERATING SYSTEM	WINDOWS 11
FRONT-END FRAMEWORK	STREAMLIT
DEVELOPING ENVIRONMENT	GOOGLE COLABATORY
DATASET	MICROSOFT EXCEL
TEXT EDITOR	VISUAL STUDIO

LIBRARIES USED

NumPy	For Numerical Computing and Array Operations.
Pandas	For Data Manipulation, particularly for tabular data.
Seaborn	For creating informative statistical charts.
Pickle	To save and load trained Machine Learning models.
Matplotlib	Used for generating 2D visualizations.
Scikit-learn	A library for machine learning algorithms and model evaluation.
Streamlit	Helps create an interactive web-based user interface.
Psycopg2	PostgreSQL adapter for connecting and storing results in a database.

DATASET DESCRIPTION

- The dataset used in this project forms the foundation for predictive analytics in retail banking.
- It is provided in **CSV (Comma-Separated Values)** format, which is a widely used and easily accessible data storage format ideal for structured data analysis.
- It is derived from a Portuguese retail bank’s telemarketing campaign data, where the objective was to encourage clients to subscribe to a term deposit product.
- The dataset provides valuable insight into customer demographics, financial standing, communication details, and past campaign outcomes, all of which are used to predict customer behaviour.

DATASET OVERVIEW

- **Total Records (Rows):** 11,162
- **Total Features (Columns):** 17
- **File Format:** CSV (.csv)
- **Type of Problem:** Binary Classification
- **Target Variable:** deposit (indicates whether the customer subscribed to a term deposit)
- **Source:** Real-world banking campaign data

DETAILED FEATURE DESCRIPTION

Feature	Description
age	Age of the customer (numeric).
job	Job title or profession (categorical: e.g., admin, technician, services).
marital	Marital status of the customer (e.g., married, single, divorced).
education	Highest level of education attained.
default	Indicates if the customer has credit in default (yes/no).
balance	Account balance in euros (numeric).
housing	Indicates if the customer has a housing loan (yes/no).
loan	Indicates if the customer has a personal loan (yes/no).
contact	Communication type used (e.g., cellular, telephone, unknown).
day	Last contact day of the month (numeric).
month	Last contact month (e.g., may, jun).
duration	Duration of the last contact in seconds (numeric).
campaign	Number of contacts performed during this campaign.
pdays	Days passed since the client was last contacted (-1 means never contacted before).
previous	Number of contacts performed before this campaign.
poutcome	Outcome of the previous marketing campaign (e.g., success, failure, unknown).
deposit	Target variable – whether the client subscribed to a term deposit (yes/no).

EXISTING SYSTEM

- Traditional banking marketing relies on mass emails, cold calls, and broad advertisements, treating all customers alike without considering individual preferences or behaviors. As a result, banks often contact uninterested customers, leading to low response rates, wasted time, and increased costs.
- Marketing decisions are usually based on assumptions or past trends rather than real-time data, with no predictive system to guide customer targeting. Without intelligent analysis of customer history and patterns, it becomes difficult to recommend the right product at the right time.
- Studies show traditional mass marketing yields less than a 2% response rate, while data-driven campaigns can achieve over 10%, highlighting the need for smarter, targeted approaches.

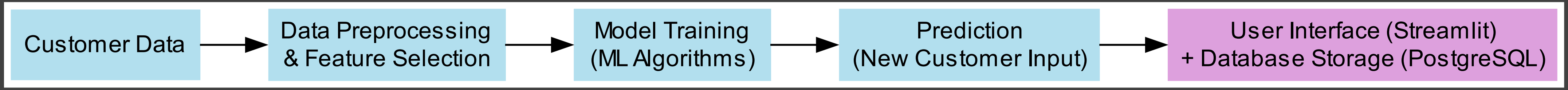
PROPOSED SYSTEM

- The proposed system introduces intelligence and automation into customer targeting by using machine learning to predict which clients are likely to subscribe to a term deposit.

- It analyzes past customer behavior through models like Logistic Regression, Random Forest, Decision Tree, K-Nearest Neighbors, and Support Vector Machines to identify key patterns influencing customer decisions.

- The system features a user-friendly **Streamlit** interface, allowing bank staff to input customer details and receive instant predictions, while a **PostgreSQL** database stores all data and results for future use.

