Configuration Manual

MSc Research Project
Predictive Analytics for Credit Risk Assessment in Peer to Peer Lending Platforms.
MSc Fintech.

Femi Benjamin Obadimu
Student ID: 22244336

School of Computing
National College of Ireland

Supervisor: Brian Byrne

National College of Ireland

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Femi Benjamin Obadimu

Student ID: 22244336

Programme: Financial Technology **Year:** ………2024…………………..

Module: …PRACTICUM

Lecturer: Brian Bryne

Submission Due Date: 12 August 2024

Project Title: Predictive Analytics For Credit Risk Assessment in Peer to Peer Lending Platforms.

Word Count: ……………………………………… **Page Count:** ………………………………….…….………

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……………………………………… ………………………………………………………………………………

Date: ………………………………12 August 2024…

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	

Date:	
Penalty Applied (if applicable):	

Configuration Manual

Femi Benjamin Obadimu
Student ID: 22244336

1 Introduction

This configuration manual will include the hard ware and software platforms used during this research. During this research, three models including logistic regression, XG Boost, and Random Forest were built to analyse loan default on the Bondora peer-to-peer platform using the Python programming language on Google Colab.

2 System Configuration.

Your second section. Change the header and label to something appropriate. This section consist of the system and type of software setup used to achieve the goal and objectives of this research.

2.1 Hardware

A personal computer was utilized for the purpose of this project and the configuration is revealed below.

Device name	DESKTOP-5K0G5DO
Processor	Intel(R) Core(TM) i7-5600U CPU @ 2.60GHz 2.60 GHz
Installed RAM	8.00 GB (7.88 GB usable)
Device ID	2C366D28-54D1-4B0E-AD66-81AD0355E097
Product ID	00342-50367-95769-AAOEM
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

2.2 Software Configuration

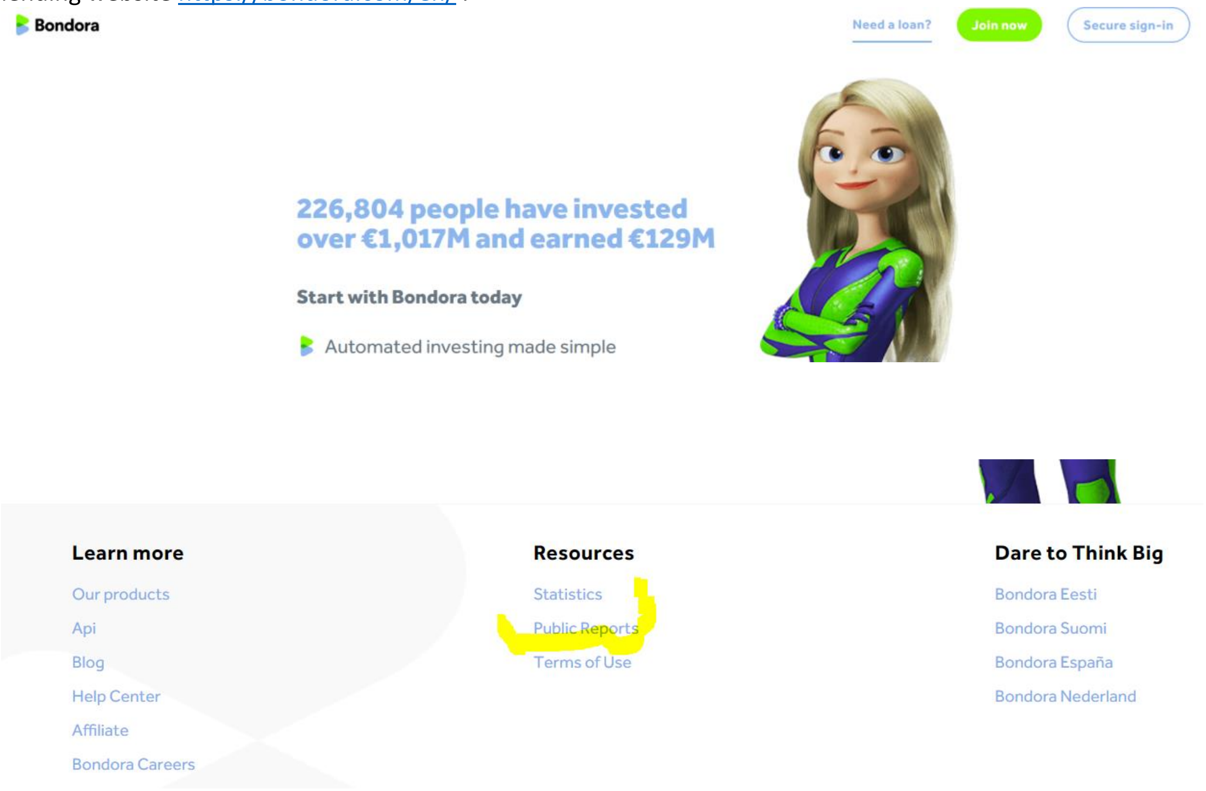
This includes the device's operating system which is listed below.

