CAPSTONE PROJECT - 3
POWER BI

Computer Hardware Software Workshop
COCSC19

Submitted by:-
Kanishk Kumar (2021UCS1627)
Shubham Yadav (2021UCS1630)
Kashish Goel (2021UCS1633)
CSE Section - 2
3rd Year

Objective:-

1. Explore Power View, Power Query
2. Visualize the result of any Machine Learning algorithm on any dataset of your choice in PowerBI.

1.Explore Power View, Power Query

Theory:

Power View and Power Query are two integral components of Microsoft's Power BI suite, aimed at data visualization and data transformation respectively. Here's a theoretical overview of both:

Power Query:

- Power Query is primarily focused on data transformation and preparation. It allows users to connect to various data sources, transform the data, and then load it into Power BI for analysis and reporting.
- Power Query provides a user-friendly interface for performing ETL (Extract, Transform, Load) operations on data. Users can perform tasks such as merging data from different sources, filtering rows, splitting columns, creating custom calculations, and more.
- Its intuitive interface and powerful transformation capabilities enable users to clean and shape data easily, even if it's coming from diverse and messy sources. This results in more accurate and reliable analyses downstream.

Power View:

- Power View is designed for interactive data visualization and exploration. It allows users to create visually compelling reports and dashboards that enable stakeholders to gain insights from the data.
- Power View provides a drag-and-drop interface for creating various types of visualizations such as charts, graphs, maps, and tables. Users can also add interactive features like slicers and filters to enable dynamic exploration of the data.
- Power View empowers users to communicate insights effectively through interactive and visually appealing reports. Its integration with Power BI allows for seamless sharing and collaboration, enabling decision-makers to make data-driven decisions.

a)Create a table Employee(empid, gender, department, salary, country, year_of_joining) connect to Employee data file.

empid	gender	departmer	salary	country	year_of_joining
1	Male	Sales	52000	USA	2016
2	Female	Marketing	61000	Canada	2019
3	Male	IT	72000	UK	2018
4	Female	HR	56000	Australia	2017
5	Male	Finance	67000	Germany	2020
6		Operation:	49000	France	2015
7	Female	Research	73000	Japan	2013
8	Male	Engineerin	76000	China	2012
9	Female		53000	Brazil	2018
10	Male	Manufactu	69000	India	2016
11			60000	Spain	2017
12	Male	Logistics	61000	Mexico	2019
13	Female	Legal	68000	Italy	2020
14	Male	Research	72000	South Afric	2018
15	Female	IT	64000	Sweden	2015

b)Remove missing gender and department values.

Query:

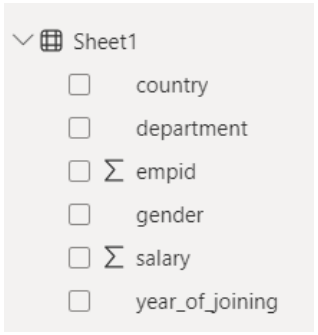
```
Table.SelectRows(#"Changed Type", each ([gender] <> null) and ([department] <> null))
```

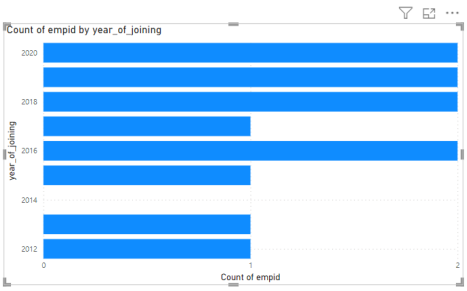
	empid	gender	department	salary	country	year_of_joining
1	1	Male	Sales	52000	USA	2016
2	2	Female	Marketing	61000	Canada	2019
3	3	Male	IT	72000	UK	2018
4	4	Female	HR	56000	Australia	2017
5	5	Male	Finance	67000	Germany	2020
6	7	Female	Research	73000	Japan	2013
7	8	Male	Engineering	76000	China	2012
8	10	Male	Manufacturing	69000	India	2016
9	12	Male	Logistics	61000	Mexico	2019
10	13	Female	Legal	68000	Italy	2020
11	14	Male	Research	72000	South Africa	2018
12	15	Female	IT	64000	Sweden	2015

