Data Intensive Computing – CSE 587A

Project Phase – 1

Problem Statement: To develop a machine learning model that can classify credit scores using information gathered from banks and credit-related datasets.

Introduction: Customer's income, payment history, number of loans, credit score, and other pertinent financial data should all be taken into account when the model assigns them to one of the three risk categories (good, standard, or poor). This will make it possible for financial organizations to decide on credit restrictions and loan approvals with knowledge. Our goal is to create a model that can classify the credit scores of potential or current clients by utilizing machine learning techniques. This would enable financial institutions to reduce risks and customize their financial offerings to suit the requirements of various consumer groups.

The potential of our Project (Objective): In our project, we are going to develop a prediction model that can accurately classify credit scores based on the dataset. Lenders use credit scores, which are numerical representations of a person's creditworthiness, to assess the risk of making a loan. This methodology attempts to help banks, credit card companies, and other lenders make well-informed judgments about credit limits, interest rates, and loan approvals by classifying credit scores.

Data Source: For this project, we have acquired the dataset from Kaggle which gives information about bank and credit-related datasets. Our dataset consists of 28 columns and 100000 rows of customer and banking information which impacts credit score. Our dataset has 100000 customers and their credit-related information. The following table includes all the features of each customer.

The link for the dataset is https://www.kaggle.com/datasets/parisrohan/credit-score-classification which we got from the Kaggle website.

Feature Tables

Features	Representation of the feature	Data type
ID	It tells us the ID of the customer	Object
Customer_ID	It is the unique ID provided to each customer	Object
Month	It has the month of the card provided	object
Name	It contains the names of the customers.	Object
Age	It tells us customer age	Int64

SSN	It contains SSN of the customer	Int 64
Occupation	It describes occupation of the customer	Object
Annual_Income	It contains customers annual income	Float64
Monthly_Inhand_Salary	It contains customers' monthly income	Float64
Num_Bank_Accounts	It tells us how many accounts the customer has	Int64
Num_Credit_Card	It tells us a number of credit cards the customer has	Int64
Interest_Rate	It contains the interest rate of the loan taken by the customer	Float64
Num_of_Loan	It tells us the total number of loans secured by the customer	Int64
Type_of_Loan	It tells us which type of loan the customer has taken	Object
Delay_from_due_date	It tells us some days that the customer has delayed the payment of the loan	Int64
Num_of_Delayed_Payment	It tells the total number of payments delayed by the customer	Int64
Changed_Credit_Limit	It tells us the updated credit limit of the customer	Float64
Num_Credit_Inquiries	It tells us about the credit inquiries made by the customer	Int64

Credit_Mix	It gives an overview of the customer's credit card payment	Object
Outstanding_Debt	It contains the total outstanding debt of the customer	Float64
Credit_Utilization_Ratio	It provides the measure of available credit	Float64
Payment_of_Min_Amount	The minimum payment that a credit card holder is required	Float64
Credit_History_Age	The credit history of the customer	Int64
Total_EMI_per_month	Monthly EMI to be paid by the customer	Float64
Amount_invested_monthly	The monthly amount invested by the customer	Float64
Payment_Behaviour	Payment behaviour of the payment	object
Monthly_Balance	The amount remaining in an account at the end of the month	Float64
Credit_Score	The credit score given to the customer is based on other attributes	object

Steps of Data Preprocessing:

1) The first step in our data preprocessing involves dropping unwanted columns. These columns ID, Customer_ID, Month, Age, Monthly_Inhand_Salary, Credit_Mix, Credit_History_Age, Payment_Behaviour, Name, SSN are dropped using drop() function. These columns do not impact the output.
2) In this step, we did datatype conversion to ensure capability and efficiency for analysis. We have used the pandas function to convert the datatypes.
3) Next, we have renamed the column for easy readability of the users using the rename() function.