Edition	Windows 10 Pro
Version	22H2
Installed on	04/09/2023
OS build	19045.4651
Experience	Windows Feature Experience Pack 1000.19060.1000.0

For the purpose o this research, google colab cloud platform was used to build and execute all necessary python codes for the purpose of this research. The mozila firefox browser was used to set up the google colab due to available extensions on the browser.

3 Project Implementation.

3.1 Data Collection: The dataset used during this research was downloaded from the Bondora peer-to-peer lending website https://bondora.com/en/ .

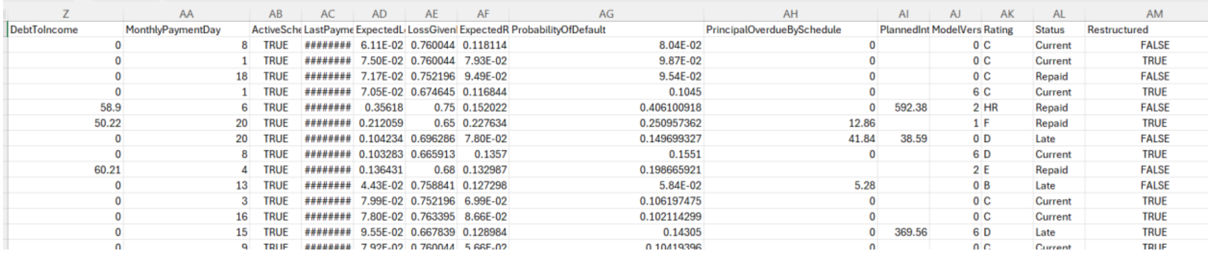


226,804 people have invested over €1,017M and earned €129M

Start with Bondora today

Automated investing made simple

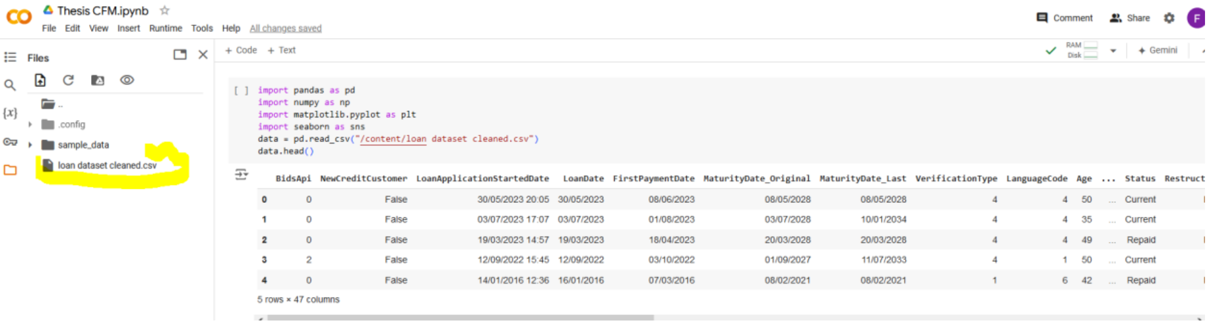
the data set consist of 15821 rows and 47 columns.



Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM
DebtToIncome	MonthlyPaymentDay	ActiveSche	LastPayme	ExpectedL	LossGivenI	ExpectedR	ProbabilityOfDefault	PrincipalOverdueBySchedule	PlannedInt	ModelVers	Rating	Status	Restructured
0	8	TRUE	########	6.11E-02	0.760044	0.118114	8.04E-02	0		0	C	Current	FALSE
0	1	TRUE	########	7.50E-02	0.760044	7.93E-02	9.87E-02	0		0	C	Current	TRUE
0	18	TRUE	########	7.17E-02	0.752196	9.49E-02	9.54E-02	0		0	C	Repaid	FALSE
0	1	TRUE	########	7.05E-02	0.674645	0.116844	0.1045	0		6	C	Current	TRUE
58.9	6	TRUE	########	0.35618	0.75	0.152022	0.406100918	0	592.38	2	HR	Repaid	FALSE
50.22	20	TRUE	########	0.212059	0.65	0.227634	0.250957362	12.86		1	F	Repaid	TRUE
0	20	TRUE	########	0.104234	0.696286	7.80E-02	0.149699327	41.84	38.59	0	D	Late	FALSE
0	8	TRUE	########	0.103283	0.665913	0.1357	0.1551	0		6	D	Current	TRUE
60.21	4	TRUE	########	0.136431	0.68	0.132987	0.198665921			2	E	Repaid	FALSE
0	13	TRUE	########	4.43E-02	0.758841	0.127298	5.84E-02	5.28		0	B	Late	FALSE
0	3	TRUE	########	7.99E-02	0.752196	6.99E-02	0.106197475	0		0	C	Current	TRUE
0	16	TRUE	########	7.80E-02	0.763395	8.66E-02	0.102114299	0		0	C	Current	TRUE
0	15	TRUE	########	9.55E-02	0.667839	0.128984	0.14305	0	369.56	6	D	Late	TRUE
0	9	TRUE	########	7.92E-02	0.760044	5.66E-02	0.10419396	0		0	C	Current	TRUE

Data preparation

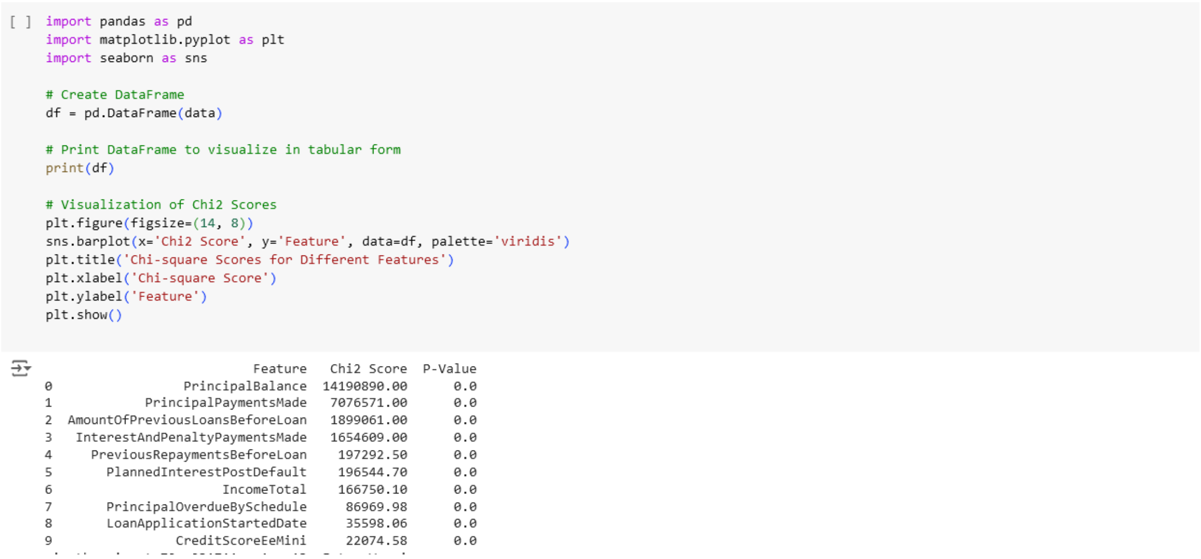
The data collected for this study was converted into a CSV file and uploaded on Google Colab for compatibility using python libraries including pandas, NumPy, matplotlib, and seaborn



```
[ ] import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    import seaborn as sns
    data = pd.read_csv("/content/loan dataset cleaned.csv")
    data.head()
```