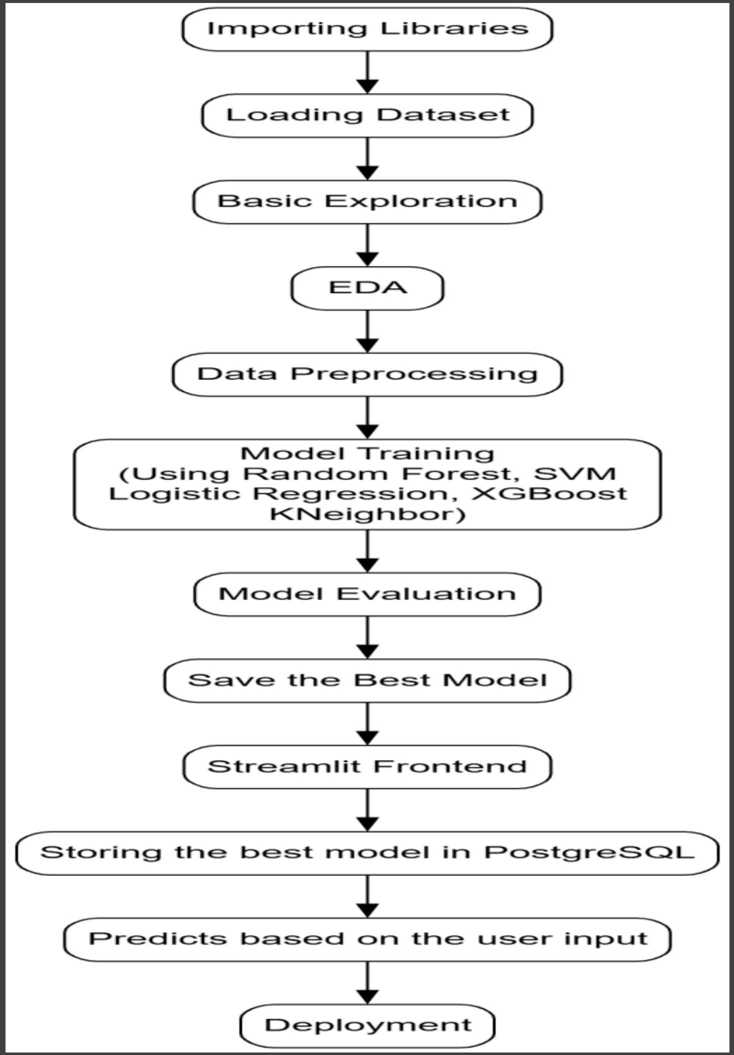
- Designed to be scalable and modular, the system can handle growing datasets, adapt to new features, and integrate advanced models or APIs in the future.
- This data-driven approach enables banks to run more targeted campaigns, reduce costs, and significantly improve customer conversion rates.


Customer Data
Data Preprocessing & Feature Selection
Model Training (ML Algorithms)
Prediction (New Customer Input)
User Interface (Streamlit) + Database Storage (PostgreSQL)

SYSTEM IMPLEMENTATION

System implementation is the crucial phase where all the designed components data, machine learning models, backend storage, and the user interface come together to form a fully functional application. In this project, we transitioned from analysis and modeling to an interactive prediction system that can be used in real-time.

The implementation integrates a trained machine learning model into a **Streamlit-based web interface**, backed by a **PostgreSQL** database for secure model and data storage. **Visual Studio Code (VS Code)** served as the development environment for writing, testing, and running the application, ensuring a seamless workflow from backend development to frontend deployment.


Importing Libraries
Loading Dataset
Basic Exploration
EDA
Data Preprocessing
Model Training
(Using Random Forest, SVM
Logistic Regression, XGBoost
KNeighbor)
Model Evaluation
Save the Best Model
Streamlit Frontend
Storing the best model in PostgreSQL
Predicts based on the user input
Deployment

1. IMPORTING NECESSARY LIBRARIES

The project begins with importing all the essential Python libraries required for data manipulation, visualization, machine learning, and frontend development.

2. LOADING THE DATASET

The dataset, which contains customer information relevant to banking services, is loaded using **pandas**. It includes attributes like age, job type, marital status, balance, loan details, etc., which are crucial in predicting whether a customer will subscribe to a term deposit. This step ensures the dataset is accessible for further analysis and modeling.

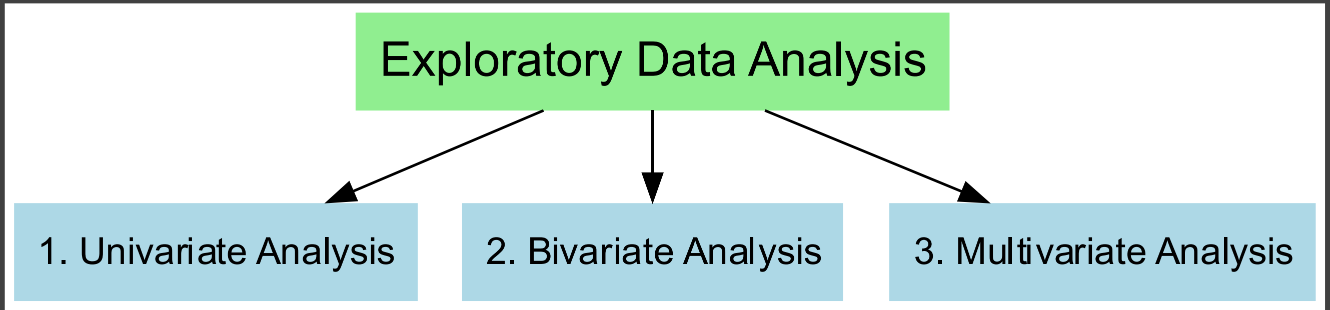
3. BASIC DATA EXPLORATION

This phase involves an initial look at the dataset to understand its structure and contents:

- **Printing the first five rows** helps us get a snapshot of the data.
- **Information about the dataset** shows the number of entries, column names, data types, and non-null values.
- **Statistical summary** (via .describe()) gives insights into the distribution of numerical variables.
- **Checking unique values in categorical columns** helps identify categorical features and their distinct values.
- **Checking duplicate values** ensures data quality by identifying any redundant entries.

4. EXPLORATORY DATA ANALYSIS (EDA)

EDA allows us to visualize and understand data patterns, trends, and relationships.