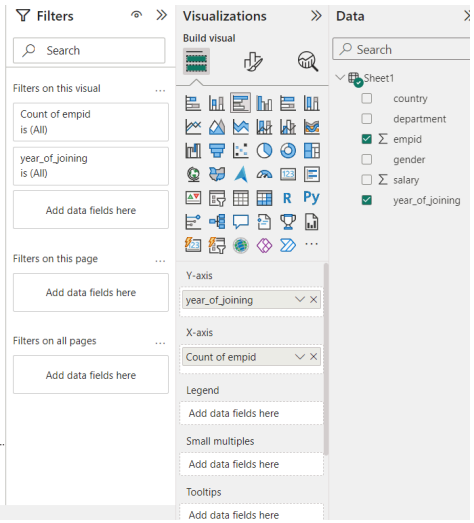
c)Extract year_of_joining column and visualize number of employees w.r.t year of experience in the company.

Query:

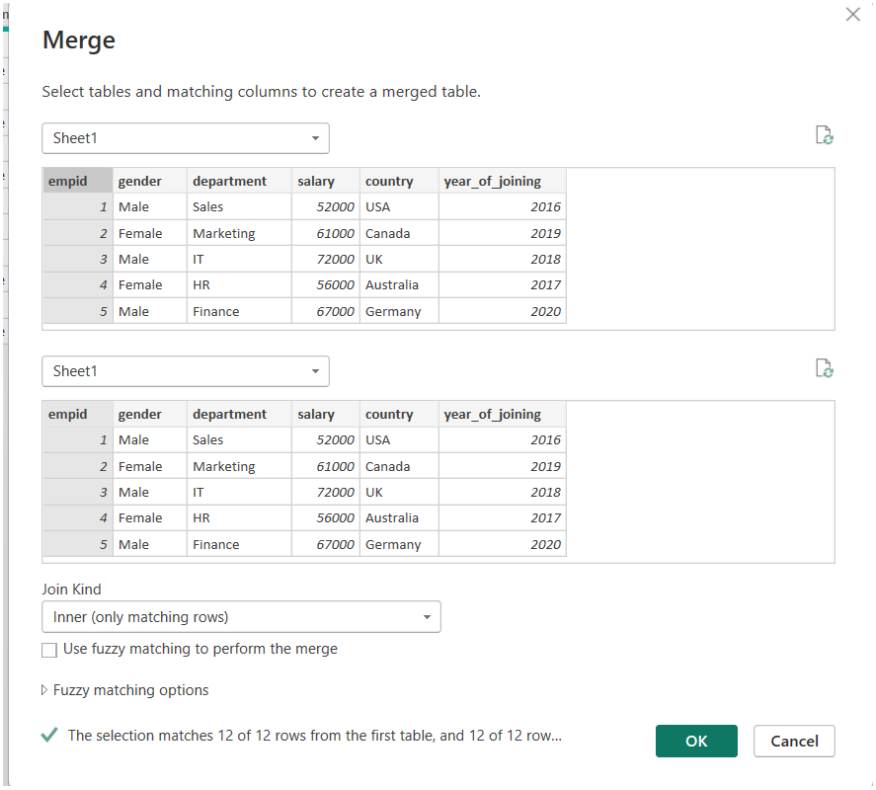
Table.SelectColumns(#"Filtered Rows",{"year_of_joining"})





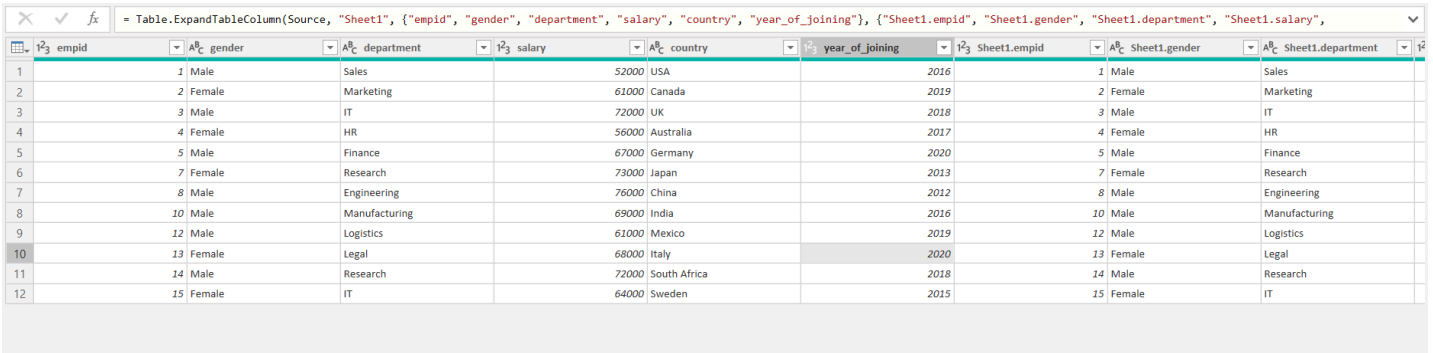


d)Perform self-join using Power Query.