4) After this step, We have handled the null values for Annual_Income, Num_of_Loan, Type_of_Loan, Num_of_Delayed_Payments, Changed_Credit_Limit, Num_of_Credit_Inquiries, Outstanding_Debt , Amount_invested_monthly , Monthly_Balance using median tendency for numeric features and unknown for categorical features.
5) In this step we have cleaned the numeric data by using numpy for Num_of_Loan, Changed_Credit_Limit, Delay_from_due_date, Amount_invested_monthly .
6) Next we have standardized the text data by using pandas inorder to remove punctuations and to convert the text into lower case.
7) In the following step, we handled the outliers for Annual_Income and Amount_Invested_Monthly by defining the bounds for outliers and filtering out outliers. Additionally we have plotted a box plot to see the effect of removing the outliers.
8) In the next step, we initialized the label encoder for the attribute Credit_Score and have applied One-hot Encoding to the attribute occupation using pandas. Then, we joined the encoded columns back to the original data frame.
9) For the next step, we have done feature Engineering by calculating the ratio of Annual_Income to Amount_Invested_Monthly by using lambda function. We have then estimated the loan affordability of Annual_Income using pandas. We have then displayed the updated data frame to verify the new features
10) Finally, We have standardized the numerical features for applying PCA (Principal Component Analysis). After applying the PCA we have reduced the data to two dimensional for illustration

<u>**Exploratory Data Analysis (EDA):**</u>

1. Exploratory data analysis is an excellent technique for deriving conclusions and understanding the data.

```
data=pd.read_csv('creditrisk.csv')
data.head()
```

	ID	Customer_ID	Month	Name	Age	SSN	Occupation	Annual_Income	Monthly_Inhand_Salary	Num_Bank_Accounts	...	Credit_Mix	Outstanding_
0	0x1602	CUS_0xd40	January	Aaron Maashoh	23	821-00-0265	Scientist	19114.12	1824.843333	3	...	_	8
1	0x1603	CUS_0xd40	February	Aaron Maashoh	23	821-00-0265	Scientist	19114.12	NaN	3	...	Good	8
2	0x1604	CUS_0xd40	March	Aaron Maashoh	-500	821-00-0265	Scientist	19114.12	NaN	3	...	Good	8
3	0x1605	CUS_0xd40	April	Aaron Maashoh	23	821-00-0265	Scientist	19114.12	NaN	3	...	Good	8
4	0x1606	CUS_0xd40	May	Aaron Maashoh	23	821-00-0265	Scientist	19114.12	1824.843333	3	...	Good	8

5 rows × 28 columns

2) We can view the columns using data.info()

