Credit Risk Prediction

Classification Project

GROUP MEMBER

1. ARYAN VAKHARIA
2. D.BHARATH
3. A.VASUDEV
4. K.MANIKANTA
5. A.AAKASH

AURORA'S DEGREE AND P.G. COLLEGE
(B.SC. COMPUTER SCIENCE & DATA SCIENCE)

Project Background/Introduction

Credit risk prediction is crucial for financial institutions to assess the likelihood of borrowers defaulting on their loans. This section will explore the importance and challenges of credit risk prediction.

Objective:

Credit Risk Drivers

- Probability Of Default
- Loss Given Default
- Exposure At Default

Determines a borrower's ability to meet their debt obligations and the lender's aim when advancing credit.

Data:

- **Structured and Unstructured Data**

Credit risk prediction involves leveraging both structured data, such as financial statements, and unstructured data, such as customer behavior patterns.

- **External Data Sources**

Additional data sources like credit bureaus and economic indicators complement internal data to enhance the accuracy of credit risk prediction models.

- **Big Data Processing**

Dealing with large volumes of data requires efficient processing techniques, such as distributed computing or cloud-based solutions.

The data consists of 21 columns and 1000 observations.

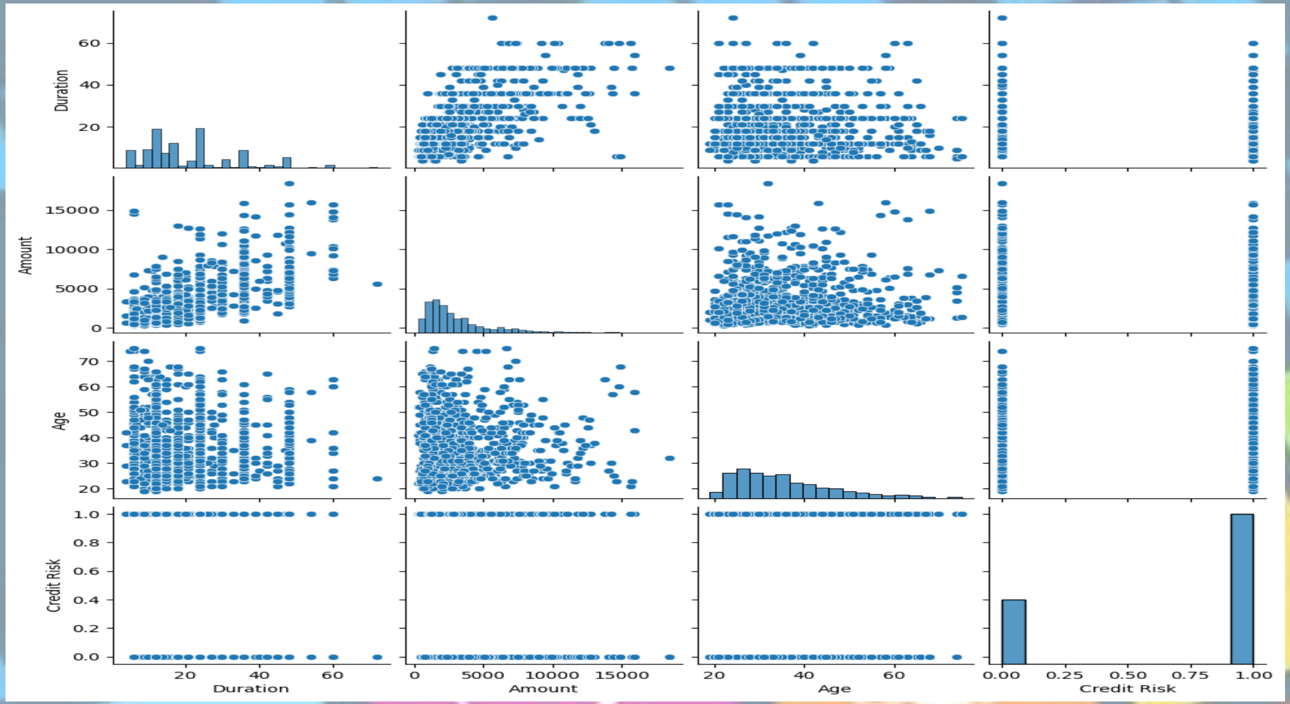
Data Cleaning:

