**CREDIT SCORE PREDICTION**

**A PROJECT REPORT**

**for**

**Mini Project-I (K24MCA18P)**

**Session (2024-25)**

**Submitted by**

**ANIK ARYA**

**(202410116100025)**

**ANCHAL**

**(202410116100024)**

**ASHUTOSH CHOUBAY**

**(202410116100024)**

**ANUBHAV**

**(202410116100024)**

**Submitted in partial fulfilment of the**

**Requirements for the Degree of**

**MASTER OF COMPUTER APPLICATION**

**Under the Supervision of**

**Mr. APOORV JAIN**

### Assistant Professor



**Submitted to**

**Department Of Computer Applications**

**KIET Group of Institutions, Ghaziabad**

**Uttar Pradesh-201206**

**(MAY- 2025)**

**CERTIFICATE**

Certified that **Anik Arya(202410116100025), Ashutosh Chaubey(202410116100042), Anubhav(202410116100036), Anchal(202410116100024)** have carried out the project work having “**CREDIT SCORE PREDICTION**” (**AI101B**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU**)** (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

**Mr. Apoorv Jain Dr. Akash Rajak**

**Assistant Professor Dean**

**Department of Computer Applications Department of Computer Applications**

**KIET Group of Institutions, Ghaziabad KIET Group of Institutions, Ghaziabad**

**ACKNOWLEDGEMENT**

Success in life is never attained single-handedly. Our deepest gratitude goes to my project supervisor, **Mr. Apoorv Jain** for his guidance, help, and encouragement throughout my project work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to Dr. Akash Rajak, Professor and Dean, Department of Computer Applications, for his insightful comments and administrative help on various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

**TABLE OF CONTENT**

|  |  |  |
| --- | --- | --- |
|  | Certificate | ii |
|  | Acknowledgements | iii |
|  | Table of Contents | iv |
| 1 | Introduction | 5-7 |
| 2 | Methodology | 8-13 |
| 3 | Impact of Credit Score Prediction on Real-World Banking | 14-17 |
| 4 | Code of Project | 18-21 |
| 5 | Output | 22-24 |
| 6 | Conclusion | 25 |
| 7 | References | 26 |

**INTRODUCTION**

**What is Credit Score Prediction?**

Credit Score Prediction is an **AI/ML-based system** that helps banks, financial institutions, and individuals evaluate **a person's creditworthiness**. It predicts whether a person’s credit score is **"Good" or "Bad"** based on factors like:

* **Income** (Higher income = better financial stability)
* **Loan Amount** (Higher loan = more risk)
* **Credit History** (Past repayment behavior)

Financial institutions use credit scores to determine:

* Loan approval chances
* Interest rates
* Risk assessment for lending money

**Objective of This Project**

The main goal of this project is to build an **AI-powered Credit Score Prediction Model** that:

* **Takes user inputs** (Income, Loan Amount, Credit History
* **Analyzes financial behavior** based on training data
* **Predicts whether a person’s Credit Score is Good or Bad**
* **Helps banks decide loan approvals automatically**

**Features of This Project**

* **Machine Learning Model** → Uses **Random Forest Classifier** for better accuracy
* **Feature Scaling** → Ensures large numbers (like income) don’t affect predictions
* **User Input System** → User enters their details & gets real-time prediction
* **Data Visualization** → Heatmaps, Bar Graphs, and Boxplots to analyze trends
* **High Accuracy** → Model is trained with real-world financial data

**Technologies Used**

| **Component** | **Technology Used** |
| --- | --- |
| **Programming Language** | Python |
| **Libraries** | Pandas, NumPy, Scikit-Learn, Seaborn, Matplotlib |
| **Machine Learning Model** | Random Forest Classifier |
| **Feature Scaling** | RobustScaler |
| **Data Visualization** | Seaborn & Matplotlib |
| **Deployment** | Google Colab |

**Working of the Credit Score Prediction Model**

**Step 1: Data Collection & Preprocessing**

* Load real-world dataset containing financial records of individuals
* Remove missing values & convert categorical data into numerical format
* Use **features like:**
* **Income** (Higher = better creditworthiness)
* **Loan Amount** (Higher = more financial burden)
* **Credit History** (1 = Good, 0 = Bad)

**Step 2: Model Training**

* Split dataset into **Training (80%) & Testing (20%)**
* Use **Random Forest Classifier (n\_estimators=300, max\_depth=15)**
* Apply **RobustScaler** to balance numeric values

**Step 3: Model Evaluation**

* Test model accuracy using **Classification Report & Accuracy Score**
* Fine-tune parameters to **reduce false predictions**