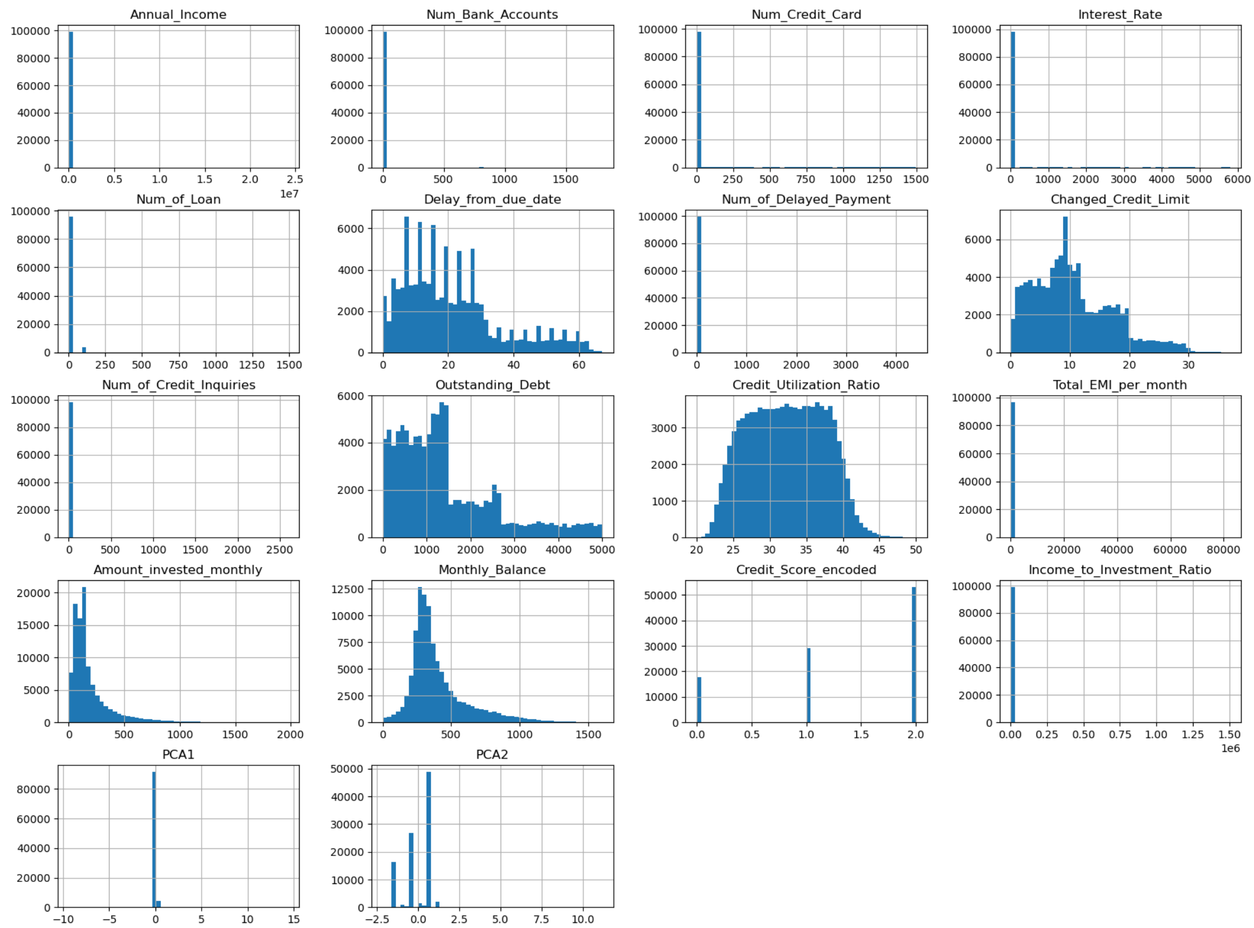
```
In [4]: data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100000 entries, 0 to 99999
Data columns (total 28 columns):
 #   Column                    Non-Null Count   Dtype  
---  ------                    --------------   -----  
 0   ID                        100000 non-null  object 
 1   Customer_ID               100000 non-null  object 
 2   Month                     100000 non-null  object 
 3   Name                      90015 non-null   object 
 4   Age                       100000 non-null  object 
 5   SSN                       100000 non-null  object 
 6   Occupation                100000 non-null  object 
 7   Annual_Income             100000 non-null  object 
 8   Monthly_Inhand_Salary     84998 non-null   float64
 9   Num_Bank_Accounts         100000 non-null  int64  
 10  Num_Credit_Card           100000 non-null  int64  
 11  Interest_Rate             100000 non-null  int64  
 12  Num_of_Loan               100000 non-null  object 
 13  Type_of_Loan              88592 non-null   object 
 14  Delay_from_due_date       100000 non-null  int64  
 15  Num_of_Delayed_Payment    92998 non-null   object 
 16  Changed_Credit_Limit      100000 non-null  object 
 17  Num_Credit_Inquiries      98035 non-null   float64
 18  Credit_Mix                100000 non-null  object 
 19  Outstanding_Debt          100000 non-null  object 
 20  Credit_Utilization_Ratio  100000 non-null  float64
 21  Credit_History_Age        90970 non-null   object 
 22  Payment_of_Min_Amount     100000 non-null  object 
 23  Total_EMI_per_month       100000 non-null  float64
 24  Amount_invested_monthly   95521 non-null   object 
 25  Payment_Behaviour         100000 non-null  object 
 26  Monthly_Balance           98800 non-null   object 
 27  Credit_Score              100000 non-null  object 
dtypes: float64(4), int64(4), object(20)
memory usage: 21.4+ MB
```

3) To get the statistics of data, we used data.describe()

