Project Report

On

Loan Default Prediction Using Machine Learning and Apache Spark for Enhanced Risk Management.

Submitted in the partial fulfillment for the award of

Post Graduate Diploma in Big Data Analytics (PG-DBDA)

from Know-IT ATC, CDAC ACTS, Pune

Guided by:

Mr. Milind Kapase

Mr. Amey Manjrekar

Submitted By:

Sukanya Nimbalkar (240843025042)

Ishika Sutane (240843025016)

Prachi Moje (240843025026)

Shreyas Kore (240843025037)

CERTIFICATE

TO WHOMSOEVER IT MAY CONCERN

This is to certify that

Sukanya Nimbalkar (240843025042)

Ishika Sutane (240843025016)

Prachi Moje (240843025026)

Shreyas Kore (240843025037)

Have successfully completed their project on

Loan Default Prediction Using Machine Learning and Apache Spark for Enhanced Risk Management.

Under the guidance of Mr. Milind Kapase and Mr. Amey Manjrekar

ACKNOWLEDGEMENT

This project Real time data analysis on ecommerce simulated data was a great learning experience for us and we are submitting this work to CDAC Know-IT (Pune).

We all are very glad to mention the name Anay Tamhankar Sir and Prasad Deshmukh Sir for his valuable guidance to work on this project. His guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

We are highly grateful to Mr. Vaibhav Inamdar Manager (KnowIT), CDAC, for his guidance and support whenever necessary while doing this course Post Graduate Diploma in Big Data Analytics (PGDBDA) through CDAC ACTS, Pune.

Our most heartfelt thanks goes to Mrs. Bakul Joshi (Course Coordinator, PGDBDA) who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in CDAC KnowIT, Pune.

From:

Sukanya Nimbalkar (240843025042)
Ishika Sutane ((240843025016)
Prachi Moje ((240843025026)
Shreyas Kore ((240843025037)

TABLE OF CONTENTS

ABSTRACT

1. INTRODUCTION

2. SYSTEM REQUIREMENTS

2.1 Software Requirements

2.2 Hardware Requirements

3. FUNCTIONAL REQUIREMENTS

4. SYSTEM ARCHITECTURE

5. METHODOLOGY

6. MACHINE LEARNING ALGORITHMS

7. DATA VISUALIZATION AND REPRESENTATION

8. CONCLUSION AND FUTURE SCOPE

9. REFERENCES

ABSTRACT

Loan default prediction is crucial for financial institutions to minimize risks and optimize lending decisions. This project leverages Machine Learning (ML) and Apache Spark to develop a scalable and accurate predictive model for identifying potential loan defaulters.

The model utilizes historical loan data, including demographic details, financial behavior, and the form of
transaction records. Data preprocessing, feature engineering, and exploratory data analysis (EDA) are performed
using Spark MLlib for efficient large-scale processing. Multiple machine learning algorithms, including
Logistic Regression, Random Forest, Gradient Boosting, SupportVector Machine,DecisionTree,KNN algo
and Neural Networks, are trained and evaluated using metrics like Accuracy,Precision, Recall and ROC-AUC.

The best-performing model is deployed as a REST API, allowing real-time loan risk assessment. The solution helps financial institutions reduce non-performing loans (NPLs), improve credit risk management, andenhance decision-making. By integrating big data analytics and AI-driven predictions,
this system ensures a smarter and more efficient loan approval process.

1. INTRODUCTION

Loan default prediction is a critical task in the financial sector to assess credit risk and minimize losses. This project leverages **machine learning** and **Apache Spark** to analyze borrower attributes and loan an details for predicting defaults. The dataset includes financial indicators such as **income, credit score, loan amount, interest rate, and debt-to-income ratio**, along with categorical factors like **employment type, education, and loan purpose**. By utilizing scalable data processing with Spark and advanced predictive models, this project aims to enhance risk management and fraud detection, decision-making in lending institutions. So ultimately improving loan approval processes and reducing bad debt.

