EXPERIMENT NO 10

Team Members : Riddesh Sonawane (66), Bhairavi Tipayle (73), Shreyash Waralkar (76) , Yashraj Zope (79)

AIM : Perform Case Study on Credit Card Default Prediction to analyze credit card default patterns using data science techniques, enabling better risk assessment and informed decision-making for financial institutions.

THEORY :

Introduction

Credit card default prediction is essential for financial institutions to minimize losses from unpaid debts. As credit usage increases, accurately assessing default risk becomes crucial for maintaining financial stability. Banks analyze factors such as demographics, transaction history, spending patterns, and repayment behavior to predict defaults.

Traditional risk assessment methods often fail to capture complex relationships in customer data. Machine learning provides a more effective solution by identifying patterns within large datasets. Algorithms like Logistic Regression, Decision Trees, and Random Forests improve prediction accuracy, helping banks make informed decisions.

By leveraging predictive models, financial institutions can proactively identify high-risk customers and implement strategies such as credit limit adjustments and personalized repayment plans to mitigate losses. This project aims to develop a machine learning model for credit default prediction, enhancing risk management and decision-making. Data-driven insights will help banks optimize resource allocation, reduce defaults, and improve customer satisfaction.

Problem Statement

Commercial banks face a growing challenge in managing credit risk due to increasing credit card defaults. Failure to predict and mitigate these defaults can lead to significant financial losses and instability. Traditional risk assessment methods often fail to capture complex patterns in customer behavior, making it difficult for banks to proactively identify high-risk customers. By leveraging machine learning techniques, financial institutions can analyze key factors such as spending habits, transaction histories, and repayment patterns to improve the accuracy of credit default predictions. This study aims to develop a predictive model that enhances risk management strategies, reduces default rates, and enables better decision-making for banks.

Objectives

1. To analyze customer attributes such as demographics, spending behavior, transaction history, and repayment patterns for identifying key risk factors.
2. To preprocess and refine financial datasets through data exploration, cleaning, and feature engineering to ensure model accuracy.
3. To implement machine learning models, including logistic regression, decision trees, and random forests, for predicting credit card default likelihood.
4. To evaluate and compare model performance using accuracy metrics to determine the most effective predictive approach.
5. To provide financial institutions with actionable insights for proactive risk management, such as credit limit adjustments and tailored repayment plans.
6. To enhance customer satisfaction by enabling personalized financial solutions based on predictive analysis.

Process Overview

1. Explore Dataset: Analyze customer attributes like demographics, spending, and repayment patterns to identify trends.
2. Pre-processing: Handle missing values, fix inconsistencies, and normalize data for better accuracy.
3. Map-Reduce Technique: Splits large data for efficient processing and aggregation.
4. Model Evaluation: Test Logistic Regression, KNN, and Random Forest to find the best predictor.
5. Model Selection: Choose the best model based on accuracy, precision, and recall.
6. User Interface: Develop a Flask-based web app for easy user interaction and predictions.

Technology Used

- Python: High-level programming language supporting multiple paradigms.
- NumPy: Handles large arrays and mathematical operations.
- Matplotlib & Seaborn: Data visualization libraries.
- Sklearn: Machine learning library for model building and evaluation.
- Pandas: Data manipulation and analysis tool.
- Random Forest & KNN: Machine learning algorithms for classification and regression.
- Logistic Regression: Model for binary classification.
- Flask: Lightweight web framework for APIs and applications.