Exploratory Data Analysis

1. Univariate Analysis

2. Bivariate Analysis

3. Multivariate Analysis

a) Univariate Analysis

- **Histograms** for numerical features (e.g., age, balance) reveal distribution and skewness.
- **Bar plots** for categorical features (e.g., job, education, marital status) show frequency distributions.

b) Bivariate Analysis

- **Boxplots** help detect outliers by comparing numerical variables across categorical groups.
- **Correlation heatmap** identifies relationships between numerical features and helps with feature selection.

c) Multivariate Analysis

- **Pair plots** provide a visual overview of multiple feature relationships.
- **Target variable analysis** (e.g., how different features influence the likelihood of deposit subscription) gives valuable predictive insights.

5. DATA PREPROCESSING

To prepare the data for modeling, several preprocessing techniques are applied:

- **Handling missing values** by imputation or removal ensures clean input.
- **Encoding categorical variables** using Label Encoding or One-Hot Encoding converts textual data into numerical format.
- **Checking and fixing skewness** ensures normally distributed data, improving model performance.
- **Handling outliers** (if needed) to prevent model bias.

6. MODEL TRAINING & EVALUATION

At this stage, the data is ready to be used for building predictive models.

- **Splitting data** into training and testing sets (80:20) ensures fair evaluation.
- **Training multiple ML models** such as Logistic Regression, Random Forest, SVM, XGBoost, and K-Nearest Neighbors.
- **Evaluating models** using performance metrics like Accuracy, Precision, Recall, and F1-score helps identify the most effective algorithm.
- **Hyperparameter tuning** is done using techniques like RandomizedSearchCV to improve model accuracy.

7. SAVING THE BEST MODEL

After model comparison, the algorithm with the highest overall performance is chosen. This model is **saved using Pickle**, making it ready for deployment in the prediction system. This saved model is later used for generating real-time predictions in the frontend.

8. FRONTEND INTEGRATION WITH STREAMLIT

To make the system user-friendly, a web-based interface is built using **Streamlit**. The frontend allows users to input customer data through a form and view instant predictions. The interface also includes result display and basic visual elements to improve usability and interaction.

9. CONNECTING WITH POSTGRESQL

In this project, **PostgreSQL** is used to:

- Store the **best-performing model** securely.
- Keep a record of user inputs and prediction results. This connection is established using psycopg2 allowing the Streamlit frontend to interact with the database seamlessly.

10. PREDICTING THE RESULTS

The final stage allows users to make predictions in real-time. Users input the required features via the frontend, which are processed and passed to the saved ML model. The prediction result whether the customer is likely to accept the term deposit is then displayed.

MACHINE LEARNING ALGORITHMS

In this project, various ML algorithms were applied to predict whether a customer will subscribe to a term deposit, based on customer demographic and behavioral data.

Algorithms Used in This Project:

1. **Logistic Regression**
2. **Support Vector Machine (SVM)**
3. **Random Forest Classifier**
4. **XGBoost Classifier**
5. **K-Nearest Neighbors (KNN)**

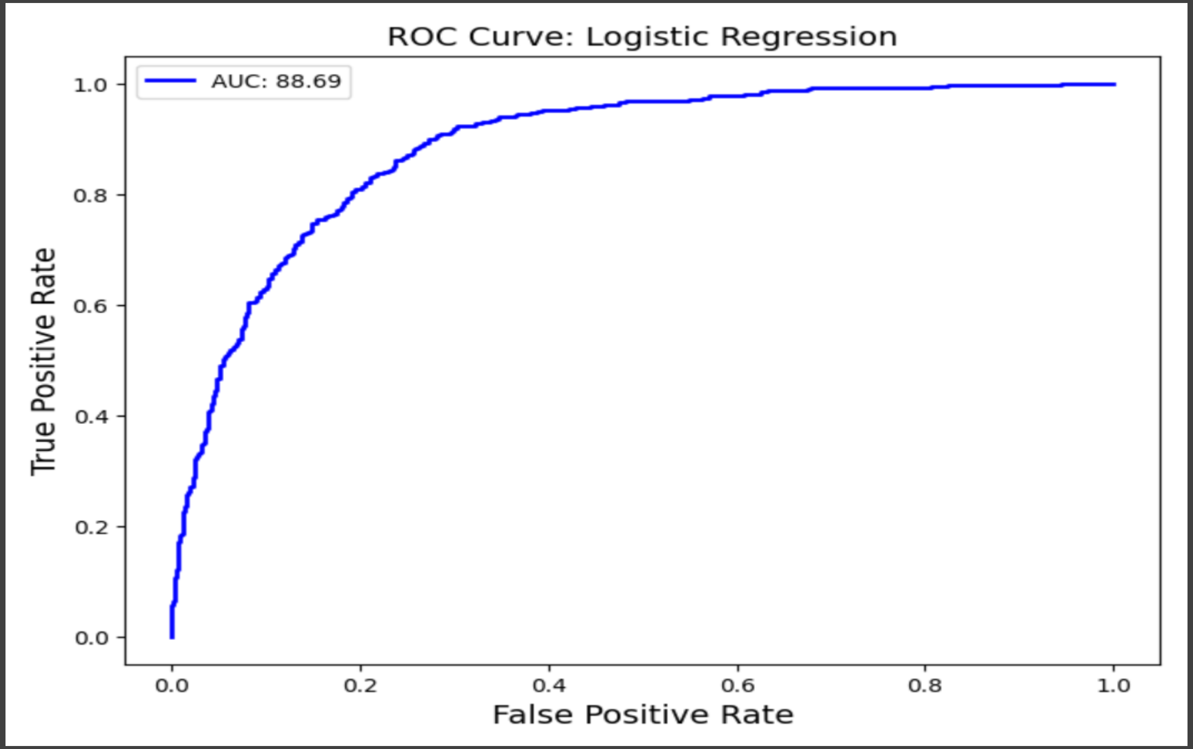
1. LOGISTIC REGRESSION

- Logistic Regression is a statistical model used for binary classification tasks, which estimates the probability of a binary outcome using a logistic function.
- In this project, Logistic Regression was used to predict whether a customer would subscribe to a term deposit based on features like age, job, and previous campaign responses.