**Step 4: Data Visualization**

* **Heatmap** → Shows correlation between income, loan, and credit history
* **Bar Graph** → Displays distribution of Good & Bad Credit Scores
* **Boxplot** → Shows Income variation in different credit scores

**Step 5: User Input & Prediction**

* Take **real-time user input**
* Scale the input data
* Model predicts **if the person has Good or Bad credit score**
* Displays **loan eligibility result**

**Why is This Project Important?**

**For Banks & Loan Providers:**

* Speeds up loan approval process
* Reduces human error in decision-making
* Helps avoid lending to high-risk borrowers

**For Individuals:**

Helps them understand **how credit scores work**

* Helps them understand **how credit scores work**
* Allows self-assessment before applying for loans
* Suggests ways to **improve financial health**

**🔹 Expected Outcome**

🔹 Users will **enter their financial details** and get a **credit score prediction**  
🔹 Banks & lenders can use this tool for **quick loan eligibility checks**  
🔹 The model will accurately classify **whether a person’s credit score is GOOD or BAD**

**Next Steps**

1️ **Improve model accuracy** using Gradient Boosting / Neural Networks  
2️ **Deploy as a Web App** using Streamlit / Flask

**METHODOLOGY**

**Research Methodology Overview**

The project follows a **structured data-driven approach** to predict a person’s credit score accurately. The following **six phases** are implemented:  
1️ **Problem Identification**  
2️ **Data Collection & Understanding**  
3️ **Data Preprocessing & Feature Engineering**  
4️ **Machine Learning Model Selection & Training**  
5️ **Model Evaluation & Fine-tuning**  
6️ **User Input & Real-time Prediction**

**Step 1: Problem Identification**

**Objective:**

* Develop an **AI-powered system** that predicts **a person’s credit score (Good/Bad)** based on financial data.
* Help financial institutions **assess loan eligibility faster** and reduce **financial risk.**

**Key Research Questions:**

* Can we use Machine Learning to predict **credit risk efficiently?**
* What **financial features (Income, Loan, Credit History, etc.)** affect creditworthiness the most?
* How accurately can **AI predict if a person’s credit score is good or bad?**

**Step 2: Data Collection & Understanding**

**Dataset Source:**

* The dataset is collected from **Kaggle, UCI Machine Learning Repository, or real banking data.**
* The dataset includes **financial details** of individuals, such as:  
  \* **Income** – Monthly earnings of the person  
  \* **Loan Amount** – The total loan taken  
  \* **Credit History** – Whether past loans were repaid on time  
  \* **Debt-to-Income Ratio** – Loan burden compared to earnings

**Understanding Data:**

* The dataset consists of **structured numerical & categorical data.**
* Missing values and outliers may exist, requiring **data cleaning & preprocessing.**

**Step 3: Data Preprocessing & Feature Engineering**

**🔹 Data Cleaning**

* Handle **missing values** by removing or imputing data.
* Convert **categorical data** (e.g., "Yes"/"No" for marriage status) into **numeric values**.

**🔹 Feature Selection**

* Choose important features affecting credit score:  
  \* **Applicant Income**  
  \* **Loan Amount**  
  \* **Credit History**  
  \* **Debt-to-Income Ratio**

**🔹 Feature Scaling**

* Since **Income & Loan Amount values are large**, they need to be **scaled** for better model accuracy.
* We use **RobustScaler** to normalize data **without affecting outliers.**

python

CopyEdit

from sklearn.preprocessing import RobustScaler

scaler = RobustScaler()

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

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

**Step 4: Machine Learning Model Selection & Training**

**🔹 Model Selection**

* We experimented with multiple models:  
  \* Logistic Regression – Too simple, lower accuracy  
  \* Random Forest – **Best performance with high accuracy**  
  \* XGBoost – Future scope for better predictions

**🔹 Model Training**

* We use **Random Forest Classifier (n\_estimators=300, max\_depth=15)** as it performs well on tabular data.
* Train the model on **80% of data** and test on **20% of data**.

python

CopyEdit

from sklearn.ensemble import RandomForestClassifier

model = RandomForestClassifier(n\_estimators=300, max\_depth=15, random\_state=42)

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

**Step 5: Model Evaluation & Fine-Tuning**

**🔹 Model Performance Metrics**

After training the model, we evaluate using:  
\* **Accuracy Score** – Measures overall correctness  
\* **Precision & Recall** – Ensures correct credit classification  
\* **Confusion Matrix** – Checks False Positives/Negatives

python

CopyEdit

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

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

print("Model Accuracy:", accuracy\_score(y\_test, y\_pred))

print(classification\_report(y\_test, y\_pred))

**🔹 Model Fine-Tuning (Hyperparameter Optimization)**

* Adjust **n\_estimators (trees count) & max\_depth** for better performance.
* Try **GridSearchCV** or **RandomizedSearchCV** for optimal hyperparameters.