Dataset Collection and Features

Data Sources

- For our project, The dataset used in this project was collected from **financial institutions, credit bureaus, and publicly available loan datasets**. The data includes historical loan records, borrower demographics, financial attributes, and repayment details to predict loan default risk.

Data Structure

- The dataset comprises multiple entities, each representing different aspects of the **loan management system**, such as **borrowers, loans, payments, and credit history**. These records are stored within a **relational database**, ensuring structured, efficient, and scalable data storage for predictive analytics and risk assessment..

Dataset Size

- The generated dataset consists of approximately **451,388 records** and contains **32 columns**.
- The size can vary according to our requirements as we have generated our own data for each collection, resulting in a dataset of **moderate to large size**.
- Each collection contains specific information related to its domain, including **financial employment, loan, and demographic details**, resulting in a **rich and diverse dataset**
- suitable for various analytical and predictive modeling tasks.

Features/Attributes

Here is an overview of the key features (attributes) within our dataset:

1. **Demographic Information:**

   Attributes:

   Age – Age of the individual.

   Marital Status – Categorical variable representing marital status

   - MaritalStatus_Single
   - MaritalStatus_Married
   - MaritalStatus Married

2. **Financial & Credit Information**:

Attributes:

- **Income** – Annual income of the individual.
- **Loan Amount** – Amount of loan requested.
- **Credit Score** – Creditworthiness score of the individual.
- **Debt-to-Income Ratio (DTIRatio)** – Ratio of total debt payments to income.
- **Number of Credit Lines (NumCreditLines)** – Total active credit lines.
- **Interest Rate** – Interest rate applicable to the loan. Order Items:

3. **Employment Information:**

Attributes:

- **Months Employed** – Duration of employment in months.
- **Employment Type** – Categorical variable representing employment status:
- EmploymentType_Full-time
- EmploymentType_Part-time
- EmploymentType_Self-employed
- EmploymentType_Unemployed

4. Loan Details:Reviews:

Attributes:

- **Loan Term** – Duration of the loan in months.
- **Loan Purpose** – Categorical variable representing the purpose of the loan:
- LoanPurpose_Auto
- LoanPurpose_Business
- LoanPurpose_Education
- LoanPurpose_Home
- LoanPurpose_Other

5. Other Factors Influencing Loan Approval:

- **Education Level** – Categorical variable representing education background:
  - Education_0, Education_1, Education_2, Education_PhD
- **Has Mortgage** – Indicates whether the individual has an existing mortgage:
  - HasMortgage_Yes
  - HasMortgage_No
- **Has Dependents** – Indicates whether the individual has dependents:
  - HasDependents_Yes
  - HasDependents_No
- **Has Co-Signer** – Indicates whether the loan application has a co-signer:
  - HasCoSigner_Yes
  - HasCoSigner_No

6. Target Variable (Loan Default Prediction):

- **Default** – Binary variable (0 or 1) indicating whether the borrower defaulted on the loan.

2. **SYSTEM REQUIREMENTS**

Hardware Requirements

1. Computer: A computer with sufficient processing power and memory to run data processing and analysis tasks. A modern multicore processor and at least 8 GB of RAM are recommended.

2. Storage: Adequate storage space to store the generated dataset and any additional datasets if required. An SSD (Solid State Drive) is recommended for faster data access.

3. Internet Connection: A stable internet connection for downloading and installing software packages and libraries, as well as for any online resources needed during the project.

Software Requirements

1. Operating System: Windows 10 / Linux / macOS

2. Python: The project heavily relies on Python for data generation, analysis, and machine learning. Ensure Python is installed on your system.

3. Python Libraries: Install the following Python libraries and dependencies using package managers like pip or conda:

NumPy: For numerical computing.

pandas: For data manipulation and analysis.

scikitlearn: For machine learning tasks.

Matplotlib and Seaborn: For data

visualization. Faker: For generating synthetic

data.

.Other libraries specific to your project's

needs.