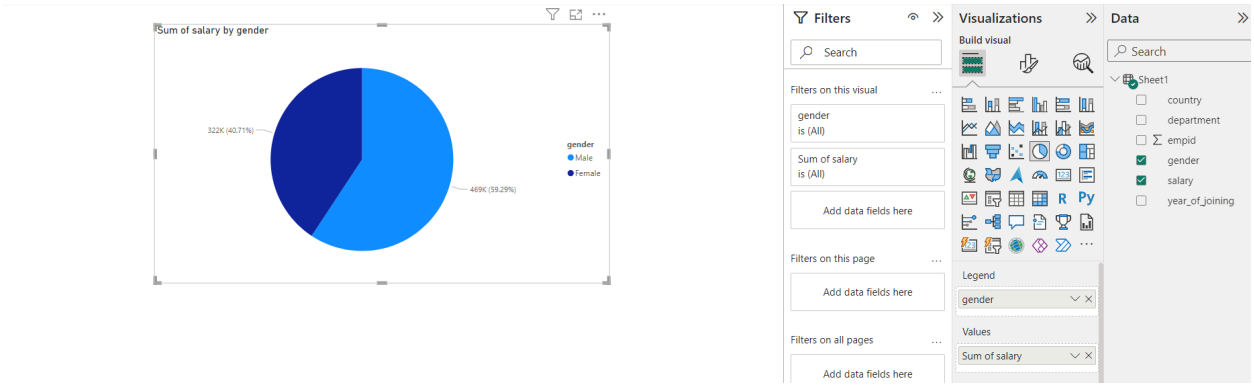
Query:

Table.ExpandTableColumn(Source, "Sheet1", {"empid", "gender", "department", "salary", "country", "year_of_joining"}, {"Sheet1.empid", "Sheet1.gender", "Sheet1.department", "Sheet1.salary", "Sheet1.country", "Sheet1.year_of_joining"})



	empid	gender	department	salary	country	year_of_joining	Sheet1.empid	Sheet1.gender	Sheet1.department
1	1	Male	Sales	52000	USA	2016	1	Male	Sales
2	2	Female	Marketing	61000	Canada	2019	2	Female	Marketing
3	3	Male	IT	72000	UK	2018	3	Male	IT
4	4	Female	HR	56000	Australia	2017	4	Female	HR
5	5	Male	Finance	67000	Germany	2020	5	Male	Finance
6	7	Female	Research	73000	Japan	2013	7	Female	Research
7	8	Male	Engineering	76000	China	2012	8	Male	Engineering
8	10	Male	Manufacturing	69000	India	2016	10	Male	Manufacturing
9	12	Male	Logistics	61000	Mexico	2019	12	Male	Logistics
10	13	Female	Legal	68000	Italy	2020	13	Female	Legal
11	14	Male	Research	72000	South Africa	2018	14	Male	Research
12	15	Female	IT	64000	Sweden	2015	15	Female	IT

e)Aggregate salary with gender and Visualize using Pie chart



2. Visualize the result of any Machine Learning algorithm on any dataset of your choice in PowerBI

Dataset:-

The dataset states the Loan Approval Status of people in relation to their Marital Status,Dependence,Educational Qualification,Employment ,Loan Amount,Credit History and Property Area.