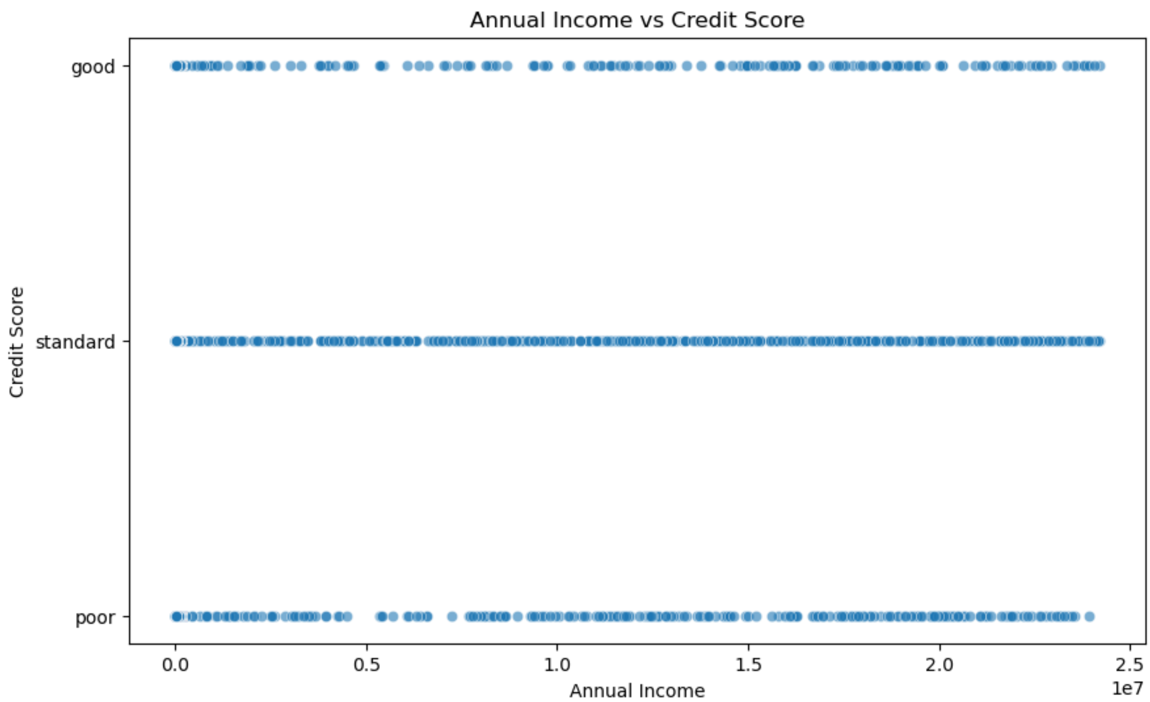
```
data.describe()
```

	Annual_Income	Num_Bank_Accounts	Num_Credit_Card	Interest_Rate	Num_of_Loan	Delay_from_due_date	Num_of_Delayed_Payment
count	1.000000e+05	100000.000000	100000.00000	100000.000000	100000.000000	100000.000000	100000.000000
mean	1.687352e+05	17.091280	22.47443	72.466040	10.542850	21.095040	29.373010
std	1.392075e+06	117.404834	129.05741	466.422621	60.133886	14.822802	215.671804
min	7.005930e+03	-1.000000	0.00000	1.000000	0.000000	0.000000	-3.000000
25%	2.006286e+04	3.000000	4.00000	8.000000	2.000000	10.000000	9.000000
50%	3.755074e+04	6.000000	5.00000	13.000000	3.000000	18.000000	14.000000
75%	7.006492e+04	7.000000	7.00000	20.000000	6.000000	28.000000	18.000000
max	2.419806e+07	1798.000000	1499.00000	5797.000000	1496.000000	67.000000	4397.000000