4. Apache Spark: If your project involves big data processing, consider installing Apache Spark. You can use PySpark to interact with Spark using Python.

5. Integrated Development Environment (IDE): Choose a Pythonfriendly IDE, such as PyCharm, Jupyter Notebook, Visual Studio Code, or your preferred text editor.

Visualization Software

1. Tableau: If you plan to visualize and analyze data with Tableau, install Tableau Desktop.

3. FUNCTIONAL REQUIREMENTS

(1) **Python 3:**

- Python is a general purpose and high level programming language.
- It is use for developing desktop GUI applications, websites and web applications.
- Python allows to focus on core functionality of the application by taking care of common programming tasks.
- Python is derived from many other languages, including ABC, Modula3, C, C++, Algol68, Small Talk, and Unix shell and other scripting languages.

(2) **Apache Spark:**

What is Spark: Apache Spark is an opensource distributed computing system designed for processing large volumes of data.

Key Features: Spark provides a number of key features that make it wellsuited for processing big data, including inmemory processing, support for various data sources and formats, faulttolerance, and scalability.

Spark also provides a range of APIs, including SQL, streaming, machine learning, and graph processing, making it a versatile platform for a wide range of use cases.

(3) **Machine Learning (ML):**

What is ML: Machine Learning (ML) is a subset of artificial intelligence (AI) that enables systems to learn patterns from data and make predictions or decisions without being explicitly programmed. It is widely used in various domains, including healthcare, finance, and automation.

Key Features; **Data-Driven Learning** – Learns from historical data to improve decision

Supervised & Unsupervised Learning – Supports different learning paradigms, including classification, regression, clustering and reinforcement learning
Automation – Reduces the need for manual rule-based programming by adapting to new
Scalabilit**y** – Can process large datasets efficiently using distributed computing frameworks.
Model Evaluation & Optimization – Uses performance metrics (e.g. accuracy, precision recall) and
hyper parameter tuning to enhance model efficiency.
Integration with Big Data – Works with tools like Apache Spark, Hadoop, and cloud platform to process massive datasets.

(4) **Tableau:**

- Data visualization is the graphical representation of information and data.
- It helps create interactive elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.
- Tableau is widely used for Business Intelligence but is not limited to it.
- It helps create interactive graphs and charts in the form of dashboards and worksheets to gain business insights.
- All of this is made possible with gestures as simple as drag and drop.