- Removal of unwanted observations
- Fixing Structural errors
- Managing Unwanted outliers
- Handling missing data

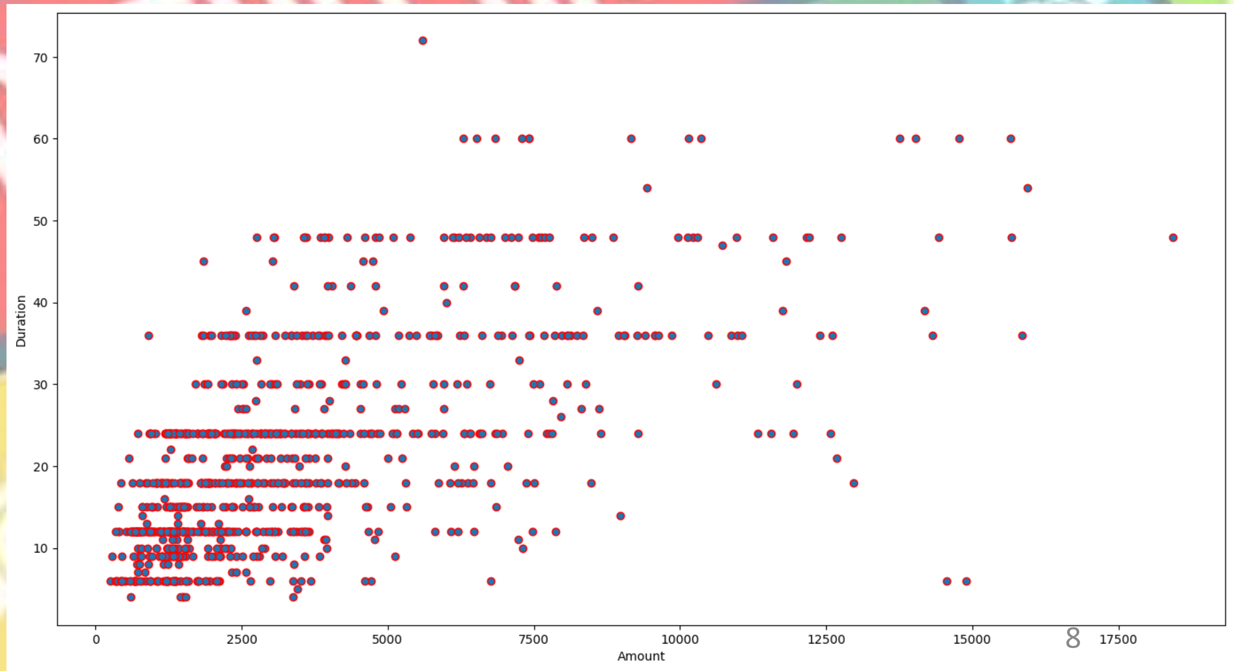
#	Column	Non-Null Count	Dtype
0	Status	1000 non-null	object
1	Duration	1000 non-null	int64
2	Credit History	1000 non-null	object
3	Purpose	1000 non-null	object
4	Amount	1000 non-null	int64
5	Savings	1000 non-null	object
6	Employment Duration	1000 non-null	object
7	Installment Rate	1000 non-null	object
8	Personal Status Sex	1000 non-null	object
9	Other Debtors	1000 non-null	object
10	Present Residence	1000 non-null	object
11	Property	1000 non-null	object
12	Age	1000 non-null	int64
13	Other Installment Plans	1000 non-null	object
14	Housing	1000 non-null	object
15	Number Credits	1000 non-null	object
16	Job	1000 non-null	object
17	People Liable	1000 non-null	object
18	Telephone	1000 non-null	object
19	Foreign Worker	1000 non-null	object
20	Credit Risk	1000 non-null	int64

