
KIET
GROUP OF INSTITUTIONS
*Connecting Life with Learning*


# REPORT
# on
# "PREDICT LOAN DEFAULT"

NAME - SHREYA TYAGI

ROLL.NO – 20240110030024 1

BRANCH – CSE(AI) – 'D'

SUBMITTED TO – ABHISHEK SHUKLA

# INTRODUCTION

Financial institutions such as banks and credit agencies face significant risks when issuing loans. One of the primary concerns is **loan default**, where a borrower fails to meet the legal obligations of the loan repayment. Defaulting impacts both the lender's financial health and the broader economy. Therefore, accurately predicting loan default has become a critical part of **risk assessment and credit evaluation**.

Traditionally, loan approvals were based on static rules—like credit scores, income thresholds, and employment history. However, with the rise of **machine learning (ML)** and **data-driven decision making**, we can now go beyond rule-based systems and uncover hidden patterns in large datasets. By using historical data of borrowers, ML models can learn which combinations of features (like low income, high loan amount, or no credit history) are more likely to result in defaults.

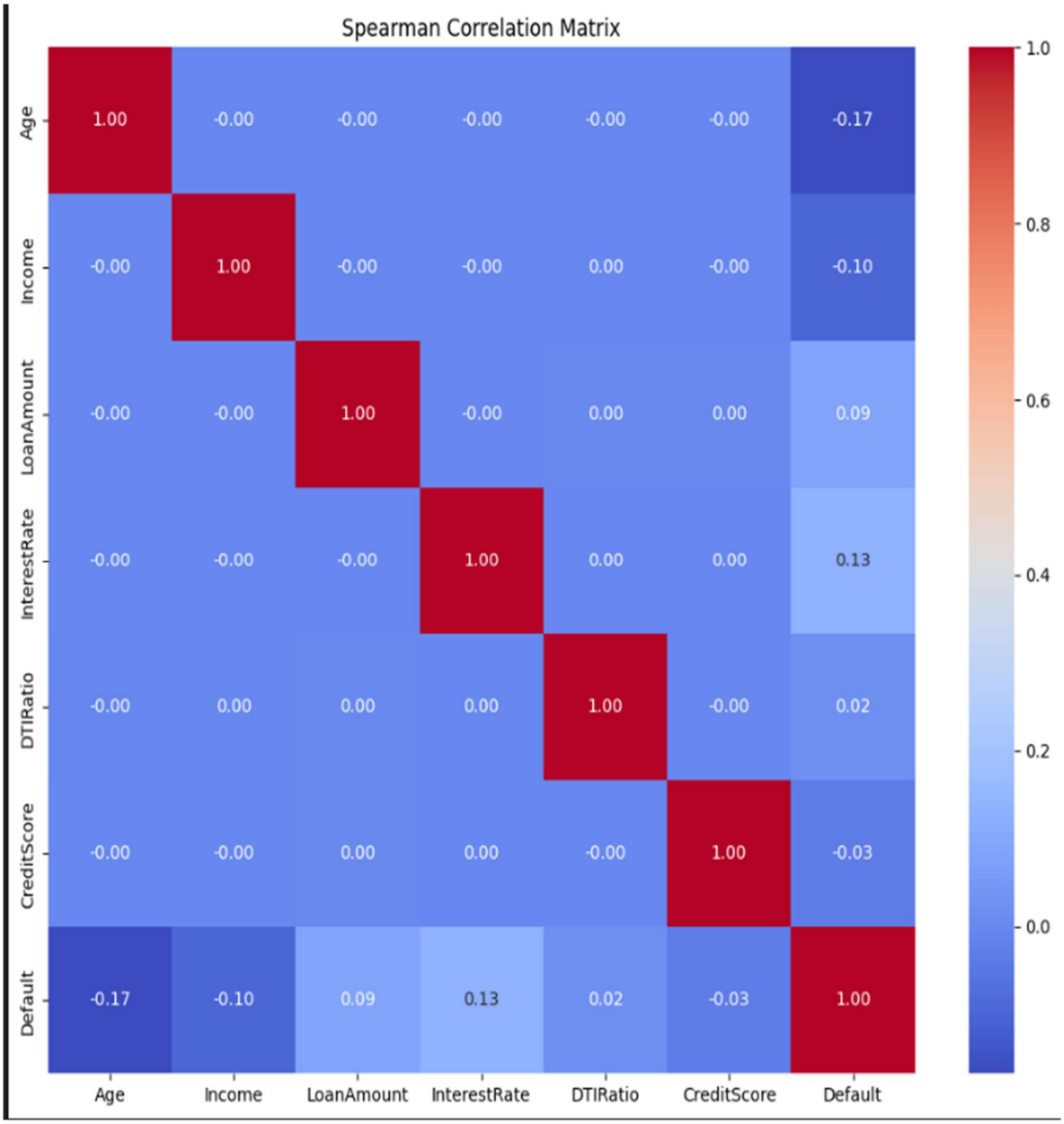
# METHODOLOGY

1. **Data Collection**:
   A loan dataset was sourced from Kaggle or other public repositories. It contains features like credit history, income, loan amount, employment status, etc.
2. **Data Preprocessing**:
    1. Handle missing values.
    2. Convert categorical variables using one-hot encoding or label encoding.
    3. Normalize/standardize numerical features.
    4. Split into training and test sets.
3. **Model Selection**:
    1. Logistic Regression (baseline model).
    2. Decision Tree / Random Forest (for comparison).
4. **Model Evaluation**:
    1. Accuracy, Precision, Recall, F1-Score.
    2. Confusion matrix for performance analysis.

# CODE

```python
# Step 1: Import libraries
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix

# Step 2: Load your dataset
# Replace 'your_dataset.csv' with the path to your dataset
data = pd.read_csv('Predict Loan Default.csv')

# Step 3: Preprocessing
# Assume features like: income, age, loan_amount, credit_score, etc.
# Target column: 'loan_default'
X = data[['income', 'age', 'loan_amount', 'credit_score']]
y = data['loan_default']

# Handle missing values if needed
X = X.fillna(X.mean())

# Normalize features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Step 4: Split dataset into train and test sets
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)

# Step 5: Train a Logistic Regression model
model = LogisticRegression()
model.fit(X_train, y_train)

# Step 6: Make predictions
y_pred = model.predict(X_test)

# Step 7: Evaluate the model
print("Accuracy:", accuracy_score(y_test, y_pred))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))
```

# OUTPUT


Spearman Correlation Matrix
Age
Income
LoanAmount
InterestRate
DTIRatio
CreditScore
Default
1.00 -0.00 -0.00 -0.00 -0.00 -0.00 -0.17
-0.00 1.00 -0.00 -0.00 0.00 -0.00 -0.10
-0.00 -0.00 1.00 -0.00 0.00 0.00 0.09
-0.00 -0.00 -0.00 1.00 0.00 0.00 0.13
-0.00 0.00 0.00 0.00 1.00 -0.00 0.02
-0.00 -0.00 0.00 0.00 -0.00 1.00 -0.03
-0.17 -0.10 0.09 0.13 0.02 -0.03 1.00
Age
Income
LoanAmount
InterestRate
DTIRatio
CreditScore
Default
1.0
0.8
0.6
0.4
0.2
0.0

# REFERENCE


- Dataset: Loan Prediction Dataset – Kaggle
- Scikit-learn Documentation: https://scikit-learn.org/
- Logistic Regression: https://en.wikipedia.org/wiki/Logistic_regression
- Pandas Documentation: https://pandas.pydata.org/