Loan_ID	Gender	Married	Dependen	Education	Self_Empl	ApplicantI	Coapplicar	LoanAmou	Loan_Amc	Credit_His	Property_/	Loan_Status
LP001003	Male	Yes	1	Graduate	No	4583	1508	128	360	1	Rural	N
LP001005	Male	Yes	0	Graduate	Yes	3000	0	66	360	1	Urban	Y
LP001006	Male	Yes	0	Not Gradu	No	2583	2358	120	360	1	Urban	Y
LP001008	Male	No	0	Graduate	No	6000	0	141	360	1	Urban	Y
LP001013	Male	Yes	0	Not Gradu	No	2333	1516	95	360	1	Urban	Y
LP001024	Male	Yes	2	Graduate	No	3200	700	70	360	1	Urban	Y
LP001027	Male	Yes	2	Graduate		2500	1840	109	360	1	Urban	Y
LP001029	Male	No	0	Graduate	No	1853	2840	114	360	1	Rural	N
LP001030	Male	Yes	2	Graduate	No	1299	1086	17	120	1	Urban	Y
LP001032	Male	No	0	Graduate	No	4950	0	125	360	1	Urban	Y
LP001034	Male	No	1	Not Gradu	No	3596	0	100	240		Urban	Y
LP001036	Female	No	0	Graduate	No	3510	0	76	360	0	Urban	N
LP001038	Male	Yes	0	Not Gradu	No	4887	0	133	360	1	Rural	N
LP001041	Male	Yes	0	Graduate		2600	3500	115		1	Urban	Y
LP001043	Male	Yes	0	Not Gradu	No	7660	0	104	360	0	Urban	N
LP001047	Male	Yes	0	Not Gradu	No	2600	1911	116	360	0	Semiurban	N
LP001050		Yes	2	Not Gradu	No	3365	1917	112	360	0	Rural	N
LP001068	Male	Yes	0	Graduate	No	2799	2253	122	360	1	Semiurban	Y
LP001073	Male	Yes	2	Not Gradu	No	4226	1040	110	360	1	Urban	Y
LP001086	Male	No	0	Not Gradu	No	1442	0	35	360	1	Urban	N
LP001087	Female	No	2	Graduate		3750	2083	120	360	1	Semiurban	Y
LP001095	Male	No	0	Graduate	No	3167	0	74	360	1	Urban	N
LP001097	Male	No	1	Graduate	Yes	4692	0	106	360	1	Rural	N
LP001098	Male	Yes	0	Graduate	No	3500	1667	114	360	1	Semiurban	Y

a)Training Machine Learning Model

Training a Decision Tree Classifier from the dataset and saving the predicted data based on the trained model along with the actual data to a new file.
The Loan approval status is predicted based on Marital Status,Dependence,Educational Qualification,Employment ,Loan Amount,Credit History and Property Area.

Decision Tree Classifier:
Decision tree classifiers are machine learning algorithms that make decisions based on input features, recursively partitioning data into regions. They're easy to interpret and handle both numerical and categorical data. However, they can overfit and be sensitive to data variations. Ensemble methods like Random Forests mitigate these issues by combining multiple trees for more robust predictions.

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix

# Load the loan prediction data
data = pd.read_csv('/content/loan_data.csv')

# Remove rows with missing values
data.dropna(inplace=True)

# Separate features (X) and target variable (y)
X = data.drop('Loan_Status', axis=1)
y = data['Loan_Status']

# Encode categorical variables (if needed)
X = pd.get_dummies(X)

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initialize the decision tree classifier
clf = DecisionTreeClassifier(random_state=42)

# Train the classifier on the training data
clf.fit(X_train, y_train)

# Make predictions on the testing data
y_pred = clf.predict(X_test)
```

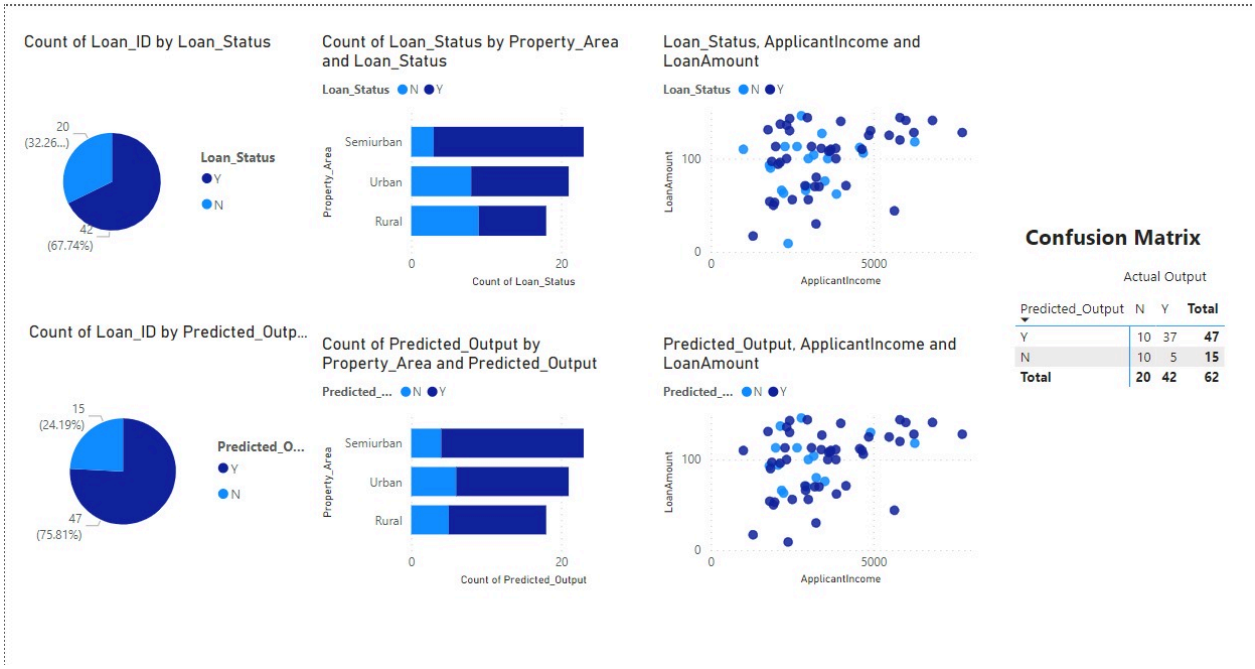
```python
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print('Accuracy:', accuracy)

