**Project Title**: "Credit Score Classification"

**Team Members**:

Rahma Ashraf, Mohamed Ghanam, Enas Ibrahim, Mohamed Saeed

**Institution**: National Telecommunication Institute (NTI)

Eng: Anas Muhamed

**Date**: 19/9/2024

Main Objective

Company has collected basic bank details and gathered a lot of credit-related information. The management wants to build an intelligent system to segregate the people into credit score brackets to reduce manual efforts.

Our Mission

Given a person’s credit-related information, build a machine learning model that can classify the credit score.

**Summary of Results**

the outcome of our model ( accuracy achieved : 71% )

Dataset Description

Dataset Link: www.kaggle.com/datasets/parisrohan/credit-score-classification/data

* **ID**: The unique identifier for each record in the dataset.
* Customer\_ID: The unique identifier for each customer.
* Month: The month to which the record corresponds.
* Name: The name of the customer.
* Age: The age of the customer.
* Note: There are some data quality issues, such as anomalous values.
* SSN: The Social Security Number of the customer.
* Occupation: The occupation of the customer.
* Annual\_Income: The annual income of the customer in USD.
* Monthly\_Inhand\_Salary: The monthly take-home (net) salary of the customer in USD.
* Num\_Bank\_Accounts: The number of bank accounts held by the customer.
* Num\_Credit\_Card: The number of credit cards held by the customer.
* Interest\_Rate: The average interest rate (%) applicable to the customer's loans or credit cards.
* Num\_of\_Loan: The number of loans (including mortgages, personal loans, auto loans, etc.) that the customer has.
* Type\_of\_Loan: The types of loans taken by the customer, listed in a single cell and separated by semicolons.
* Delay\_from\_due\_date: The average delay (in days) from the due date for payments across all the customer's loans and credit cards.
* Num\_of\_Delayed\_Payment: The total number of times the customer has delayed payments across all loans and credit cards.
* Changed\_Credit\_Limit: Any recent changes to the customer's credit limit (requires further context).
* Num\_Credit\_Inquiries: The number of inquiries made on the customer's credit report, typically indicating applications for new credit.
* Credit\_Mix: The diversity of credit types held by the customer, affecting credit scores. Specific categories (e.g., "Good", "Bad") need further definition.
* Outstanding\_Debt: The total outstanding debt the customer has across all loans and credit lines.
* Credit\_Utilization\_Ratio: The ratio of total debt to total available credit, a key indicator of credit risk.
* Credit\_History\_Age: The age of the customer's credit history.
* Payment\_of\_Min\_Amount: Indicates whether the customer typically pays at least the minimum amount due on their debts.
* Total\_EMI\_per\_month: The total monthly payment the customer makes towards all their loans as EMI (Equated Monthly Installment).
* Amount\_invested\_monthly: The amount of money the customer invests monthly (could be in savings accounts, stocks, bonds, etc.).
* Payment\_Behaviour: Characterizes the customer's payment behavior in terms of expenditure and value of transactions.
* Monthly\_Balance: The average monthly balance maintained by the customer in their accounts.
* Credit\_Score: The credit score of the customer, a summary measure of their creditworthiness.

**1. Handling Missing Values:**

* **Identify Missing Data**:
  + Start by using techniques like df.isnull().sum() to check for any missing values in the dataset.
* **Handling Techniques**:
  + **Removing Missing Data**: If the percentage of missing data in certain columns is high and the columns are less significant, you may choose to drop them using df.dropna().
  + **Imputation**: For columns where data is crucial, you can fill in missing values using techniques like:
    - **Mean/Median/Mode Imputation**: Replacing missing values with the mean or median of the column (e.g., for numerical data), or the mode (for categorical data) using df.fillna(df['column'].mean()).
    - **Forward Fill/Backward Fill**: Propagating the next/previous value (useful for time series data).
    - **K-Nearest Neighbors (KNN) Imputation**: An advanced technique using similar data points to fill in missing values.

**2. Handling Outliers:**

* **Identifying Outliers**:
  + Use box plots, scatter plots, or z-scores to detect outliers.
  + Calculate z-scores or IQR (Interquartile Range) to define threshold points for outliers.
* **Handling Techniques**:
  + **Removal**: If the outliers are genuine errors or highly improbable, you may decide to remove them.
  + **Capping**: For extreme outliers, you can cap the data to a certain percentile (e.g., 99th or 1st).
  + **Transformation**: Applying log, square root, or other transformations to reduce the impact of extreme outliers.

**3. Handling Imbalanced Data:**

* **Identify Imbalance**: Check class distribution for imbalance using value\_counts().
* **Balancing Techniques**:
  + **Over-sampling**: Increase the size of the minority class by duplicating samples (e.g., using SMOTE - Synthetic Minority Over-sampling Technique).
  + **Under-sampling**: Reduce the size of the majority class.
  + **Class Weights**: In algorithms like Logistic Regression or SVM, adjust class weights to give more importance to the minority class.

**4. Feature Scaling / Normalization:**

* **Why It's Needed**: Many algorithms like SVM, KNN, and Neural Networks are sensitive to the scale of data. So, feature scaling helps improve performance.

**6. Exploratory Data Analysis (EDA)**

* **Target variable (credit score):**
* **Visualizations**:

A group of graphs showing the value of credit

Description automatically generated

A group of blue and white graphs

Description automatically generatedA group of blue and white bars

Description automatically generated with medium confidence

A close-up of a graph

Description automatically generated

Classification Report Of The Project:

A screen shot of a black screen

Description automatically generated