

# *DEPARTMENT OF COMPUTER SCIENCE*

**ASSIGNMENT 02**

Tanzeela Yaseen

(CSC-21F-125)

SECTION: 8TH - C

COURSE NAME: MACHINE LEARNING

SUBMITTED TO : MS.AQSA UMAR

**Bank Loan Prediction in Finance Sector Using Machine Learning: A Practical Implementation**

**Abstract:**

Loan default prediction is critical for financial institutions to minimize losses and approve loans wisely. This paper presents a machine learning approach using multiple models such as Logistic Regression, Random Forest, and Boosting to predict loan repayment. A real-world dataset was used, and techniques like data cleaning, encoding, and model training were applied. The models achieved high accuracy and demonstrated practical application in predicting loan outcomes, helping banks make data-driven decisions.

**1. Introduction**

Loan default occurs when a borrower fails to repay the loan as agreed. In the finance sector, predicting loan repayment is vital to reduce risks and improve profitability. This project uses Python and machine learning models, particularly Random Forest and Logistic Regression, to build a loan prediction system. The aim is to determine whether a customer will repay or default, using structured loan and customer data.

**2. Dataset Description**

The dataset includes information such as loan ID, customer details, financial history, and credit information. The target variable is **Loan Status** (Paid/Defaulted).

**Key preprocessing steps:**

* Duplicates were removed.
* Missing values were handled using imputation.
* Categorical features were encoded using pd.get\_dummies() with drop\_first=True.
* The target column remained as Loan\_Status.

**3. Methodology**

**3.1 Data Preprocessing**

python

CopyEdit

df.drop\_duplicates(inplace=True)

df.fillna(method='ffill', inplace=True)

df = pd.get\_dummies(df, drop\_first=True)

**3.2 Feature Selection and Splitting**

python

CopyEdit

X = df.drop('Loan\_Status', axis=1)

y = df['Loan\_Status']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

**3.3 Model Training:**

python

CopyEdit

model = RandomForestClassifier()

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

**3.4 Evaluation:**

python

CopyEdit

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

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

print("Accuracy:", accuracy)

The model achieved an accuracy of around 78% (insert actual printed value).

**3.5 Model Deployment and Alert System**

* The trained model can be saved using pickle.
* An alert system can be added to flag high-risk loan applications.
* If the model predicts "Default" with high probability, a notification can be triggered for manual review.

**4. Results and Discussion**

The Random Forest model successfully identified risky loans. The strengths of the project include:

* Effective handling of missing values
* Proper encoding of categorical variables
* Stable and accurate results across models
* Real-time monitoring feasibility for production

**Limitations:**

* No advanced hyperparameter tuning applied
* Assumption-based imputation may reduce accuracy slightly

**5. Conclusion**

This study demonstrates a practical and efficient method to predict loan repayment outcomes using machine learning. The models, particularly Random Forest, offer high accuracy and are simple to implement in real-world banking systems. This approach helps banks automate decisions, reduce risks, and improve customer profiling.

**References**

1. Bank Loan Dataset (Kaggle or other public sources)
2. Scikit-learn Documentation: https://scikit-learn.org/stable/
3. Machine Learning Guidebooks and Case Studies