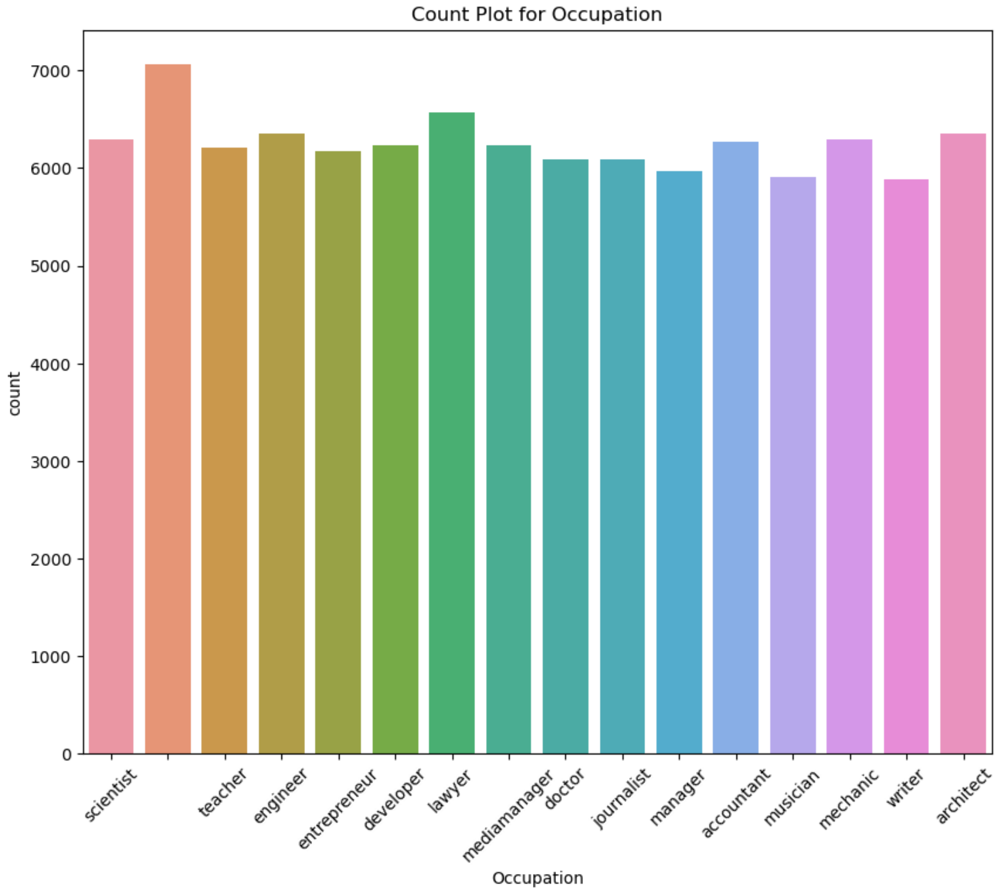
4)The below image is the output of the histogram plotting code which is typically used to visualize the distribution of numerical data within a dataset.

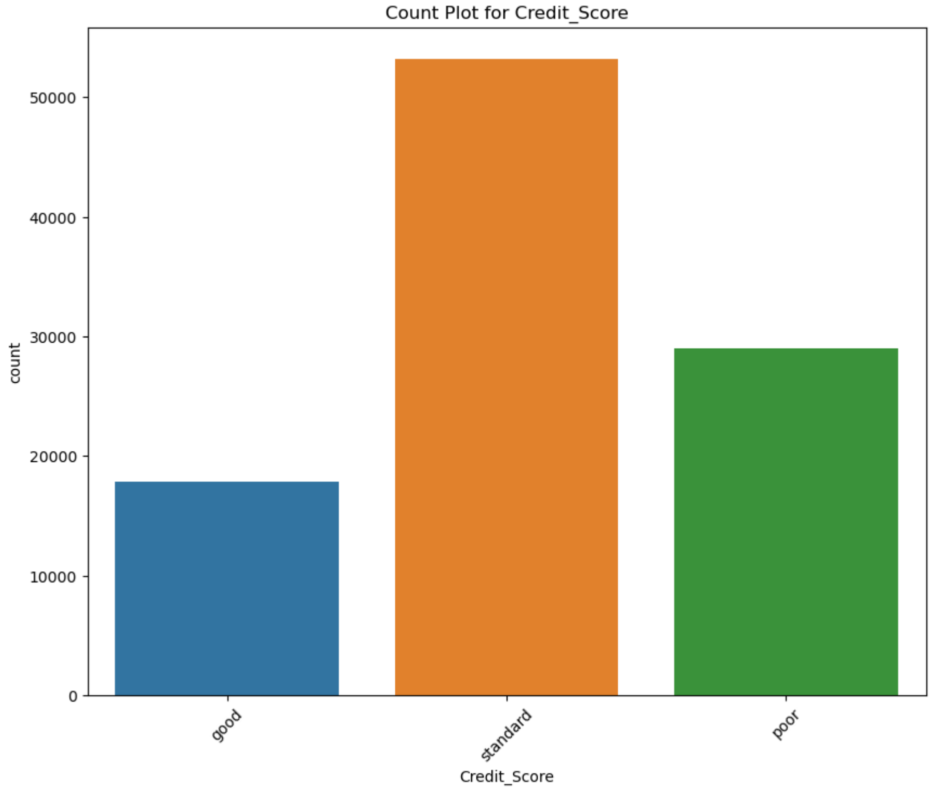


5) The image provided below is the output of a scatter plot which is used to examine the relationship between two variables - Annual_Income and Credit_Score.