Exploratory Data Analysis (EDA)

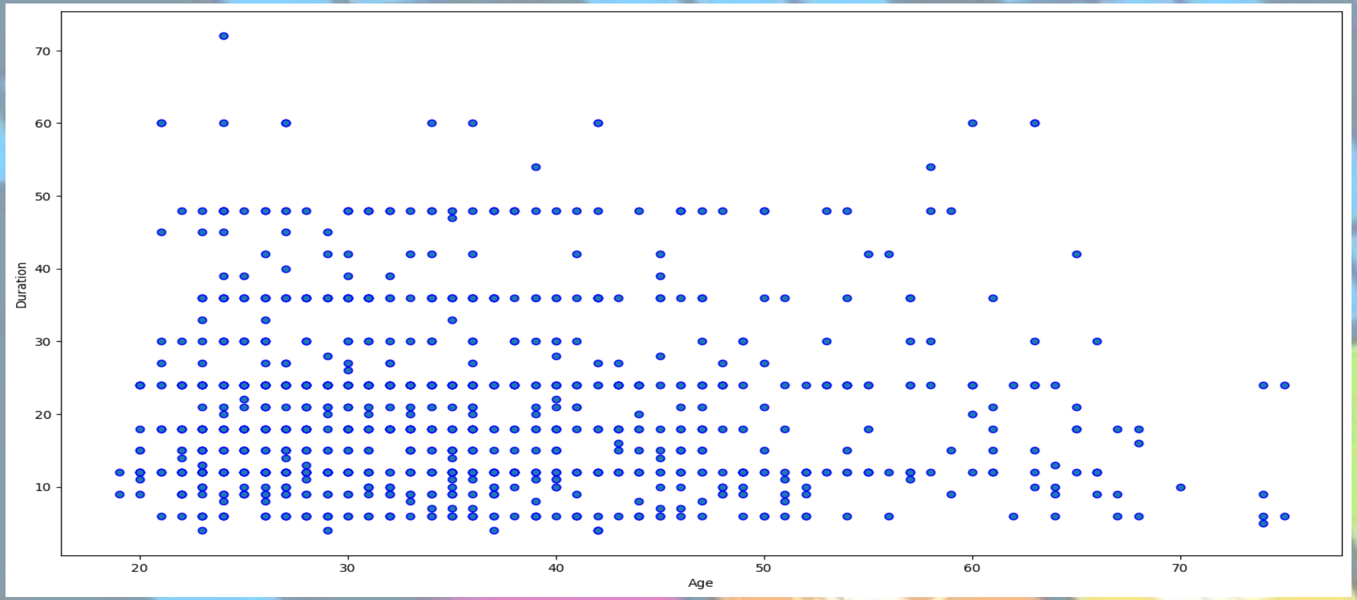
In EDA we find Heat map , Distribution polt , Box plot , Scatter plots , Pair plots and at last we have credit risk prediction Classification using EDA.



