**Project Topic: Loan Default Prediction**

**Project Overview:**

Loan default prediction is a concern for financial institutions as it directly impacts their profitability and risk management strategies. By guessing these defaults correctly, banks can take early action to lower risks, make better loan decisions, and get better financial results. This project aims to build a predictive model capable of accurately identifying potential loan defaults before they occur.

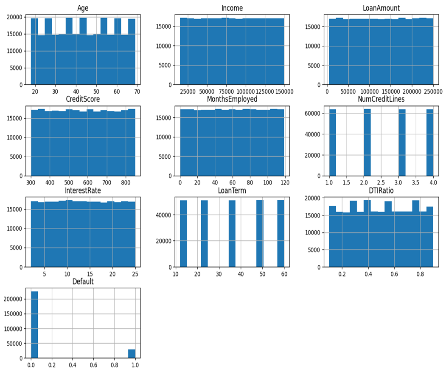
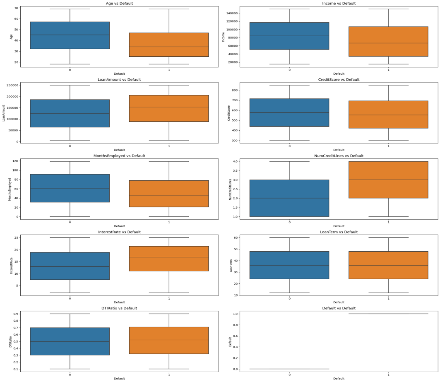
**A diagram of a computer

Description automatically generated**



1. **Data Collection:**
   * Connect to data sources- csv files from Kaggle to gather relevant data.
   * Dataset contains the following features - Age, Income, LoanAmount, CreditScore, MonthsEmployed, NumCreditLines , InterestRate, LoanTerm, DTIRatio, Education, EmploymentType, HasMortgage, HasDependents, LoanPurpose, HasCoSigner, Default.
   * The label is Default (0/1).
2. **Data Cleaning and Preprocessing:**
   * Use libraries like Pandas and NumPy for data manipulation.
   * Check for null values and missing values. Check for ranges of each feature values and the distribution in the data.
   * Understand the domain significance of each feature.
   * Check for the count of data points present in each class.
3. **Exploratory Data Analysis:**
   * Utilize visualization tools like Matplotlib and Seaborn.
   * Identify key features and correlations.
   * Create plots to see the relationships between the features and the label.
   * Generate the heatmap to depict the correlation between the dependent and independent variables.

A blue and red chart with white text

Description automatically generated

1. **Feature Transformation:**
   * Identify what are the numeric and categorical columns and choose appropriate data encoding techniques for the same.
   * Choose what features are relevant to the prediction of the label and what can improve the model accuracy.
   * In this case all the categorical columns are encoded using one hot encoding. This is because the number of unique values is low i.e. We have low cardinality.
2. **Model Building/Tuning:**
   * Implement multiple models using Scikit-Learn, XGBoost, or similar libraries – Random Forest, Decision Trees, KNN, AdaBoost, Naïve bayes, SVM, Logistic regression.
   * Use GridSearchCV for hyperparameter tuning and tweak the model parameters to improve accuracy.
   * Used parameter ‘balanced’ to handle the label imbalance present in the data.
   * Choose the model with the best accuracy and save it as a pickle file in a storage container (Model Artifactory).
   * From all the models, AdaBoost gave the best accuracy, so it was chosen for the next steps.
3. **Model Predictions and Evaluation:**
   * Use cloud platforms (AWS, Azure) for deployment and scaling.
   * Use the stored trained model in the container blob storage and use it to make the predictions on the test dataset.
   * Generate relevant metrics like accuracy, precision, confusion matrix etc.
4. **Pipelining and Automation:**
   * Set up automated training pipeline using Azure Data Factory.
   * Regularly update the model with new data and monitor performance.

A screenshot of a computer

Description automatically generated

**Expected Outcomes:**

* Reduced Loan Defaults: By predicting which loans are likely to default, the institution can take proactive measures to mitigate risk, such as offering revised payment plans, increasing monitoring, or even denying high-risk loans.
* Enhanced Risk Management: With a predictive model, the bank can better manage the risk profile of its loan portfolio.
* Increased Financial Stability

**Tools and Technologies:**

* **Programming Languages:** Python, SQL, Spark
* **Libraries:** Pandas, NumPy, Scikit-Learn, XGBoost, Matplotlib, Seaborn, Pickle
* **Deployment:** Azure Databricks, Azure Data Factory, Azure storage container
* **Cloud Platforms:** Azure

---------------------------------------------------------------------------------------------------------------------

**Appendix:**

A blue and red chart with white text

Description automatically generated

A group of blue and orange boxes

Description automatically generated

A group of blue bars

Description automatically generated

A screenshot of a graph

Description automatically generated

A graph with blue rectangular bars

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Age 255347 non-null int32

1 Income 255347 non-null int32

2 LoanAmount 255347 non-null int32

3 CreditScore 255347 non-null int32

4 MonthsEmployed 255347 non-null int32

5 NumCreditLines 255347 non-null int32

6 InterestRate 255347 non-null float64

7 LoanTerm 255347 non-null int32

8 DTIRatio 255347 non-null float64

9 Default 255347 non-null int32

10 Education\_High School 255347 non-null Sparse[float64, 0]

11 Education\_Master's 255347 non-null Sparse[float64, 0]

12 Education\_PhD 255347 non-null Sparse[float64, 0]

13 EmploymentType\_Part-time 255347 non-null Sparse[float64, 0]

14 EmploymentType\_Self-employed 255347 non-null Sparse[float64, 0]

15 EmploymentType\_Unemployed 255347 non-null Sparse[float64, 0]

16 MaritalStatus\_Married 255347 non-null Sparse[float64, 0]

17 MaritalStatus\_Single 255347 non-null Sparse[float64, 0]

18 HasMortgage\_Yes 255347 non-null Sparse[float64, 0]

19 HasDependents\_Yes 255347 non-null Sparse[float64, 0]

20 LoanPurpose\_Business 255347 non-null Sparse[float64, 0]

21 LoanPurpose\_Education 255347 non-null Sparse[float64, 0]

22 LoanPurpose\_Home 255347 non-null Sparse[float64, 0]

23 LoanPurpose\_Other 255347 non-null Sparse[float64, 0]

24 HasCoSigner\_Yes 255347 non-null Sparse[float64, 0]