Mathematical Function:

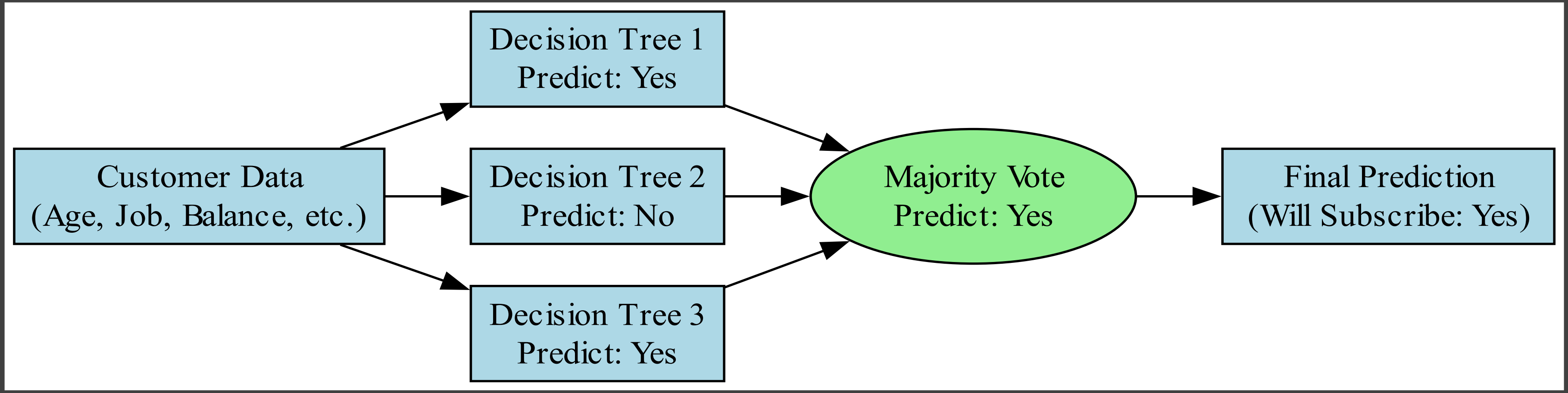
$$f(x) = 1 / (1 + e\hat{}-(\beta_0 + \beta_1 x_1 + ... + \beta_n x_n))$$

- Logistic Regression helped establish a baseline model for predicting customer subscriptions and was used to compare the effectiveness of more complex algorithms.

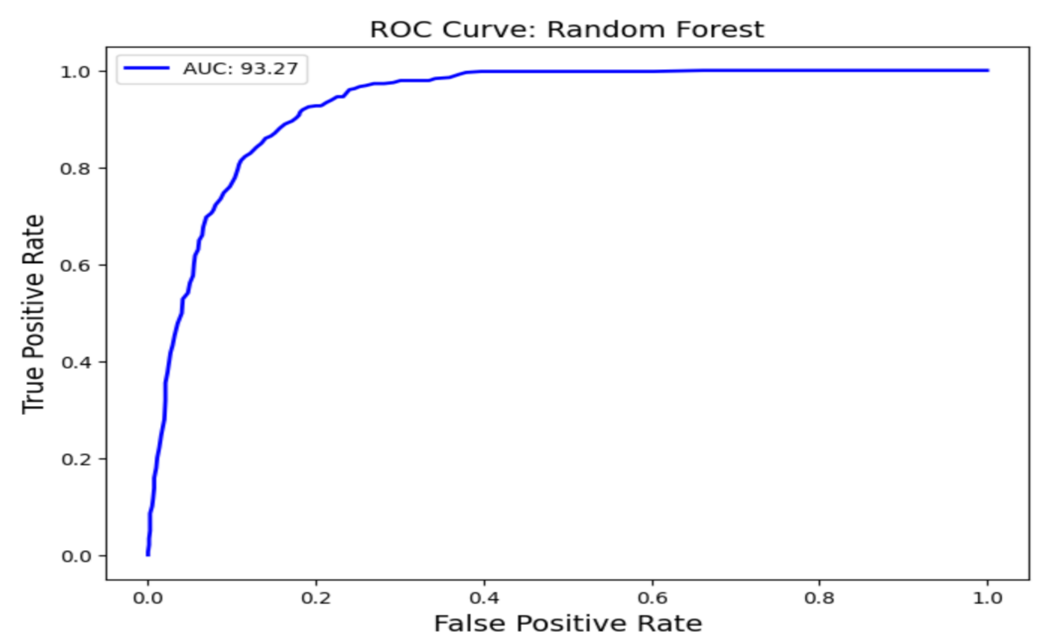


2. RANDOM FOREST CLASSIFIER

- Random Forest is an ensemble learning technique that builds multiple decision trees and aggregates their predictions to improve accuracy and stability.
- It helps classify customers effectively by capturing nonlinear relationships between features.