print('\nClassification Report:')
print(classification_report(y_test, y_pred))

print('\nConfusion Matrix:')
print(confusion_matrix(y_test, y_pred))

# Add a column for predicted outputs to the DataFrame
data['Predicted_Output'] = clf.predict(X)

# Filter the DataFrame to include only rows from the testing set
tested_data = data.loc[X_test.index]

# Save the DataFrame with actual and predicted values for the tested data to a new CSV file
tested_data.to_csv('tested_loan_prediction_actual_vs_predicted_with_all_columns.csv', index=False)
```

```
Accuracy: 0.7580645161290323

Classification Report:
              precision    recall  f1-score   support

           N       0.67      0.50      0.57        20
           Y       0.79      0.88      0.83        42

    accuracy                           0.76        62
   macro avg       0.73      0.69      0.70        62
weighted avg       0.75      0.76      0.75        62
```

Saved file which has both Actual and Predicted Output:-

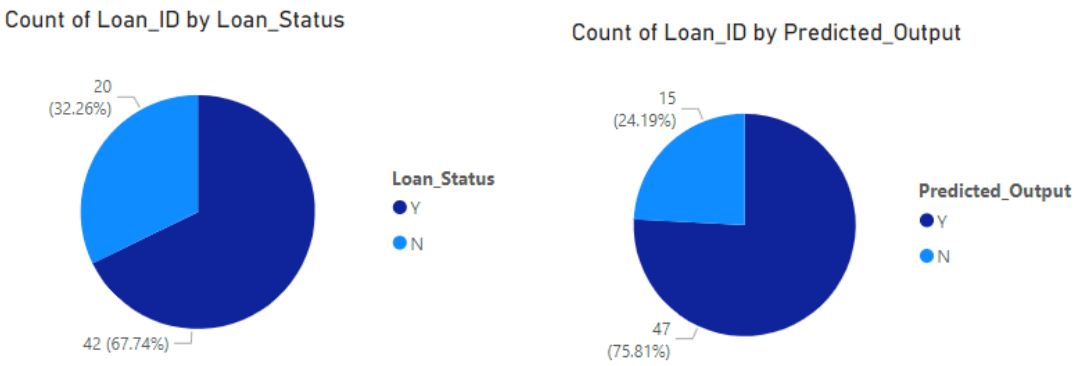
Loan_ID	Gender	Married	Dependen	Education	Self_Empl	ApplicantI	Coapplicar	LoanAmou	Loan_Amc	Credit_His	Property_A	Loan_Stat	Predicted_Output
LP002840	Female	No	0	Graduate	No	2378	0	9	360	1	Urban	N	Y
LP001036	Female	No	0	Graduate	No	3510	0	76	360	0	Urban	N	N
LP001343	Male	Yes	0	Graduate	No	1759	3541	131	360	1	Semiurban	Y	Y
LP001384	Male	Yes	3+	Not Gradu	No	2071	754	94	480	1	Semiurban	Y	N
LP001144	Male	Yes	0	Graduate	No	5821	0	144	360	1	Urban	Y	Y
LP001431	Female	No	0	Graduate	No	2137	8980	137	360	0	Semiurban	Y	N
LP001639	Female	Yes	0	Graduate	No	3625	0	108	360	1	Semiurban	Y	Y
LP002180	Male	No	0	Graduate	Yes	6822	0	141	360	1	Rural	Y	Y
LP002517	Male	Yes	1	Not Gradu	No	2653	1500	113	180	0	Rural	N	N