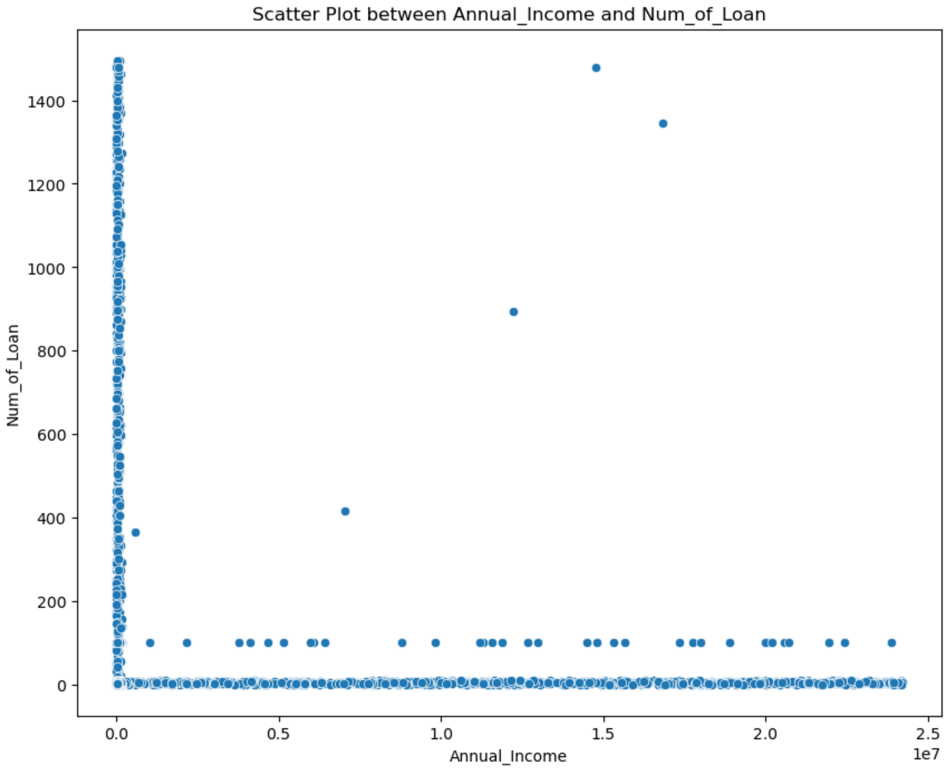


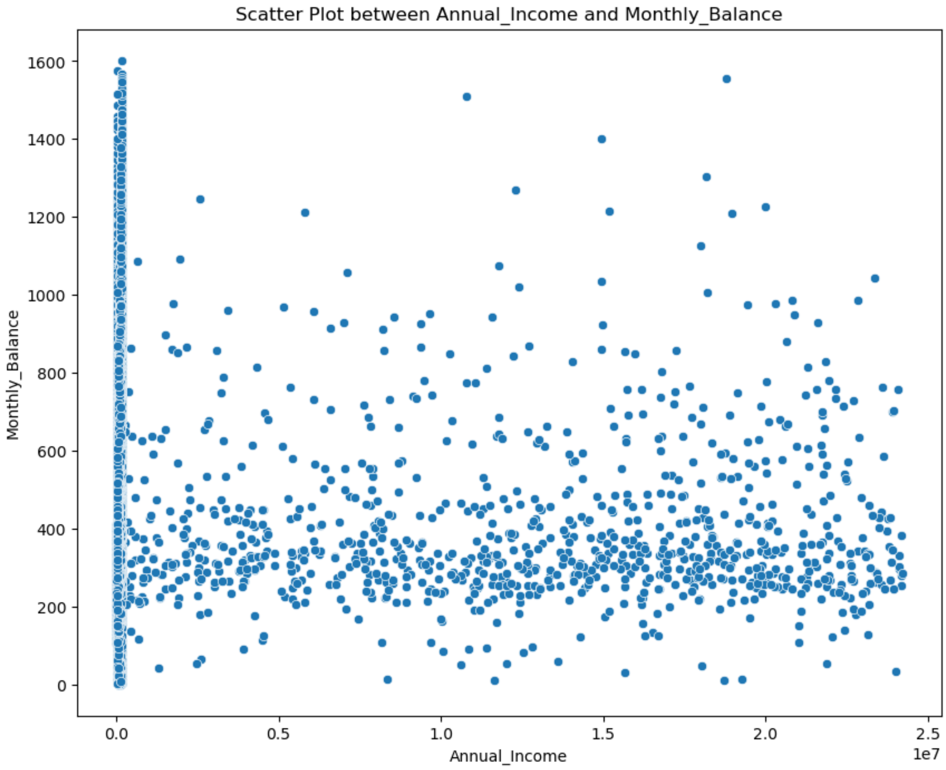
6) The image presented below is the count plot representation which is used to display the frequency distribution of a categorical variable – ‘Occupation’ and ‘Credit_Score’