- **Model Training**: Trained the Random Forest model with 100 decision trees using customer data.
- **Hyperparameter Tuning**: Fine-tuned **n_estimators** and **random_state** for optimal performance.
- **Evaluation Metrics**: Assessed model performance with **accuracy**, **recall**, and **ROC AUC**.
- **Cross-Validation**: Used cross-validation to check model accuracy across multiple data splits.
- **Performance Analysis**: Analyzed results using **confusion matrix** and **classification report**.



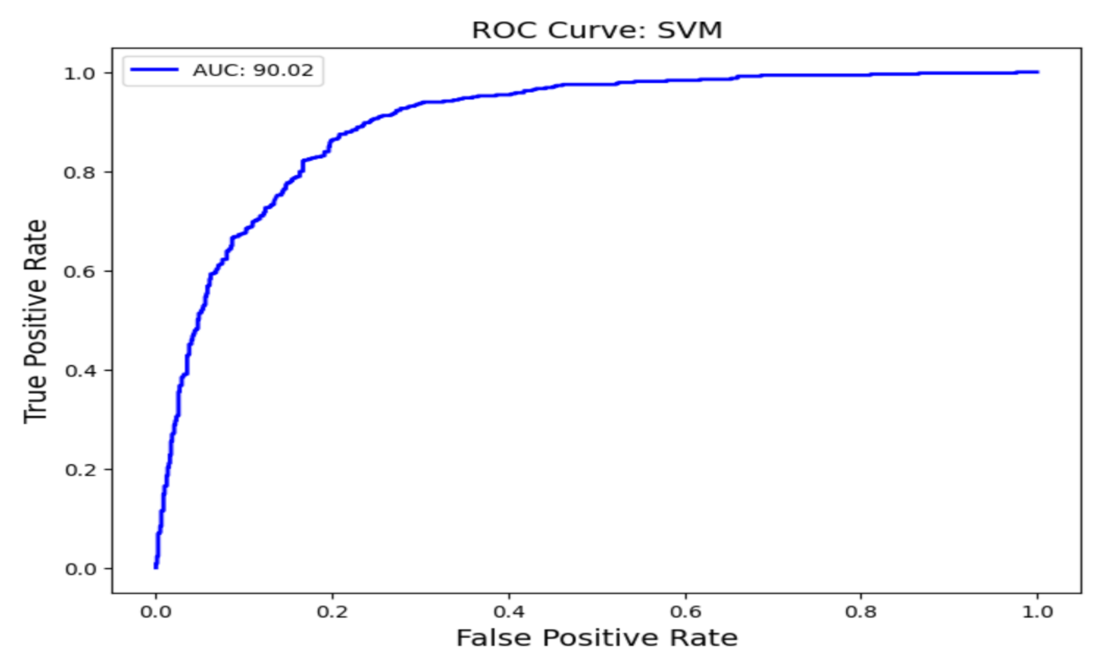
3. SUPPORT VECTOR MACHINE (SVM)

- Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification tasks.
- It works by finding the optimal boundary (called a hyperplane) that separates data points of different classes in the feature space.
- The goal is to maximize the margin between the classes for better generalization to unseen data.
- In this project, SVM is used to classify whether a customer will subscribe to a term deposit by identifying an optimal hyperplane that separates customers who accept the offer from those who don’t.
- SVM aims to provide a robust prediction by handling high-dimensional data efficiently.

Mathematical Function: f(x)=wTx+b

where,

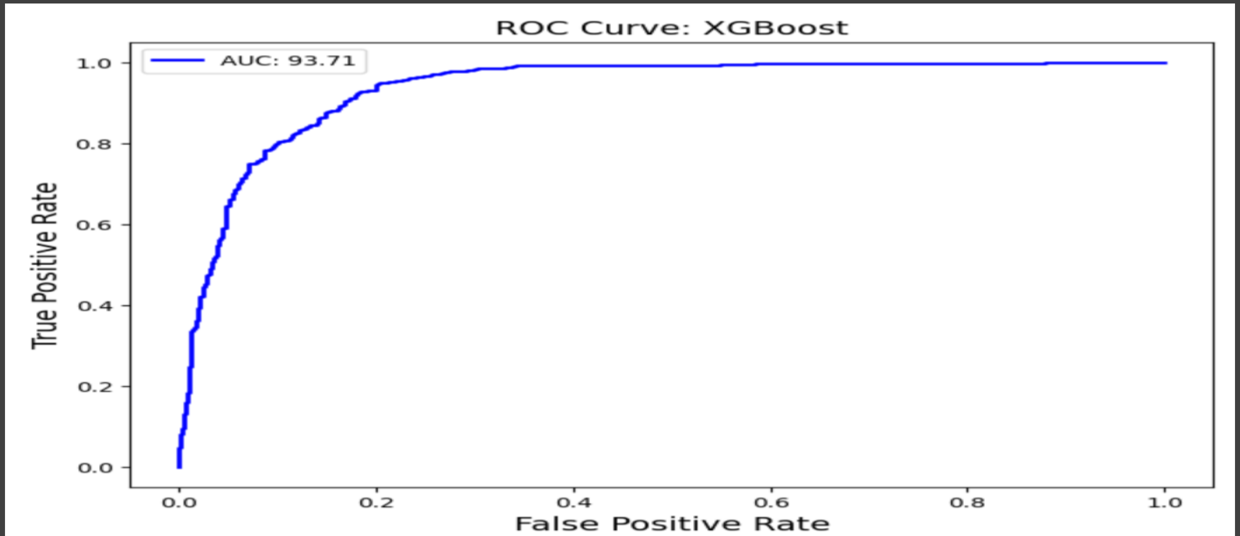
- ‘w’ is the weight vector (normal to the hyperplane),
- ‘x’ is the input feature vector,
- ‘b’ is the bias term that shifts the hyperplane.



