Jamar\_Sanders\_Understanding\_Business\_Problems

<https://github.com/ajuolpoows123/Understanding-Business-Problems>

### Predicting Default Using Machine Learning

#### Formulating the Questions

Machine learning plays a crucial role in risk assessment for financial institutions. One of the major challenges lenders face is predicting whether an applicant will default on a loan. This problem can be addressed using predictive modeling. Below are the formulated questions for our analysis:

1. Prediction Question:  
   * How accurately can I predict whether a borrower will default on a loan based on their financial and demographic characteristics?
2. Inferential Question:  
   * Which factors (e.g., income, credit score, loan amount) have the most significant effect on the probability of loan default?

The prediction question focuses on developing a machine learning model to classify borrowers as either likely to default or not. The inferential question aims to understand the relative importance of different variables in influencing loan default.

#### Dataset Selection and Origin

For this analysis, I use the "Loan Default Dataset" available on Kaggle (yasserh/loan-default-dataset). This dataset is suitable as it contains over 10,000 records with more than 10 variables, making it ideal for predictive modeling.

##### Ethical Considerations

* The dataset should be anonymized, meaning it does not contain personally identifiable information (PII) such as names or Social Security numbers.
* The data should comply with privacy regulations like GDPR (General Data Protection Regulation) and CCPA (California Consumer Privacy Act) to prevent misuse.
* It is essential to ensure that the dataset is representative and free from bias, as biased datasets can lead to unfair loan decisions, particularly against specific demographics.

#### Exploring the Dataset Variables

The dataset contains various demographic, financial, and loan-related features. Below is a breakdown of these variables:

| Variable Name | Type | Scale | Description |
| --- | --- | --- | --- |
| LoanID | Categorical | Nominal | Unique identifier for each loan. |
| Age | Numerical | Interval | Borrower’s age in years. |
| Income | Numerical | Interval | Monthly income of the borrower. |
| LoanAmount | Numerical | Interval | Amount of the loan requested. |
| CreditScore | Numerical | Interval | Credit score of the borrower (typically ranges 300-850). |
| MonthsEmployed | Numerical | Interval | Number of months the borrower has been employed. |
| NumCreditLines | Numerical | Interval | Total number of active credit lines. |
| InterestRate | Numerical | Interval | Interest rate of the loan in percentage. |
| LoanTerm | Numerical | Interval | Duration of the loan in months or years. |
| DTIRatio | Numerical | Ratio | Debt-to-income ratio (total monthly debt payments / monthly income). |
| Education | Categorical | Ordinal | Borrower's education level (e.g., High School, Bachelor's, Master's, PhD). |
| EmploymentType | Categorical | Nominal | Employment status (e.g., Self-employed, Salaried, Unemployed). |
| MaritalStatus | Categorical | Nominal | Marital status (Single, Married, Divorced, etc.). |
| HasMortgage | Categorical | Nominal | Whether the borrower has an existing mortgage (Yes/No). |
| HasDependents | Categorical | Ordinal | Numbers of dependents (0, 1, 2, etc.). |
| LoanPurpose | Categorical | Nominal | Purpose of the loan (Home purchase, Education, Car, etc.). |
| HasCoSigner | Categorical | Nominal | Whether the loan has a cosigner (Yes/No). |
| Default (Target Variable) | Categorical | Nominal | Whether the borrower defaulted on the loan (1 = Yes, 0 = No). |

#### Dependent and Independent Variables

* Dependent Variable (Target): Default (whether a borrower defaults on a loan).
* Independent Variables (Predictors):  
  + Financial indicators: Income, CreditScore, LoanAmount, InterestRate, DTIRatio.
  + Employment & stability: MonthsEmployed, NumCreditLines, EmploymentType.
  + Demographics: Age, Education, MaritalStatus, HasDependents.
  + Loan-specific details: LoanTerm, LoanPurpose, HasMortgage, HasCoSigner.