Applications

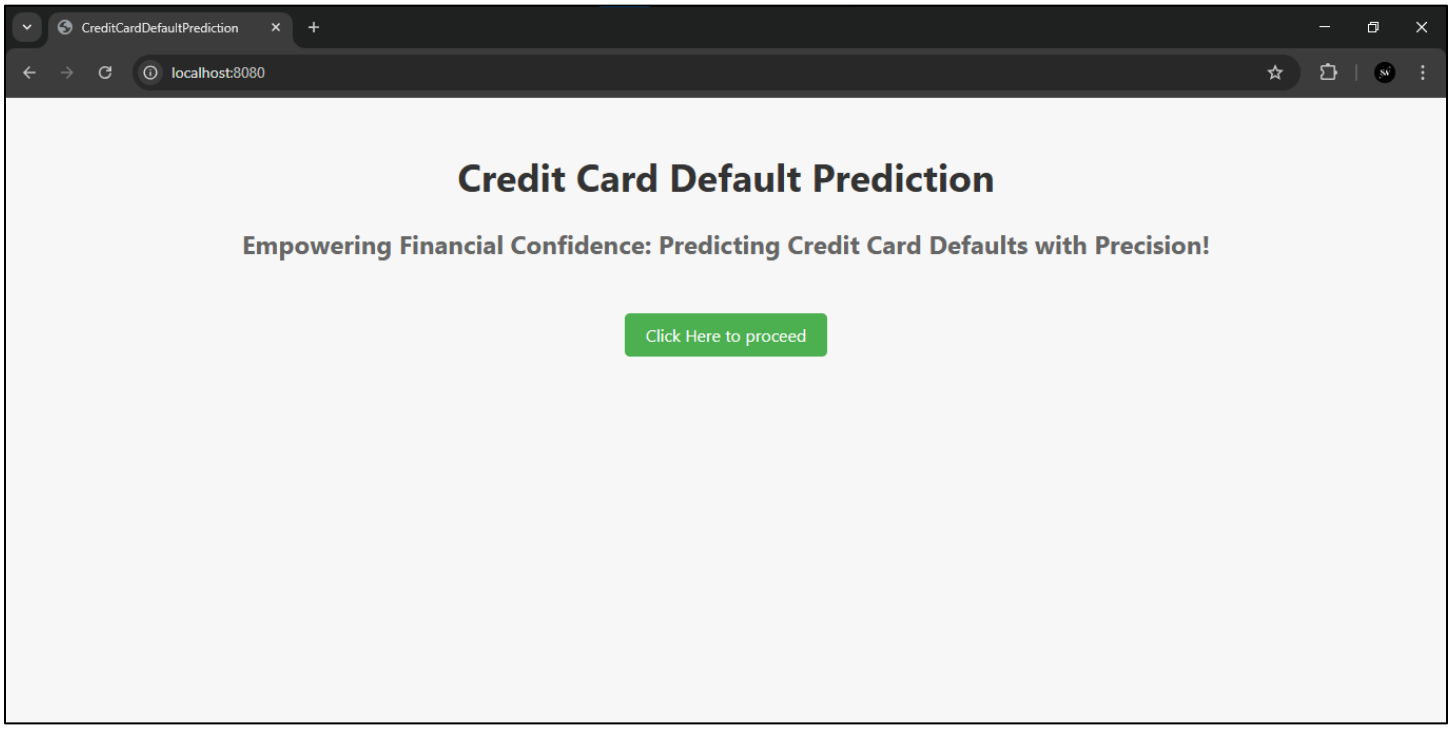
- **Risk Assessment:** Predictive models help banks evaluate credit risk before issuing cards by analyzing spending habits, income, and repayment history.
- **Fraud Detection:** Machine learning identifies suspicious transaction patterns, detecting anomalies in real-time to prevent fraud.
- **Customer Profiling:** Customers are categorized based on repayment behavior, enabling banks to offer personalized credit limits and repayment plans.
- **Loan Approval:** Data-driven models assess an applicant's likelihood of repayment, aiding in informed lending decisions.
- **Financial Planning:** Insights from customer behavior and market trends help optimize credit policies, ensuring profitability and risk minimization.