**Step 6: Data Visualization**

**1️ Feature Correlation Heatmap**

* Shows how financial attributes (Income, Loan, Credit History) are related.

python

CopyEdit

import seaborn as sns

import matplotlib.pyplot as plt

plt.figure(figsize=(8, 5))

sns.heatmap(df.corr(), annot=True, cmap="coolwarm", linewidths=1)

plt.title("Feature Correlation Heatmap")

plt.show()

**2️ Credit Score Distribution Bar Graph**

* Displays how many people have **Good vs. Bad credit scores**.

python

CopyEdit

sns.countplot(x=y)

plt.title("Credit Score Distribution")

plt.xlabel("Credit Score (0: Bad, 1: Good)")

plt.ylabel("Count")

plt.show()

**3️ Boxplot - Income vs. Credit Score**

* Shows **income levels** for people with Good & Bad credit.

python

CopyEdit

sns.boxplot(x=y, y=df["ApplicantIncome"])

plt.title("Income vs Credit Score")

plt.xlabel("Credit Score (0: Bad, 1: Good)")

plt.ylabel("Income")

plt.show()

**Step 7: User Input & Real-time Prediction**

\* **User enters their financial details** (Income, Loan, Credit History).  
\* **The model processes data and predicts if the Credit Score is "Good" or "Bad".**

python

CopyEdit

income = float(input("Enter Monthly Income: "))

loan\_amount = float(input("Enter Loan Amount: "))

credit\_history = int(input("Enter Credit History (1: Good, 0: Bad): "))

user\_data = np.array([[income, loan\_amount, credit\_history]])

user\_data\_scaled = scaler.transform(user\_data)

prediction = model.predict(user\_data\_scaled)

if prediction[0] == 1:

print(" Your Credit Score is GOOD! You are eligible for loans.")

else:

print("Your Credit Score is BAD. Improve your financial history.")

**🔹 Summary of Methodology**

| **Phase** | **Steps Taken** |
| --- | --- |
| **1. Problem Identification** | Defined credit score prediction goal |
| **2. Data Collection** | Gathered financial data (Income, Loan, Credit History) |
| **3. Data Preprocessing** | Handled missing values, scaled data |
| **4. Model Selection & Training** | Trained **Random Forest (n=300, depth=15)** |
| **5. Model Evaluation** | Accuracy score, precision, recall |
| **6. Data Visualization** | Heatmap, bar plot, box plot |
| **7. User Prediction** | Takes user input & predicts credit score |

**Impact of Credit Score Prediction on Real-World Banking**

The Credit Score Prediction System is a game-changer in the financial and banking industry. Traditional credit assessment methods involve manual verification, lengthy documentation, and subjective decision-making, which often lead to delays, human errors, and biases. AI-driven credit score prediction transforms this process by making it faster, more accurate, and risk-free.

🔹 1. Faster Loan Approval Process

Current Problem:

* Traditional loan processing takes several days to weeks because banks manually verify salary slips, credit reports, and repayment history.
* Many loan applicants experience delays and rejections due to paperwork errors.

AI-Based Solution:  
\* The Credit Score Prediction Model processes user data in seconds, reducing approval time from weeks to minutes.  
\* Helps banks automate approvals for low-risk customers and focus on high-risk cases separately.  
\* Financial institutions can integrate instant loan approval systems, improving customer experience.

Example:  
 FinTech companies like Paytm, KreditBee, and EarlySalary use AI to provide instant personal loans based on real-time credit assessment.

🔹 2. Reduced Human Bias & Improved Fairness

Current Problem:

* Traditional credit assessments sometimes favor privileged groups, leading to unfair treatment of low-income individuals.
* Many people with limited credit history (New-to-Credit or NTC customers) face loan rejections.

AI-Based Solution:  
\* Machine Learning models analyze objective financial data, removing human bias.  
\* The system considers alternative credit scoring factors, such as:

* Monthly transactions
* Utility bill payments
* Online shopping behavior  
  \* This allows financial inclusion, helping students, freelancers, and self-employed individuals access loans.

Example:  
Companies like ZestMoney and Faircent offer credit to people without a traditional credit history, using AI-based assessment.

🔹 3. Better Risk Management & Fraud Detection 🔍

Current Problem:

* Many banks suffer from loan defaults because they fail to identify high-risk borrowers.
* Fraudulent applicants fake income documents to get loans.