Pair plot

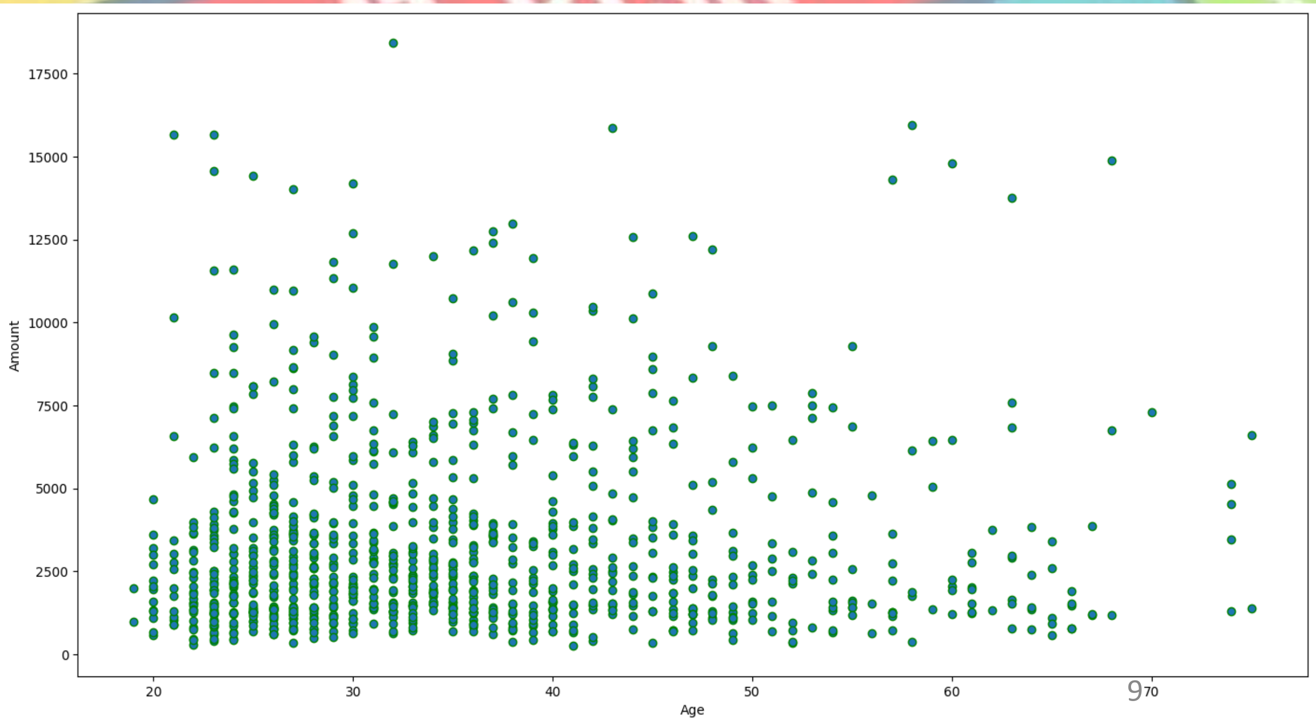


Scatter Plot of Duration and Amount

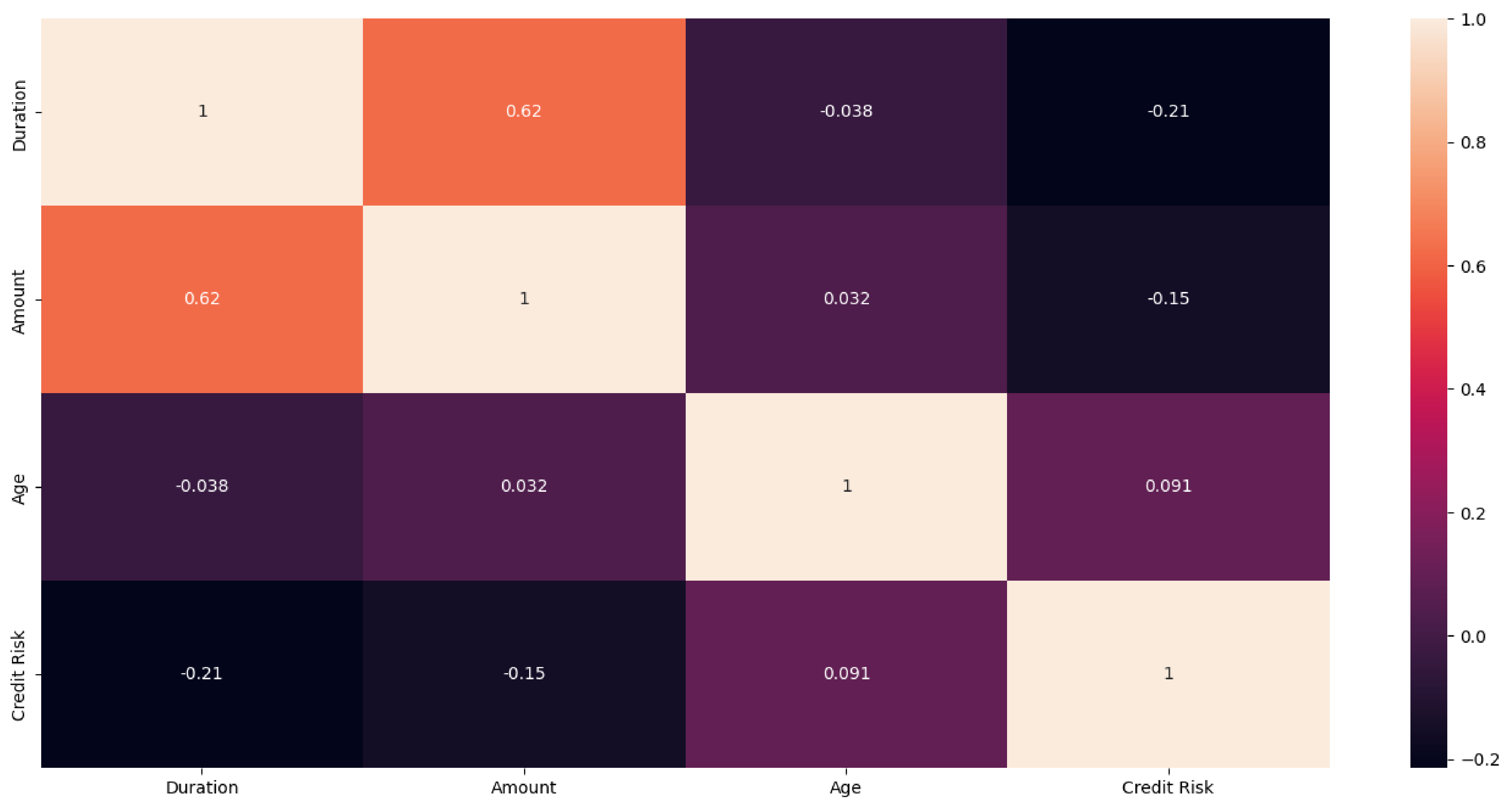


Scatter Plot of Duration and Age

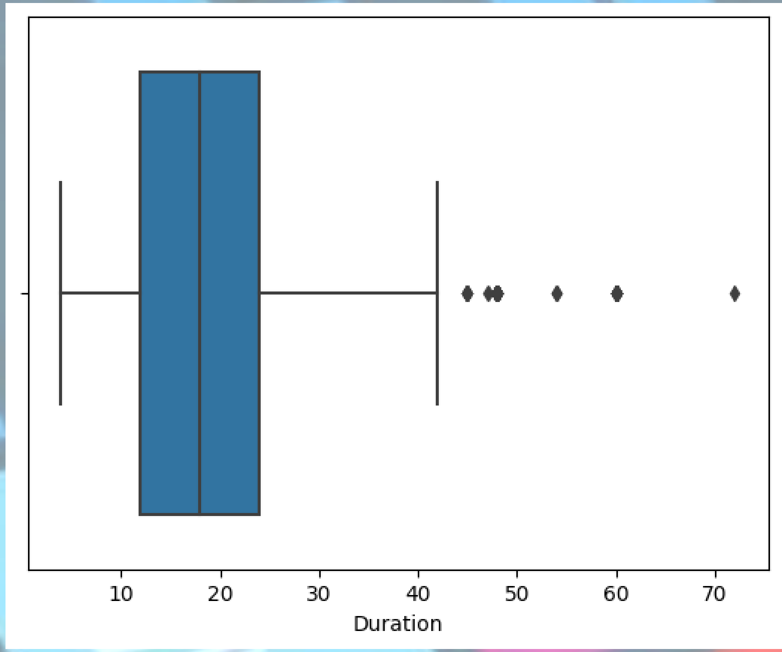
Scatter Plot of Amount and Age



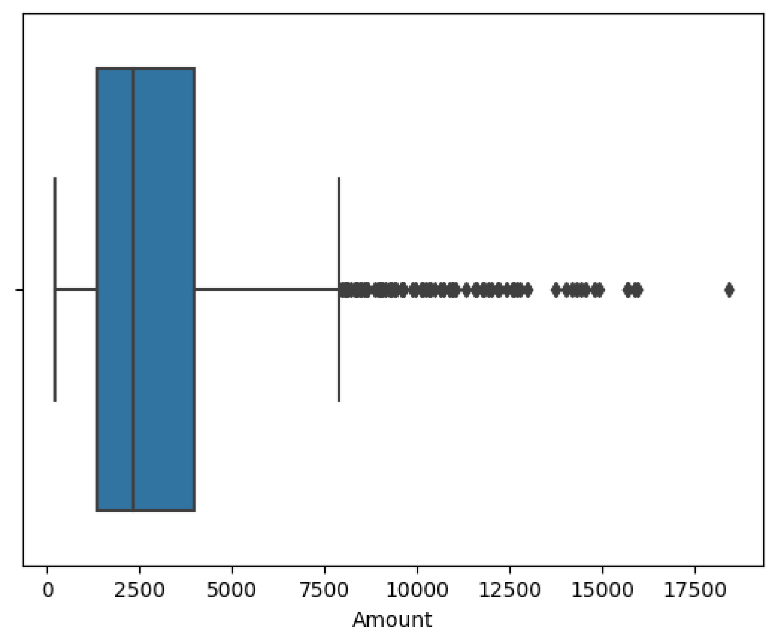
HEAT MAP:


Duration
Amount
Age
Credit Risk
1
0.62
-0.038
-0.21
0.62
1
0.032
-0.15
-0.038
0.032
1
0.091
-0.21
-0.15
0.091
1
Duration
Amount
Age
Credit Risk