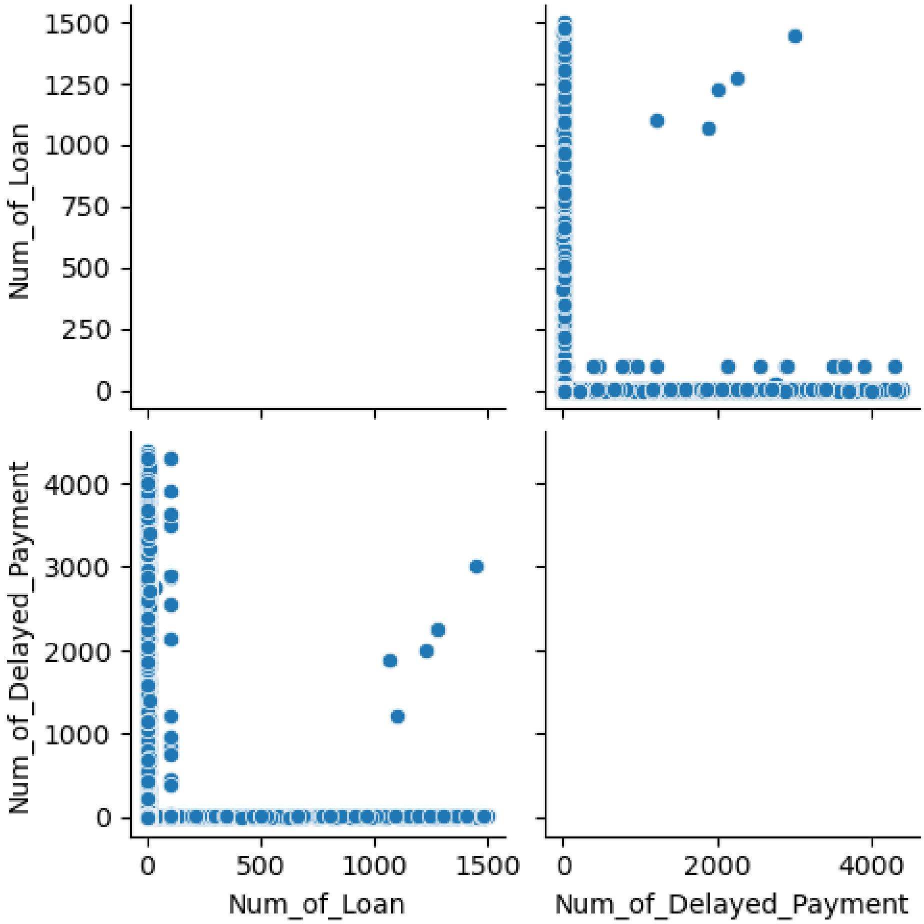


7) The image represented below displays a scatter plot which is a type of data visualization that is used to show the relationship between two numerical variables – Annual_Income and Num_of_Loan., Annual_Income and Monthly_Balance.