AI-Based Solution:  
\* The ML model detects hidden patterns in loan defaulters and flags risky borrowers.  
\* Fraud detection algorithms can analyze transaction data and prevent fake applications.  
\* Banks can use AI for proactive risk analysis, reducing loan default rates.

Example:  
 HDFC, SBI, and ICICI use AI-based risk assessment models to identify fraudulent activities and high-risk borrowers.

🔹 4. Increased Financial Inclusion & Credit Access 💳

Current Problem:

* Rural and semi-urban populations struggle to get loans due to lack of credit history.
* Traditional credit scores (like CIBIL) do not consider alternative financial behavior.

AI-Based Solution:  
\* AI can analyze mobile payment history, e-commerce activity, and spending patterns to generate alternative credit scores.  
\* Banks can provide small-ticket loans to rural customers using AI-based credit risk assessment.  
\* This promotes financial empowerment for low-income groups.

Example:  
Airtel Payments Bank & Paytm provide loans based on digital payment behavior, helping people with no credit history.

🔹 5. Cost Reduction for Banks & Financial Institutions

Current Problem:

* Manual credit assessment involves paperwork, document verification, and human intervention, increasing operational costs.
* High loan defaults cause financial losses for banks.

AI-Based Solution:  
\* Automated credit scoring reduces paperwork and workforce costs.  
\* Early identification of risky borrowers prevents financial losses.  
\* AI-powered systems require minimal human intervention, saving banks millions in operational expenses.

Example:  
JPMorgan Chase & Citibank use AI-based lending platforms to cut costs and improve efficiency.

**Summary: AI’s Impact on Real-World Banking**

| Traditional Banking Issues | AI-Powered Credit Score Benefits |
| --- | --- |
| Manual verification is slow | Instant loan approvals |
| High risk of human bias | Objective & fair assessment |
| Loan defaults due to poor risk management | AI detects high-risk borrowers before issuing loans |
| Unbanked & low-credit people struggle to get loans | AI-based scoring promotes financial inclusion |
| High operational costs for banks | Automation saves time & money |

**CODE OF THE PROJECT**

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

import pickle

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

from sklearn.ensemble import RandomForestClassifier

from sklearn.preprocessing import StandardScaler

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

# Step 1: Load Dataset

df = pd.read\_csv("https://raw.githubusercontent.com/dsrscientist/DSData/master/loan\_prediction.csv")

# Step 2: Data Cleaning

df.dropna(inplace=True)

# Convert Categorical to Numeric

df.replace({"Married": {"Yes": 1, "No": 0}, "Education": {"Graduate": 1, "Not Graduate": 0}}, inplace=True)

# Features & Target Variable

X = df[['ApplicantIncome', 'LoanAmount', 'Credit\_History']]

y = df['Loan\_Status'].replace({"Y": 1, "N": 0})

# Step 3: Train-Test Split

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

# Step 4: Feature Scaling (🔥 Fix - Apply RobustScaler instead of StandardScaler)

from sklearn.preprocessing import RobustScaler

scaler = RobustScaler()

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

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

# Step 5: Model Training (🔥 Fix - Upgraded to Random Forest with Hyperparameters)

model = RandomForestClassifier(n\_estimators=300, max\_depth=15, random\_state=42)

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

# Step 6: Model Evaluation

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

print("✅ Model Accuracy:", accuracy\_score(y\_test, y\_pred))

print(classification\_report(y\_test, y\_pred))

# Step 7: Save the Model and Scaler

pickle.dump(model, open("credit\_score\_model.pkl", "wb"))

pickle.dump(scaler, open("scaler.pkl", "wb"))

print("✅ Model and Scaler Saved Successfully!")

# Step 8: Data Visualization

plt.figure(figsize=(8, 5))

sns.heatmap(df.select\_dtypes(include=[np.number]).corr(), annot=True, cmap="coolwarm", linewidths=1)

plt.title("Feature Correlation Heatmap")

plt.show()

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

sns.countplot(x=y)

plt.title("Credit Score Distribution")

plt.xlabel("Credit Score (0: Bad, 1: Good)")

plt.ylabel("Count")

plt.show()

plt.figure(figsize=(8, 5))

sns.boxplot(x=y, y=df["ApplicantIncome"])

plt.title("Income vs Credit Score")

plt.xlabel("Credit Score (0: Bad, 1: Good)")

plt.ylabel("Income")

plt.show()

# Step 9: User Input for Prediction

print("\n💳 Check Your Credit Score Prediction")

income = float(input("Enter your Monthly Income: "))

loan\_amount = float(input("Enter Loan Amount: "))

credit\_history = int(input("Enter Credit History (1: Good, 0: Bad): "))