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

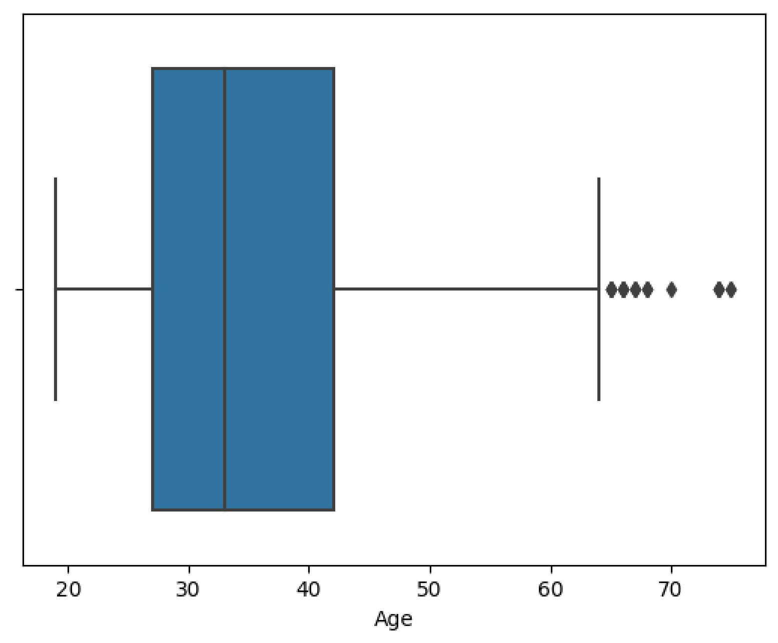
Box plot



Duration plot



Amount plot



Age plot

Interpretation from pair plot and heat map

- Age is highly positively correlated with the target variable.
- Duration is highly negatively correlated with the target variable.
- Duration has a strong correlation with Credit Risk.
- Duration and Amount are negatively correlated with the target variable (Credit Risk).
- Age is positively correlated with the target variable (Credit Risk).

Applying Dummy Variable

1) Dummy variables are used in statistical analysis, particularly in regression analysis, to handle categorical data or factors.
2) Categorical data represents categories, groups, or labels, rather than numerical values.
3) These variables need to be converted into a format that can be used in regression models, and this is where dummy variables come into play.