4. XGBOOST CLASSIFIER

- XGBClassifier is an advanced machine learning algorithm based on gradient boosting.
- It is a high-performance machine learning algorithm that builds an ensemble of decision trees, where each tree corrects the errors of the previous one to improve model accuracy.
- XGBoost was used to predict whether a customer will subscribe to a term deposit by leveraging its ability to handle complex patterns in structured banking data and enhance model performance.

- In this project, XGBoost was trained on customer features and used to classify customers effectively, contributing to improved prediction accuracy and outperforming other models in terms of efficiency and speed.



5. K-Nearest Neighbors (KNN)

- **K-Nearest Neighbors (KNN)** is a simple machine learning algorithm used for both classification and regression tasks.
- It works by finding the 'k' closest data points (neighbors) in the feature space and classifying a new data point based on the majority class of those neighbors.
- KNN does not require any model training and makes predictions based on the entire dataset, which makes it effective for smaller datasets
- In this project, KNN was used to predict whether a customer would subscribe to a term deposit by comparing each customer’s features with the nearest neighbors in the feature space.

SAVING THE MODEL IN POSTGRESQL

- Once the Random Forest Classifier was trained and evaluated to achieve the highest accuracy, the model was saved for future use.
- The trained model was serialized using the **Pickle** library, which allows the model to be stored as a binary file.
- This serialized model was then uploaded and stored in a **PostgreSQL database**, ensuring easy access and retrieval whenever needed.
- Storing the model in PostgreSQL ensures that the model can be efficiently used for future predictions without the need to retrain it each time.

INTEGRATION WITH STREAMLIT:

- To enable easy interaction with the model, **Streamlit** was used to develop a simple web-based interface.
- The user interface allows bank staff or marketing teams to input customer details, after which the stored model is fetched from PostgreSQL, and predictions are generated in real-time.
- This integration makes the model easily accessible through a user-friendly interface, streamlining the process for making customer subscription predictions.

HYPOTHESIS TESTING

- Hypothesis testing is a statistical method used to determine whether there is enough evidence to reject a null hypothesis based on sample data.
- It helps in making decisions or inferences about a population based on sample data.

Tests Performed in the Project:

1. T-Test
2. Chi-Square Test

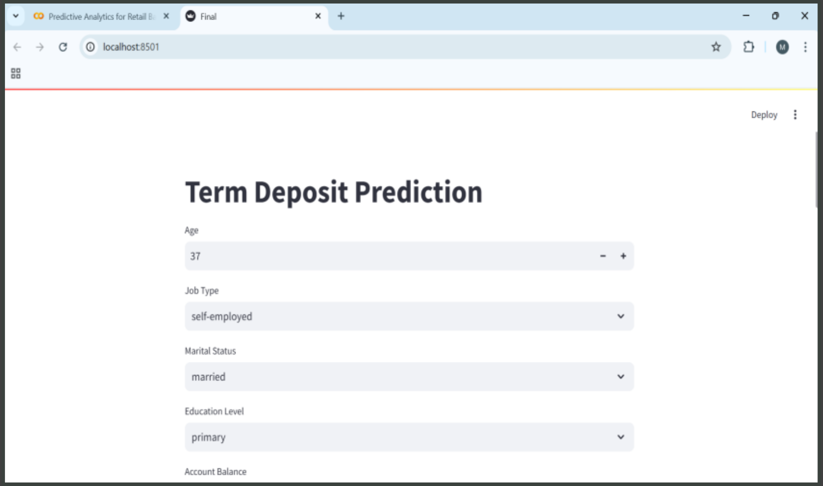
1. T-Test:

- **Definition**: A statistical test to compare the means of two independent groups.
- **Purpose**: To compare the balance of customers who accepted (yes) vs. rejected (no) the term deposit offer.
- **Result**:
  - **T-Test Statistic**: 8.5204
  - **P-Value**: 0.0000
  - **Conclusion**: Reject the null hypothesis, indicating a significant difference in balance between the two groups.
  - **Insight**: Balance significantly affects deposit acceptance.

2. Chi-Square Test:

- **Definition**: A statistical test to check if two categorical variables are independent.
- **Purpose**: To analyze the relationship between categorical features (e.g., job, marital status) and deposit acceptance.
- **Result**:
  - **Chi-Square Statistic**: Computed for each feature.
  - **Conclusion**: Identifies features significantly related to deposit acceptance.
  - **Insight**: Key categorical features influence customer decisions.