ARCHITECTURE

Python, Pyspark, Report, Ml, Tablue give me ARCHITECTURE Diagram

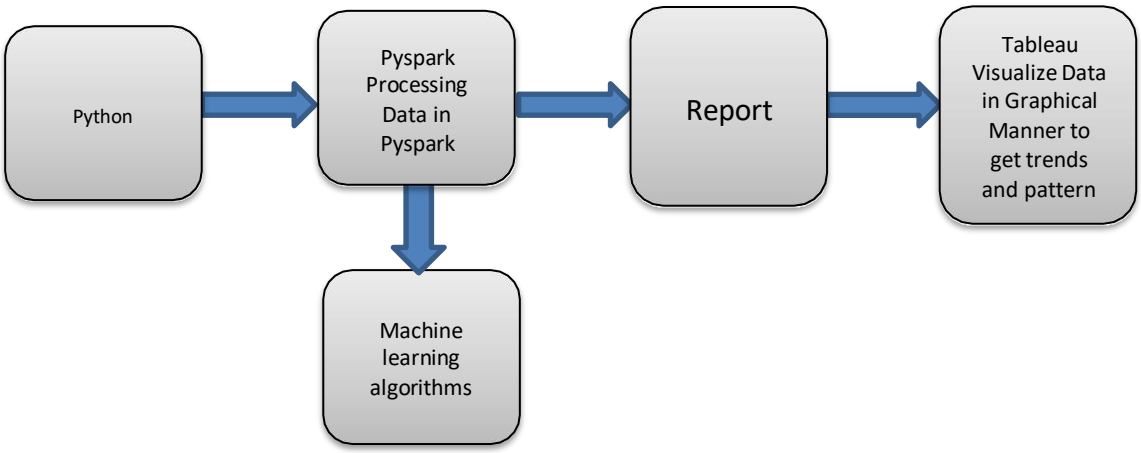


Fig: System Architecture of Real time Data Analysis on ecommerce simulated data

MACHINE LEARNING ALGORITHMS

1. **SMOTE (Synthetic Minority Over-sampling Technique)**

- **Purpose**: Balances imbalanced datasets by generating synthetic samples for the minority class.
- **How It Works**:
  - Identifies the k-nearest neighbors for each minority class sample.
  - Generates synthetic samples by interpolating between existing samples and their neigh
- **Use Case in File**: Applied to resample the dataset for class balancing.

2. **One-Hot Encoding (OHE)**

- **Purpose**: Converts categorical variables into numerical format for compatibility with ML
- **How It Works**:
  - Creates binary columns for each category in a feature.
  - Ensures no ordinal relationship is implied between categories.
- **Use Case in File**: Applied to transform categorical variables such as education levels.

3. Train-Test Split

- **Purpose**: Splits the dataset into training and testing subsets to evaluate model performance.
- **Key Concepts**:
  - Training Set: Used for learning.
  - Test Set: Used for evaluation after training.
- **Importance**: Prevents over fitting by testing on unseen data.

4. **Logistic Regression**

- **Purpose**: A supervised learning algorithm for binary classification tasks.
- **Mathematics**:
  - Models the probability that a given input belongs to a specific class using the sigmoid function:
- **When to Use**: Suitable for predicting categorical outcomes like "Yes/No, Default/No Default.

5. **Decision Trees**

- **Purpose**: A tree-structured algorithm used for classification and regression.
- **Key Components**:
  - Root Node: Represents the entire dataset.
  - Splits: Decision rules based on features.
  - Leaf Nodes: Predictions or final outcomes.

- **Advantages**:
  - Easy to interpret.
  - Handles non-linear relationships well.

5. **Random Forest**

- **Purpose**: An ensemble learning method based on multiple decision trees.
- **How It Works**:
  - Trains multiple trees on random subsets of data and features.
  - Combines predictions from all trees (majority voting for classification).
- **Key Benefits**:
  - Reduces over fitting compared to single decision trees.
  - Improves accuracy.

6. Support Vector Machine

- **Purpose:** Support Vector Machine (SVM) **is a** supervised machine learning algorithm used a classification and regression tasks**.**
- **How It Works:**
  - **Linear SVM**: For linearly separable data, SVM finds a hyperplane that divides the data points into two classes. The goal is to maximize the margin between the closest points of the two classes, called support vectors.
  - **Non-linear SVM**: When data is not linearly separable, SVM uses kernel tricks to map the data into a higher-dimensional space where a hyperplane can effectively separate the classes.
  - **Classification**: In classification tasks, once the optimal hyperplane is determined, new data points can be classified based on which side of the hyperplane they fall on.

- **Key Benefits**:

  - Effective in High-Dimensional Spaces
  - Handles Both Linear and Non-Linear Data
  - Robust Against Overfitting
  - Good Generalization Performance

7. **Evaluation Metrics**

- **Accuracy**: Percentage of correct predictions out of total predictions.
- **Precision, Recall, F1-Score**:
  - Precision: Proportion of true positives among predicted positives.
  - Recall: Proportion of true positives among actual positives.
  - F1-Score: Harmonic mean of precision and recall.
- **ROC Curve & AUC**:
  - Evaluates model performance by plotting true positive rate vs. false positive rate.
  - AUC represents the area under the ROC curve (closer to 1 is better).

Benefits:

1. **Improved Decision-Making**

- Machine learning (ML) models can analyze vast amounts of data quickly and derive
- meaningful patterns and trends.

2. Enhanced Predictive Power

- Techniques such as **SMOTE** for data balancing improve the model's ability to predict
- outcomes, even in cases with imbalanced data.
- ML models provide reliable predictions for future events, enabling better planning and trategy.