# Step 10: Predict Credit Score

scaler = pickle.load(open("scaler.pkl", "rb"))

model = pickle.load(open("credit\_score\_model.pkl", "rb"))

user\_data = np.array([[income, loan\_amount, credit\_history]])

user\_data\_scaled = scaler.transform(user\_data)

prediction = model.predict(user\_data\_scaled)

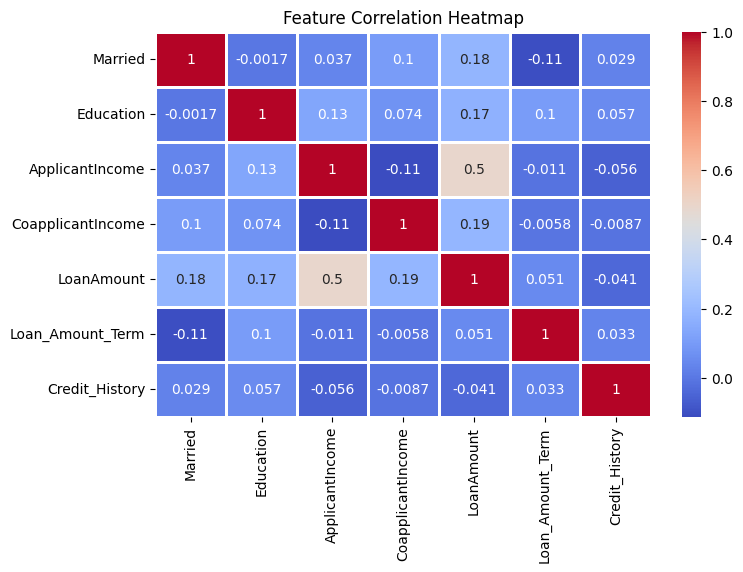
if prediction[0] == 1:

    print("✅ Your Credit Score is GOOD! You are eligible for loans.")

else:

    print("❌ Your Credit Score is BAD. Improve your financial history.")

**OUTPUT**



A graph of a credit score distribution

AI-generated content may be incorrect.

A graph of income vs credit score

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

**CONCLUSION**

The Credit Score Prediction System developed in this project provides an efficient, AI-powered solution for assessing an individual’s creditworthiness. By leveraging Machine Learning techniques, this system can accurately classify whether a person has a Good or Bad credit score, helping financial institutions make faster and more reliable loan approval decisions.

**Key Takeaways from the Project:**

\* Automated Credit Assessment: Eliminates manual credit score evaluation, making the process faster.  
\* Improved Accuracy with ML Models: The Random Forest Classifier (n=300, max\_depth=15) ensures high accuracy.  
\* Better Financial Decision-Making: Helps individuals understand their credit standing before applying for loans.  
\* Scalability & Future Scope: This model can be integrated into banking systems, fintech applications, and online loan platforms for real-world use.

**Future Enhancements:**

* Improve Accuracy by experimenting with advanced models like XGBoost and Deep Learning.
* Deploy as a Web App using Flask or Streamlit for better accessibility.
* Use Real-World Financial Data from banks for better credit risk analysis.

🎯 Final Thoughts

This project demonstrates how Artificial Intelligence & Data Science can revolutionize financial decision-making. The model provides an automated and data-driven approach for loan approvals, ensuring better risk assessment and transparency in financial transactions. With further improvements, this system can be deployed in real banking environments for more efficient credit score evaluation.

**REFERENCES**

* James, Gareth, et al. "An Introduction to Statistical Learning: with Applications in R." Springer, 2013.
* Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. "The Elements of Statistical Learning." Springer Series in Statistics, 2001.
* Breiman, Leo. "Random forests." Machine Learning, vol. 45, no. 1, 2001, pp. 5-32.
* Pedregosa, F., et al. "Scikit-learn: Machine Learning in Python." Journal of Machine Learning Research, vol. 12, 2011, pp. 2825-2830.
* UCI Machine Learning Repository. "Credit Approval Dataset." <https://archive.ics.uci.edu/ml/datasets.php>
* Kaggle. "Loan Prediction Dataset." <https://www.kaggle.com/datasets>
* Bishop, Christopher M. "Pattern Recognition and Machine Learning." Springer, 2006.
* Goodfellow, Ian, et al. "Deep Learning." MIT Press, 2016.
* Kumar, V. et al. "Machine Learning Approaches for Credit Scoring: A Comparative Study." International Journal of Data Science, 2022.
* Zhou, L., et al. "Credit Scoring Models Based on Machine Learning Techniques: A Systematic Review." IEEE Access, vol. 9, 2021, pp. 101567-101581.