SCREENSHOTS



Term Deposit Prediction

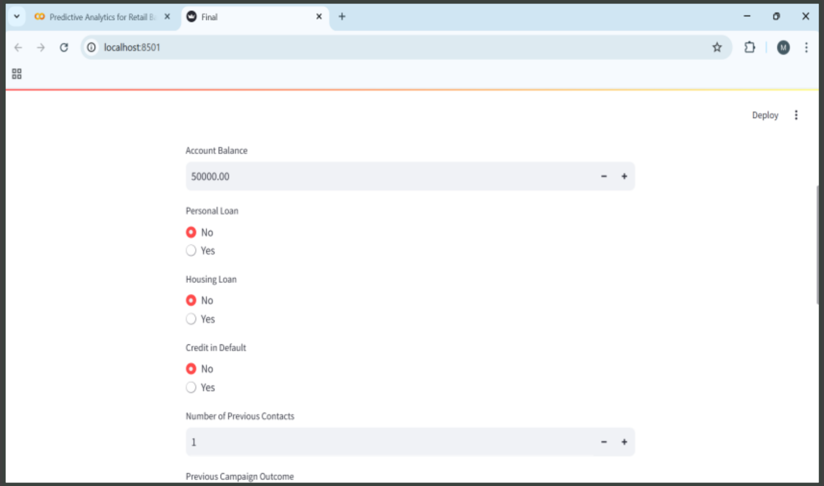
Age
37

Job Type
self-employed

Marital Status
married

Education Level
primary

Account Balance



Account Balance
50000.00

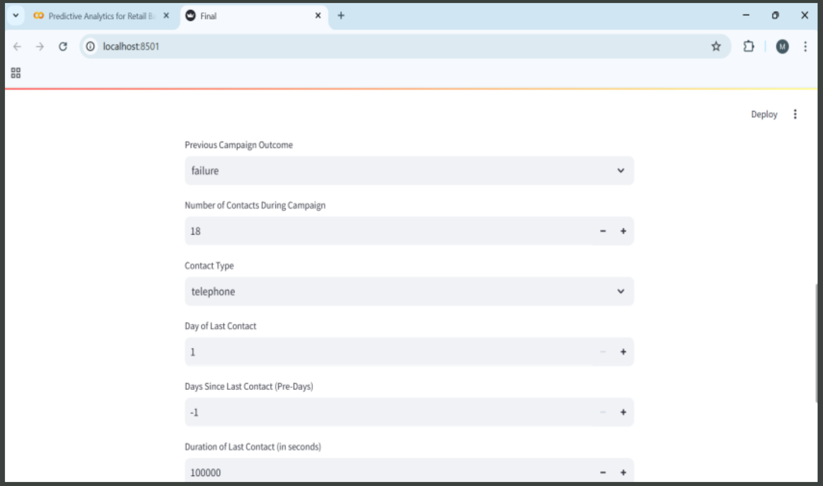
Personal Loan
● No
○ Yes

Housing Loan
● No
○ Yes

Credit in Default
● No
○ Yes

Number of Previous Contacts
1

Previous Campaign Outcome



Previous Campaign Outcome
failure

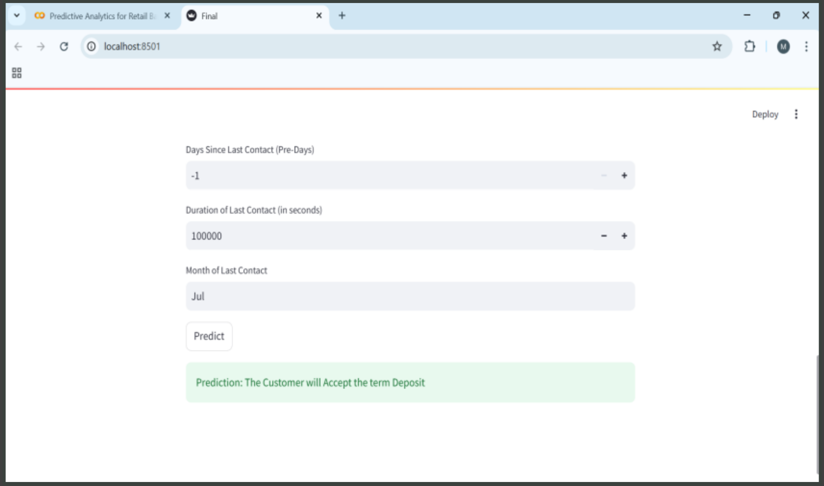
Number of Contacts During Campaign
18

Contact Type
telephone

Day of Last Contact
1

Days Since Last Contact (Pre-Days)
-1

Duration of Last Contact (in seconds)
100000

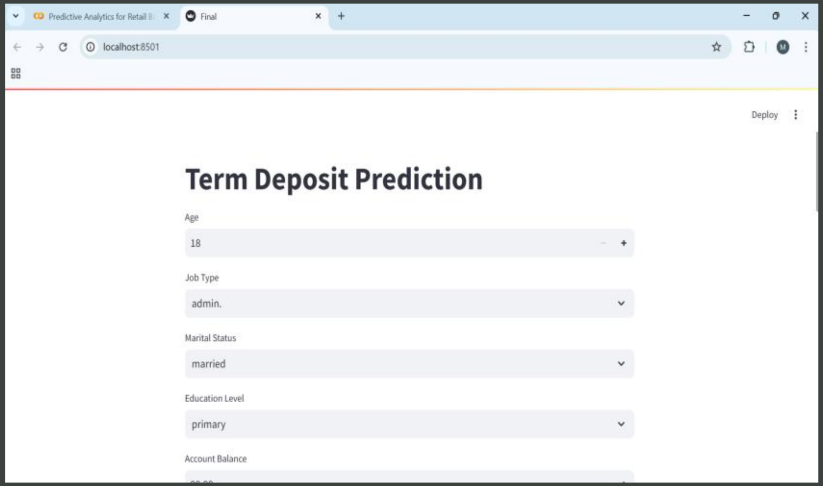


Days Since Last Contact (Pre-Days)
-1

Duration of Last Contact (in seconds)
100000

Month of Last Contact
Jul

Prediction: The Customer will Accept the term Deposit



Term Deposit Prediction

Age

18

Job Type

admin.