3. Versatility Across Domains

- ML algorithms are versatile and can be applied to various fields such as:
  - **Finance**: Predicting loan defaults.
  - **Healthcare**: Disease diagnosis.
  - **Retail**: Customer behavior analysis.
  - **Marketing**: Lead scoring and campaign optimization.

4. Efficiency and Automation

- ML reduces manual effort by automating tasks like data cleaning, feature selection, and analysis.
- Algorithms such as **Random Forest** and **Decision Trees** require minimal preprocessing and
- can quickly process large datasets.

5. Scalability

- ML models can scale with the growth of data.
- Algorithms such as **Logistic Regression** and **Random Forest** handle datasets of varying sizes and complexities efficiently.

6. Better Interpretability (With Some Algorithms)

- Models like **Logistic Regression** and **Decision Trees** provide clear insights into which
- features influence outcomes.
- Feature importance tools in **Random Forest** highlight which variables matter the most in decision-making.

7. Enhanced Model Performance

- Using evaluation metrics (e.g., **Accuracy**, **Precision**, **Recall**, **F1-Score**, **ROC-AUC**) ensures models are tuned and optimized for real-world deployment.

Conclusion:

, This project aimed to predict loan defaults using machine learning techniques. Key steps included:

Data Preprocessing – Cleaning the dataset, encoding categorical features, and handling missing values.

Class Imbalance Handling – Using **SMOTE** to balance the dataset and improve modelfairness.

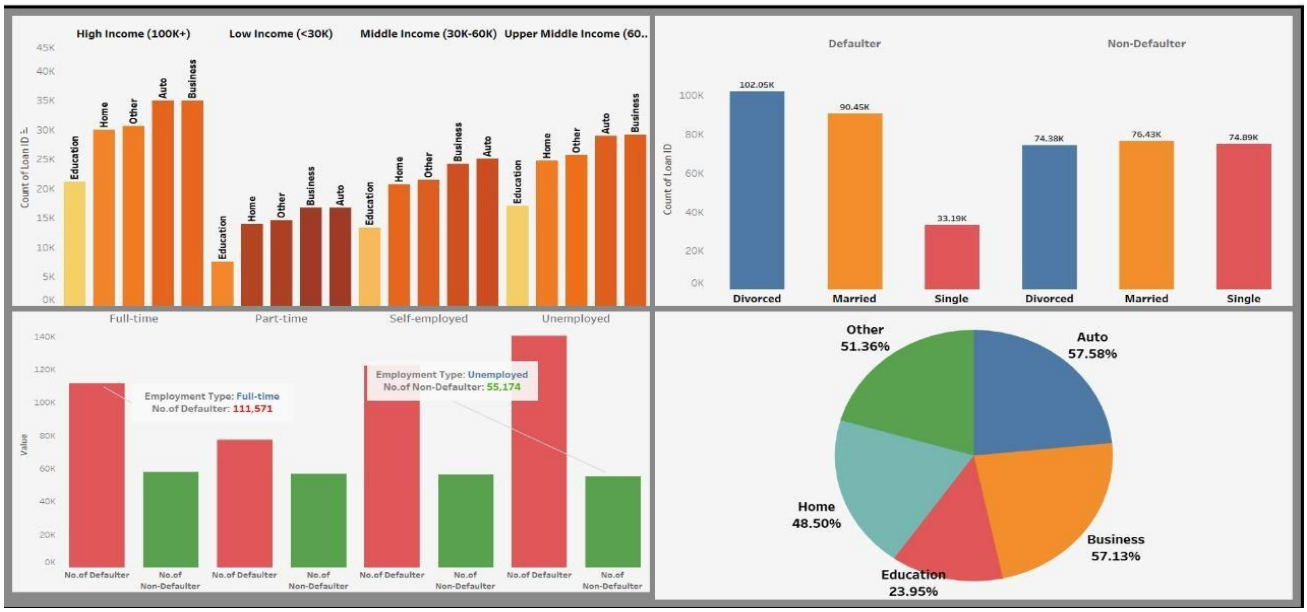
Feature Engineering – Transforming data to enhance machine learning performance.