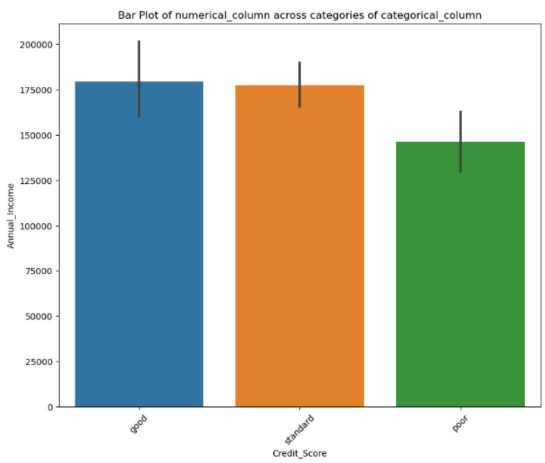


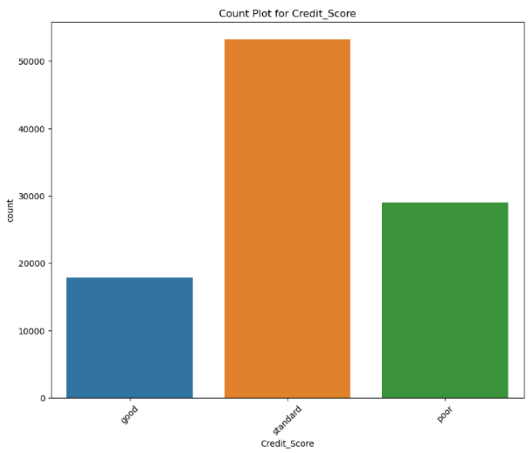


8) The image represents two scatter plots which are used to visualize the relation between pair of variables – Num_of_Loan and Num_of_Delayed_Payments

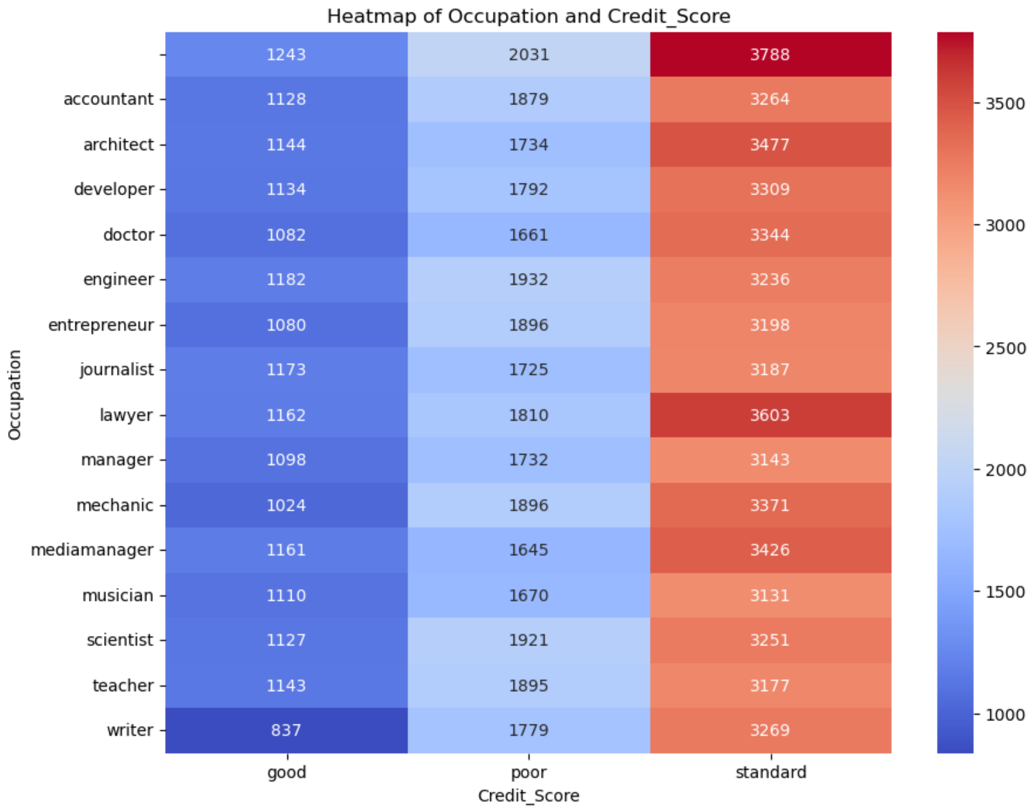


9) The below represented image displays a barplot that compares the average Annual_Income across different Credit_Score categories.





10) The below-provided image is a heat map() that visualizes the relationship between two categorical variables – ‘Occupation’ and ‘Credit_Score’.



11) The image represented below is a lower triangular heatmap() which is used in statistical analysis to represent the correlation matrix between different variables in a dataset.