Purpose_car (new)	**Purpose _furniture/equipment**	**Purpose_ repairs**
Purpose_car (used)	**Purpose_others**	**Purpose_retraining**
Purpose_domestic appliances	**Purpose_radio/television**	**Purpose_vacation**

Machine Learning Algorithms

1 **Logistic regression**

Logistic regression aims to solve classification problems. It does this by predicting categorical outcomes, unlike linear regression that predicts a continuous outcome

2 **Decision tree**

A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks.

3 **Random forest**

Random forest is a commonly-used machine learning algorithm, which combines the output of multiple decision trees to reach a single result.

4 **K-Nearest neighbour**

The k-nearest neighbors , also known as KNN , is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point.

5

Ridge Logistic Regression

Ridge Regression adds a penalty term proportional to the square of the coefficients

6

K-Means Clustering

k-means clustering tries to group similar kinds of items in form of clusters.

7

XG Boost

XG Boost, which stands for Extreme Gradient Boosting, is a scalable, distributed gradient-boosted decision tree (GBDT) machine learning library

8

Support Vector Machine

A support vector machine (SVM) is a type of supervised learning algorithm used in machine learning to solve classification and regression tasks

Logistic regression

Train test ratio	Accuracy
65-35	71.14%
70-30	70%
70-25	70%
80-20	70.5%

Decision tree

Train test ratio	Accuracy
65-35	66.7%
70-30	66%
70-25	68%
80-20	67%

Random forest

Train test ratio	Accuracy
65-35	70.28%
70-30	70.33%
70-25	72.4%
80-20	74.5%

K – Nearest Neighbour (KNN)

Train test ratio	Accuracy
65-35	66.86%
70-30	66.86%
70-25	66%
80-20	66%

Ridge Logistic Regression

Train test ratio	Accuracy
65-35	66.86%
70-30	67%
70-25	66%
80-20	66%

K – Means Clustering

Train test ratio	Accuracy
65-35	66.86%
70-30	66.33%
70-25	66%
80-20	66%

XG Boost

Train test ratio	Accuracy
65-35	69.71%
70-30	68.67%
70-25	73.2%
80-20	71.5%

Support Vector Machine(SVM)

Train test ratio	Accuracy
65-35	70.29%
70-30	69.67%
70-25	69.6%
80-20	69.5%

A comparison between the implemented models

MODEL	**ACCURACY**
Logistic regression	71.14%
Decision tree	68%
Random forest	74.5%
K – Nearest Neighbour (KNN) (65-30)	66.86%
K – Nearest Neighbour (KNN) (70-30)	66.86%
Ridge Logistic Regression	67%
K – Means Clustering	66.86%
XG Boost	73.2%
Support Vector Machine(SVM)	70.29%

APPENDIX:

Applying Dummy Variable

```python
Elements = ['Purpose']
df = pd.get_dummies(df, columns = Elements, drop_first = True)
df.head()
```

	Status	Duration	Credit History	Amount	Savings	Employment Duration	Installment Rate	Personal Status Sex	Other Debtors	Present Residence	...	Credit Risk	Purpose_car (new)	Purpose_car (used)	Purpose_domestic appliances	Purpose_furniture/equipment	Purpose_others	Purpose_radio/television	Purpose_repairs	Purpose_retraining	Purpose_vacation
0	no checking account	18	all credits at this bank paid back duly	1049	unknown/no savings account	< 1 yr	< 20	female : non-single or male : single	none	>= 7 yrs	...	1	0	1	0	0	0	0	0	0	0
1	no checking account	9	all credits at this bank paid back duly	2799	unknown/no savings account	1 <= ... < 4 yrs	25 <= ... < 35	male : married/widowed	none	1 <= ... < 4 yrs	...	1	0	0	0	0	1	0	0	0	0
2	... < 0 DM	12	no credits taken/all credits paid back duly	841	... < 100 DM	4 <= ... < 7 yrs	25 <= ... < 35	female : non-single or male : single	none	>= 7 yrs	...	1	0	0	0	0	0	0	0	1	0
3	no checking account	12	all credits at this bank paid back duly	2122	unknown/no savings account	1 <= ... < 4 yrs	20 <= ... < 25	male : married/widowed	none	1 <= ... < 4 yrs	...	1	0	0	0	0	1	0	0	0	0
4	no checking account	12	all credits at this bank paid back duly	2171	unknown/no savings account	1 <= ... < 4 yrs	< 20	male : married/widowed	none	>= 7 yrs	...	1	0	0	0	0	1	0	0	0	0