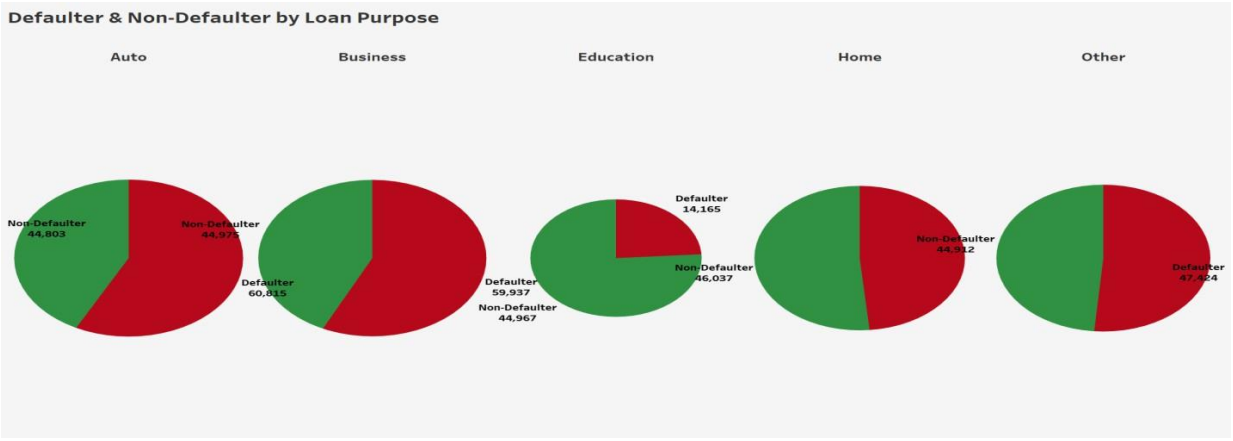
Model Training & Evaluation – Applying predictive models and assessing performance using accuracy, precision, recall, and other metrics.

By implementing these techniques, we improved the reliability of loan default predictions, helping financial institutions minimize risk and make informed lending decisions.

.

DATA VISUALIZATION AND REPRESENTATION





CONCLUSION AND FUTURE SCOPE

Conclusion

Loan Default Prediction using ML and Spark Enhanced risk management demonstrates the power of **machine learning and big data analytics** in the financial domain. By integrating advanced predictive models with real-time data processing, this project provides a robust framework for financial institutions to **assess borrower risks, make informed lending decisions, and minimize losses due to defaults**.

As financial markets continue to evolve, leveraging AI-driven risk assessment will be a game-changer in ensuring sustainable and profitable lending practices.

Our project has revolved around several key aspects:

- **Data Collection and Preprocessing:**
  - We gathered historical loan data, including borrower details, loan amounts, repayment history, and financial indicators.
  - The data underwent cleaning, handling missing values, feature engineering, and transformation to ensure high-quality inputs for model training.
- **Data Storage and Management:**
  - Apache Hadoop and Apache Spark were employed for distributed storage and processing, enabling scalability and efficiency when handling large datasets.
- **Machine Learning Model Development:**
  - Various machine learning models, including **Logistic Regression, Random Forest, Gradient Boosting, and XGBoost**, were implemented and evaluated.
  - Feature selection techniques helped in identifying the most significant factors contributing to loan defaults.
- **Model Training and Evaluation:**
  - Models were trained using Spark MLlib, leveraging distributed computing for faster processing.
  - Metrics such as **accuracy, precision, recall, F1-score, and AUC-ROC** were analysed to assess model performance and ensure reliability.
- **Predictive Analytics and Risk Assessment:**
  - The final model provided real-time risk assessments, categorizing borrowers based on their likelihood of defaulting.
  - Financial institutions can leverage these insights to make informed lending decisions and minimize credit risks.
- **Data Visualization and Insights:**
  - Tools like **Tableau and Matplotlib** were used to visualize loan trends, borrower risk profiles, and feature importance, aiding in better decision-making.
- **Future Scope and Enhancements:**
  - The project has vast potential for expansion, including **deep learning models, time-series analysis for trend forecasting, and integration with external credit scoring data**.
  - Deploying the model as an API for real-time loan applications could enhance financial services.
- **Business Relevance:**