#### Importance of This Dataset for Predictive Modeling

1. Large Sample Size:  
   * The dataset contains more than 10,000 observations, ensuring statistical significance.
2. Rich Feature Set:  
   * It has a mix of financial, demographic, and loan-specific attributes, making it ideal for identifying default patterns.
3. Categorical and Numerical Variables:  
   * The presence of both types of variables allows for comprehensive feature engineering.
4. Real-World Relevance:  
   * Predicting loan defaults is crucial for banks, credit unions, and financial institutions to minimize financial risk.

#### Potential Machine Learning Approaches

For prediction (classification of borrowers as likely to default or not), I can use:

* Logistic Regression: A simple yet interpretable model for binary classification.
* Random Forest: Can handle mixed data types and rank variable importance.
* Gradient Boosting (XGBoost, LightGBM): Provides better accuracy in large datasets.
* Neural Networks: Useful for complex relationships between features.

For inferential analysis (understanding the impact of each feature on default probability):

* SHAP (Shapley Additive Explanations): Helps interpret model predictions.
* Feature Importance Analysis: Identifies the strongest predictors of default.

#### Conclusion

This project demonstrates how machine learning can help predict loan defaults and assess risk factors in lending. By building predictive models, financial institutions can make smarter lending decisions, reduce risk, and improve customer screening processes.

Here is a code for predicting loan approval. It won’t work as is but is an example.

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler, LabelEncoder

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

# Load dataset

file\_path = "archive (1).zip/Loan\_default.csv" # 🔹 Update this path

df = pd.read\_csv(file\_path)

# Drop LoanID (not useful for prediction)

df.drop(columns=["LoanID"], inplace=True)

# Encode categorical variables

categorical\_cols = ["Education", "EmploymentType", "MaritalStatus", "LoanPurpose", "HasMortgage", "HasDependents", "HasCoSigner"]

for col in categorical\_cols:

df[col] = LabelEncoder().fit\_transform(df[col])

# Define features (X) and target (y)

X = df.drop(columns=["Default"])

y = df["Default"]

# Split data into training and test sets (80% train, 20% test)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Standardize numerical features

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

# Train Random Forest Classifier

model = RandomForestClassifier(n\_estimators=100, random\_state=42)

model.fit(X\_train, y\_train)

# Predictions

y\_pred = model.predict(X\_test)

# Model Evaluation

accuracy = accuracy\_score(y\_test, y\_pred)

print("Model Accuracy:", accuracy)

print("\nClassification Report:\n", classification\_report(y\_test, y\_pred))

# Confusion Matrix Visualization

plt.figure(figsize=(6,4))

sns.heatmap(confusion\_matrix(y\_test, y\_pred), annot=True, fmt='d', cmap="Blues", xticklabels=["No Default", "Default"], yticklabels=["No Default", "Default"])

plt.xlabel("Predicted")

plt.ylabel("Actual")

plt.title("Confusion Matrix")

plt.show()

# Predict a new borrower's default status

new\_borrower = np.array([[35, 5000, 20000, 720, 36, 3, 7.5, 60, 25, 2, 1, 0, 1, 1, 0, 0]])

new\_borrower = scaler.transform(new\_borrower) # Standardize

prediction = model.predict(new\_borrower)

print("Loan Default Prediction (1=Default, 0=No Default):", prediction[0])

**References**:

Chatterjee, S., & Simonoff, J. (2020). Handbook of regression analysis with applications in R (2nd ed.). Wiley and Sons, Inc.

Chua, W. (n.d.). Chapter 11: Multinomial logistic regression. In Companion to BER 642: Advanced regression methods. Retrieved January 31, 2025, from<https://bookdown.org/chua/ber642_advanced_regression/multinomial-logistic-regression.html>

Larose C. D., & Larose D. T. (2019). Data science using Python and R. ﻿John Wiley & Sons, Inc. ﻿﻿ISBN-13: 9781119526841