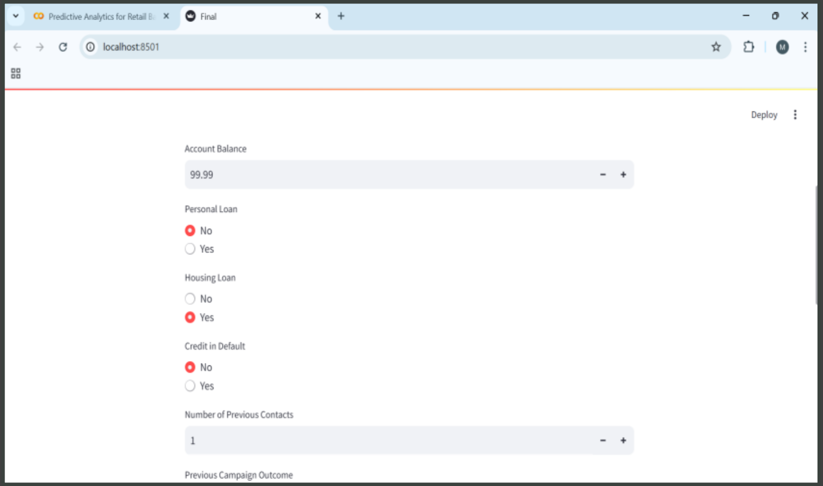
Marital Status

married

Education Level

primary

Account Balance



Account Balance

99.99

Personal Loan

● No
○ Yes

Housing Loan

○ No
● Yes

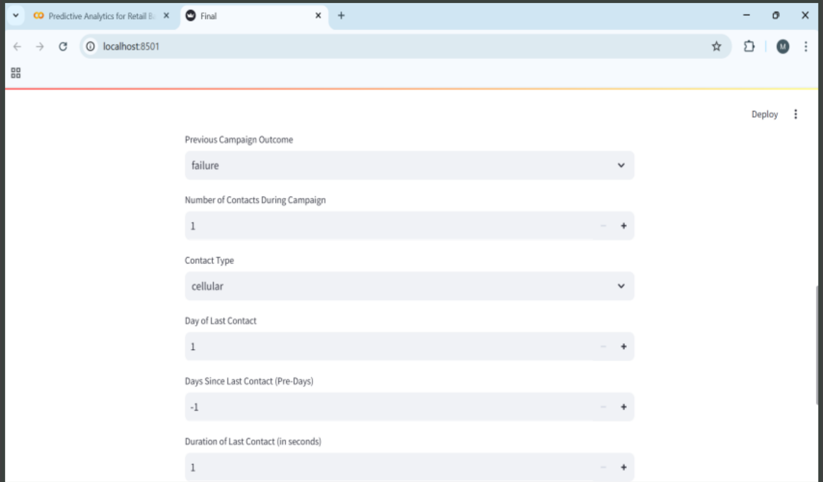
Credit in Default

● No
○ Yes

Number of Previous Contacts

1

Previous Campaign Outcome



Previous Campaign Outcome

failure

Number of Contacts During Campaign

1

Contact Type

cellular

Day of Last Contact

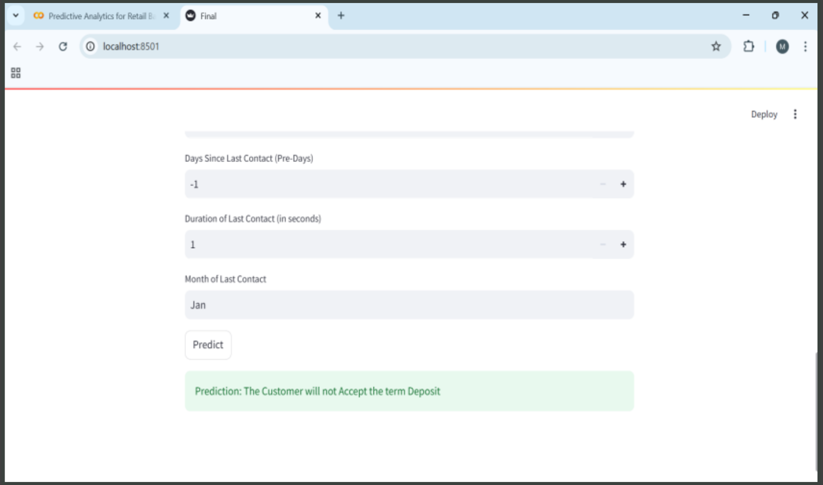
1

Days Since Last Contact (Pre-Days)

-1

Duration of Last Contact (in seconds)

1



Days Since Last Contact (Pre-Days)

-1

Duration of Last Contact (in seconds)

1

Month of Last Contact

Jan

Predict

Prediction: The Customer will not Accept the term Deposit

THANK YOU