- This project highlights the significance of **data-driven lending** in the banking and financial sector.
- By leveraging machine learning and big data technologies, financial institutions can reduce **non-performing assets (NPAs), improve credit risk management, and enhance overall profitability**.

Future Scope:

The **Loan Default Prediction using ML and Spark** Enhanced risk management is continuously evolving, and there are numerous opportunities for future advancements. Below are some key areas where the project can be the enhanced:

1. **Integration with Real-Time Credit Scoring**

- Incorporate **real-time credit bureau data** (e.g., CIBIL, Experian, Equifax) to enhance model accuracy.
- Use **alternative credit scoring** methods, such as social media behavior, digital transactions,
- and employment history.

2. **Advanced Machine Learning and Deep Learning Models**

- Implement **Deep Neural Networks (DNNs)** and **Recurrent Neural Networks (RNNs)** to capture complex relationships in borrower data.
- Use **AutoML frameworks** to automatically optimize hyperparameters for better predictive performance.
- Experiment with **graph-based models** to analyze borrower relationships and fraud detection.

3. **Explainable AI (XAI) for Transparent Decision-Making**

- Implement **SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable**
- **Model-agnostic Explanations)** for better interpretability.
- Ensure **regulatory compliance** by providing explainable insights to financial institutions and auditors.

4. **Real-Time Loan Application Processing**

- Deploy the trained ML model as an **API** for real-time risk assessment during loan applications.
- Integrate **chatbots and AI-powered assistants** to guide customers through risk assessment
- and financial planning.

5. **Fraud Detection and Anomaly Detection**

- Use **unsupervised learning techniques (e.g., Isolation Forest, Autoencoders)** to detect fraudulent loan applications.
- Implement **blockchain technology** for secure and transparent loan transactions.

6. **Time-Series Forecasting for Financial Stability**

- Apply **LSTM (Long Short-Term Memory) models** to analyze repayment behavior over time.
- Predict **loan delinquency trends** for better financial planning and risk mitigation.

7. Big Data and Cloud Deployment

- Implement **Spark on Kubernetes** for scalable and efficient model training.
- Deploy on cloud platforms like **AWS, Azure, or Google Cloud** for global accessibility.
- Utilize **serverless architectures** (e.g., AWS Lambda) to process loan applications in real-time.

8. Personalized Loan Recommendations

- Use **AI-powered recommendation systems** to suggest personalized loan plans based on risk assessment.
- Enhance user experience by integrating predictive analytics with **customer segmentation**.

9. Regulatory and Compliance Enhancements

- Ensure models comply with financial regulations like **GDPR, Basel III, and Fair Lending Laws**.
- Implement **bias detection algorithms** to ensure fair and unbiased credit decisions.

10. Hybrid Approaches for Improved Accuracy

- Combine **rule-based systems with AI models** to improve prediction accuracy.
- Utilize **ensemble learning (Stacking, Bagging, Boosting)** for more robust performance.

REFERENCES

1. https://towardsdatascience.com/top-3-python-packages-to-generate-synthetic-data-33a351a5de0c

2. Apache Spark. [https://spark.apache.org/]

3. Python. [https://www.python.org/]

4. scikit-learn. [https://scikit-learn.org/]

5. Confluent Python Client. [https://docs.confluent.io/platform/current/clients/confluent- kafka-python/html/index.html]

6. Faker. [https://faker.readthedocs.io/en/master/]

7. "Machine Learning using Python" by Prof. U Dinesh Kumar, IIM Bangalore