Logistic Regression

```
model = LogisticRegression()
model.fit(xtrain, ytrain)
ypred = model.predict(xtest)
accuracy = accuracy_score(ytest, ypred)
mae = mean_absolute_error(ytest, ypred)
print('Accuracy: ', accuracy)
```

```
Accuracy:  0.705
```

Decision Tree

```
model = DecisionTreeClassifier()
model.fit(xtrain, ytrain)
ypred = model.predict(xtest)
accuracy = accuracy_score(ytest, ypred)
mae = mean_absolute_error(ytest, ypred)
print('Accuracy: ', accuracy)
```

```
Accuracy:  0.63
```

Random Forest

```
model = RandomForestClassifier(n_estimators = 100, random_state = 42)
model.fit(xtrain,ytrain)
ypred = model.predict(xtest)
accuracy = accuracy_score(ytest, ypred)
mae = mean_absolute_error(ytest, ypred)
print('Accuracy: ', accuracy)
```

```
Accuracy:  0.745
```

K-Nearest Neighbour (KNN)

```
k = 3
model = KNeighborsClassifier(n_neighbors = k)
model.fit(xtrain, ytrain)
ypred = model.predict(xtest)
accuracy = accuracy_score(ytest, ypred)
mae = mean_absolute_error(ytest, ypred)
print('Accuracy: ', accuracy)
```

```
Accuracy:  0.66
```

Ridge Logistic Regression

```
ridge_model = LogisticRegression(penalty = 'l2', C = 1.0)
ridge_model.fit(xtrain, ytrain)
ypred = model.predict(xtest)
accuracy = accuracy_score(ytest, ypred)
mae = mean_absolute_error(ytest, ypred)
print('Accuracy: ', accuracy)
```

```
Accuracy:  0.66
```

K-Means Clustering

```
k = 3
kmeans = KMeans(n_clusters = k)
clusters = kmeans.fit_predict(xtest)
ypred = model.predict(xtest)
accuracy = accuracy_score(ytest, ypred)
mae = mean_absolute_error(ytest, ypred)
print('Accuracy: ', accuracy)
```

```
Accuracy:  0.66
```

XG Boost

```
model = XGBClassifier()
model.fit(xtrain, ytrain)
ypred = model.predict(xtest)
accuracy = accuracy_score(ytest, ypred)
mae = mean_absolute_error(ytest, ypred)
print('Accuracy: ', accuracy)
```

```
Accuracy:  0.715
```

Support Vector Machine (SVM)

```
model = SVC(kernel='linear')
model.fit(xtrain, ytrain)
ypred = model.predict(xtest)
accuracy = accuracy_score(ytest, ypred)
mae = mean_absolute_error(ytest, ypred)
print('Accuracy: ', accuracy)
```

```
Accuracy:  0.695
```

CONCLUSION

- Eight machine learning algorithms were used for the Credit risk Prediction dataset: Logistic Regression, Decision Tree, Random Forest, K – Nearest Neighbour (KNN), Ridge Logistic Regression, K – Means Clustering, XG Boost, and Support Vector Machine (SVM).

- After conducting various analyses and implementing different algorithms, we discovered that the Random Forest Regression Algorithm yielded the best results. Specifically, when using an 80-20 train-test ratio, we obtained a Accuracy of 74.5%, which is the Highest Accuracy value compared to all the other errors we encountered.

- After analyzing the Credit Risk dataset, we can conclude that the Random Forest Regression Algorithm is the most effective model.

THANK YOU

from Group 2