	BidsApi	NewCreditCustomer	LoanApplicationStartedDate	LoanDate	FirstPaymentDate	MaturityDate_Original	MaturityDate_Last	VerificationType	LanguageCode	Age	...	Status	Restruct
0	0	False	30/05/2023 20:05	30/05/2023	08/06/2023	08/05/2028	08/05/2028	4	4	50	...	Current	
1	0	False	03/07/2023 17:07	03/07/2023	01/08/2023	03/07/2028	10/01/2034	4	4	35	...	Current	
2	0	False	19/03/2023 14:57	19/03/2023	18/04/2023	20/03/2028	20/03/2028	4	4	49	...	Repaid	
3	2	False	12/09/2022 15:45	12/09/2022	03/10/2022	01/09/2027	11/07/2033	4	1	50	...	Current	
4	0	False	14/01/2016 12:36	16/01/2016	07/03/2016	08/02/2021	08/02/2021	1	6	42	...	Repaid	

5 rows × 47 columns

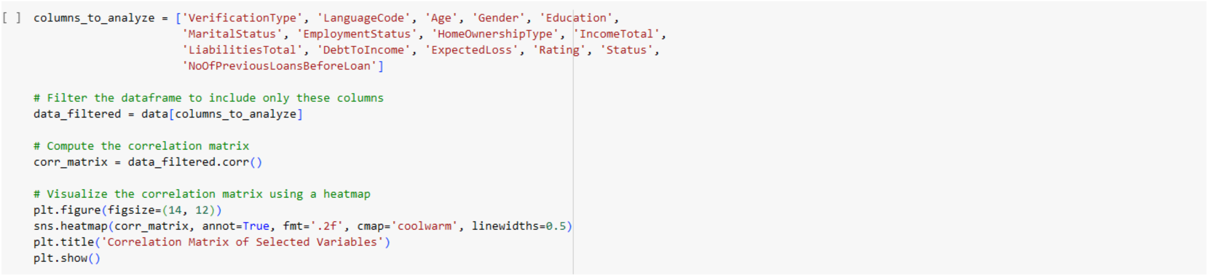


```
[ ] import pandas as pd
    import matplotlib.pyplot as plt
    import seaborn as sns

    # Create DataFrame
    df = pd.DataFrame(data)

    # Print DataFrame to visualize in tabular form
    print(df)

    # Visualization of Chi2 Scores
    plt.figure(figsize=(14, 8))
    sns.barplot(x='Chi2 Score', y='Feature', data=df, palette='viridis')
    plt.title('Chi-square Scores for Different Features')
    plt.xlabel('Chi-square Score')
    plt.ylabel('Feature')
    plt.show()
```

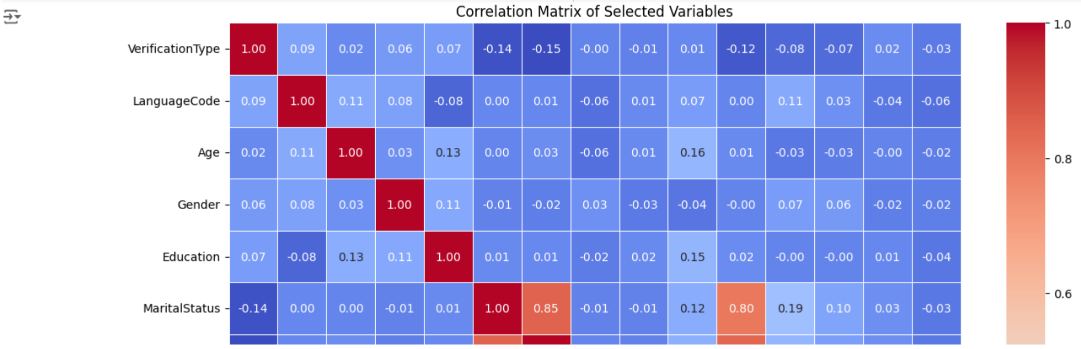
```
                        Feature  Chi2 Score  P-Value
0              PrincipalBalance 14190890.00      0.0
1         PrincipalPaymentsMade  7076571.00      0.0
2  AmountOfPreviousLoansBeforeLoan  1899061.00      0.0
3   InterestAndPenaltyPaymentsMade  1654609.00      0.0
4     PreviousRepaymentsBeforeLoan   197292.50      0.0
5       PlannedInterestPostDefault   196544.70      0.0
6                    IncomeTotal   166750.10      0.0
7       PrincipalOverdueBySchedule    86969.98      0.0
8     LoanApplicationStartedDate    35598.06      0.0
9               CreditScoreEeMini    22074.58      0.0
```