LP001265	Female	No	0	Graduate	No	3846	0	111	360	1	Semiurban	Y	Y
LP001520	Male	Yes	0	Graduate	No	4860	830	125	360	1	Semiurban	Y	Y
LP002006	Female	No	0	Graduate	No	2507	0	56	360	1	Rural	Y	Y
LP002893	Male	No	0	Graduate	No	1836	33837	90	360	1	Urban	N	Y
LP002978	Female	No	0	Graduate	No	2900	0	71	360	1	Rural	Y	Y
LP002807	Male	Yes	2	Not Gradu	No	3675	242	108	360	1	Semiurban	Y	Y

b)Visualizing the model



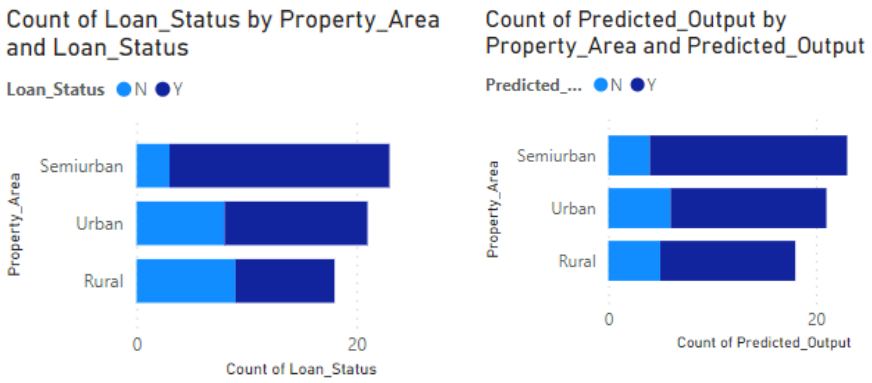
<u>i) Distribution of Loan Status in Actual result and predicted results</u>

The plot shows the share of loans approved and the share of loans predicted to be approved.



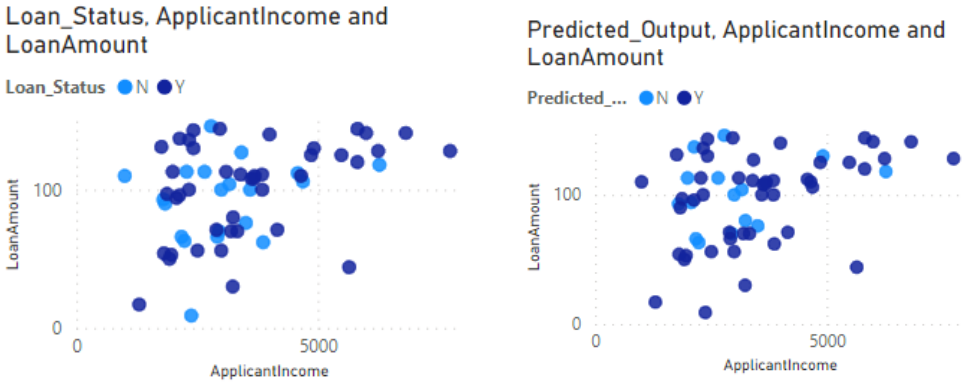
ii) The share of Loan Status approved by Property Area

The plot shows the distribution of load status approved and prediction in relation to the the area of property.



iii) Scatter Plot for Loan Amount vs Applicant Income

The scatter plot shows the Loan Amount vs Applicant Income along with the distribution of Loan Status.



iv) Confusion Matrix

A confusion matrix is a table used in machine learning to evaluate the performance of a classification model. It compares predicted and actual class

labels, showing counts of true positives, false positives, true negatives, and false negatives.

- True Positives (TP): Instances that are correctly predicted as positive.
- False Positives (FP): Instances that are incorrectly predicted as positive when they are actually negative.
- True Negatives (TN): Instances that are correctly predicted as negative.
- False Negatives (FN): Instances that are incorrectly predicted as negative when they are actually positive.

Confusion Matrix

	Actual Output		
Predicted_Output	N	Y	**Total**
N	10	5	**15**
Y	10	37	**47**
Total	**20**	**42**	**62**