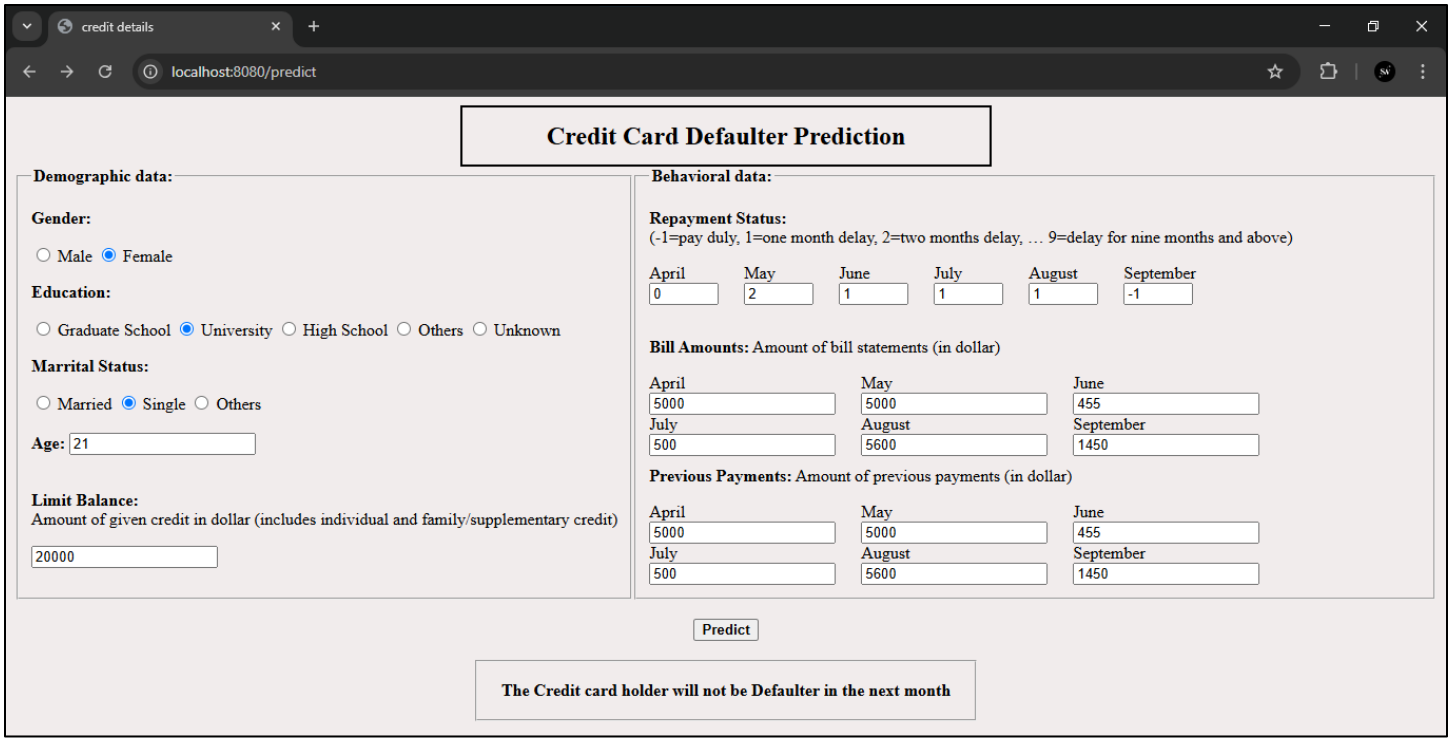
// **CODE**

```
import pandas as pd
import numpy as np
import warnings
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
from sklearn.compose import ColumnTransformer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score

warnings.filterwarnings('ignore')
data = pd.read_csv("./data/UCI_Credit_Card.csv").drop(columns=["ID"])
data.rename(columns={'PAY_0': 'PAY_1', 'default.payment.next.month': 'Default'}, inplace=True)

x, y = data.drop(columns=["Default"]), data[["Default"]]
numerical_columns = x.select_dtypes(exclude="object").columns
num_pipeline = Pipeline(steps=[("imputer", SimpleImputer()), ("scalar", StandardScaler())])
preprocessor = ColumnTransformer([("num_pipeline", num_pipeline, numerical_columns)])

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.40, random_state=42)
x_train = pd.DataFrame(preprocessor.fit_transform(x_train),
columns=preprocessor.get_feature_names_out())
x_test = pd.DataFrame(preprocessor.transform(x_test),
columns=preprocessor.get_feature_names_out())
```

```
def evaluate_model(true, predicted):
    return accuracy_score(true, predicted)

models = {
    "LogisticRegression": LogisticRegression(),
    "RandomForestClassifier": RandomForestClassifier(),
    "SupportVectorMachine": SVC(),
    "KNeighborsClassifier": KNeighborsClassifier()
}
for name, model in models.items():
    model.fit(x_train, y_train)
    acc_score = evaluate_model(y_test, model.predict(x_test))
    print(f"{name}\nAccuracy Score: {acc_score}\n" + "="*35 + "\n")
```

// **OUTPUT :**





CONCLUSION : The credit card default prediction project used machine learning to assess default risk based on customer data, including demographics, spending patterns, and repayment behaviors. Data preprocessing ensured accuracy, while Map-Reduce enabled efficient large-scale processing. Various models, including Logistic Regression, K-Nearest Neighbors (KNN), and Random Forest, were evaluated, with Random Forest achieving the highest accuracy (0.8148). A Flask-based interface allowed users to input data and receive predictions. This project demonstrated the effectiveness of machine learning in improving credit risk management, helping financial institutions identify high-risk customers, reduce defaults, and make informed, data-driven lending decisions.