The code above shows how the chi square technique has been used to rank the predictive power of the variables in data set.

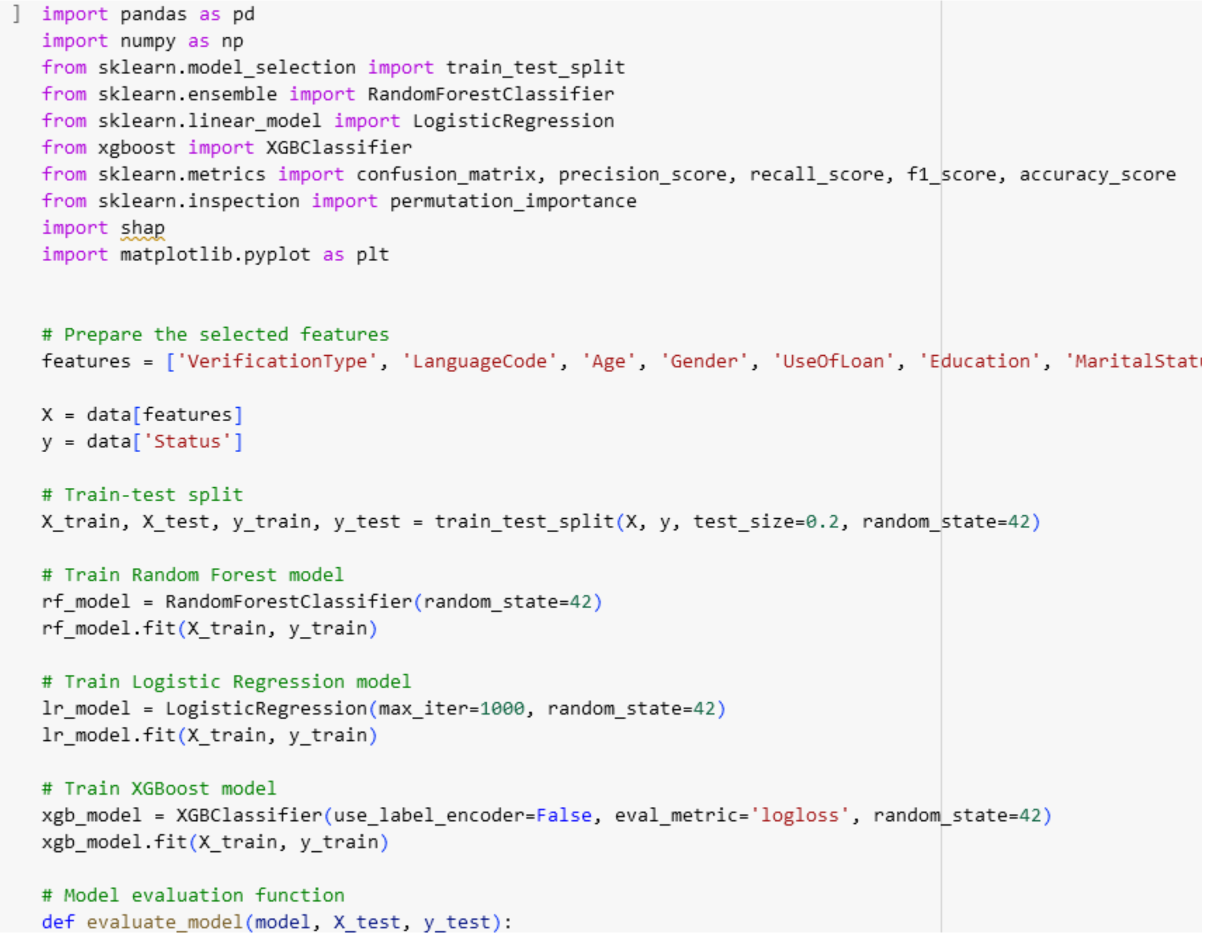
During the data preprocessing stage, correlation matrix code was executed to further analyse the linear relationship between variables as shown below.



```
[ ] columns_to_analyze = ['VerificationType', 'LanguageCode', 'Age', 'Gender', 'Education',
                          'MaritalStatus', 'EmploymentStatus', 'HomeOwnershipType', 'IncomeTotal',
                          'LiabilitiesTotal', 'DebtToIncome', 'ExpectedLoss', 'Rating', 'Status',
                          'NoOfPreviousLoansBeforeLoan']

    # Filter the dataframe to include only these columns
    data_filtered = data[columns_to_analyze]

    # Compute the correlation matrix
    corr_matrix = data_filtered.corr()

    # Visualize the correlation matrix using a heatmap
    plt.figure(figsize=(14, 12))
    sns.heatmap(corr_matrix, annot=True, fmt='.2f', cmap='coolwarm', linewidths=0.5)
    plt.title('Correlation Matrix of Selected Variables')
    plt.show()
```



Model Building, Evaluation and Visualization



```
] import pandas as pd
  import numpy as np
  from sklearn.model_selection import train_test_split
  from sklearn.ensemble import RandomForestClassifier
  from sklearn.linear_model import LogisticRegression
  from xgboost import XGBClassifier
  from sklearn.metrics import confusion_matrix, precision_score, recall_score, f1_score, accuracy_score
  from sklearn.inspection import permutation_importance
  import shap
  import matplotlib.pyplot as plt


  # Prepare the selected features
  features = ['VerificationType', 'LanguageCode', 'Age', 'Gender', 'UseOfLoan', 'Education', 'MaritalStat

  X = data[features]
  y = data['Status']

  # Train-test split
  X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

  # Train Random Forest model
  rf_model = RandomForestClassifier(random_state=42)
  rf_model.fit(X_train, y_train)

  # Train Logistic Regression model
  lr_model = LogisticRegression(max_iter=1000, random_state=42)
  lr_model.fit(X_train, y_train)

  # Train XGBoost model
  xgb_model = XGBClassifier(use_label_encoder=False, eval_metric='logloss', random_state=42)
  xgb_model.fit(X_train, y_train)

  # Model evaluation function
  def evaluate_model(model, X_test, y_test):
```

Python programming language was used to execute the codes in the research. During this research, essential python libraries were imported and installed before the models were built. The libraries utilized during for the purpose of model building includes:

1. **Numpy:** This library is used to create arrays, matrices. This python library is used majorly for mathematical computations of models.
2. **Pandas:** Panda library is used to clean and analyse data set uploaded to a python library.

Model Evaluation and Visualizations:

The models built for the purpose of this research are trained and tested by testing and splitting them to achieve a desirable result. The following libraries were used to execute this commands in python via the use of colab.

3. **Sklearn metric**: This library is used to import confusion matrix, precision score, F1 score, and accuracy score for the purpose of this study.
4. **Seaborn:** This is a python library used to visualize data based on matplotlib. Seaborn is used to draw informative statistical graphics
5. **Sklearn ensemble:** this library was used to import random forest classifier.
6. **Sk learn linear model:** this library was used to import logistic regression
7. **Sklearn Inspection**: This python library was used to import the permutation importance used to rank variables in the data set
8. **Matplotlib.pyplot:** This is used to generate plots, bar charts, histogram for adequate Exploratory Data Analysis (EDA).
9. **SHAP:** SHApley Additive Explanations is a game theoretic approach used to explain the output of any machine learning model SHAP is a unified framework for interpreting machine learning models.