Name Harshita Sharma
Sub. Data Mining
Branch:- AIDS-B
Enroll no.:- 21117711922 School

योगः कर्मसु कौशलम्
IN PURSUIT OF PERFECTION

VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS
Grade A++ Accredited Institution by NAAC
NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;
Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE
An ISO 9001:2015 Certified Institution

SCHOOL OF ENGINEERING & TECHNOLOGY

BTECH Programme: AIDS-B

Course Title: Data Mining

Course Code: AIDS305

Submitted To

Dr. Lakshita Aggarwal

Submitted By

Name: Harshita Sharma
Enrollment No:21117711922

VIPS
योगः कर्मसु कौशलम्
IN PURSUIT OF PERFECTION

VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS
Grade A++ Accredited Institution by NAAC
NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;
Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE
An ISO 9001:2015 Certified Institution

SCHOOL OF ENGINEERING & TECHNOLOGY

BTECH Programme: AIDS-B

Course Title: Data Mining

Course Code: AIDS305

Submitted To

Dr. Lakshita Aggarwal

Submitted By

Name: Harshita Sharma
Enrollment No:21117711922

VIPS
योगः कर्मसु कौशलम्
IN PURSUIT OF PERFECTION

VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS
Grade A++ Accredited Institution by NAAC
NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;
Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE
An ISO 9001:2015 Certified Institution
SCHOOL OF ENGINEERING & TECHNOLOGY

VISION OF INSTITUTE

To be an educational institute that empowers the field of engineering to build a sustainable future by providing quality education with innovative practices that supports people, planet and profit.

MISSION OF INSTITUTE

To groom the future engineers by providing value-based education and awakening students' curiosity, nurturing creativity and building capabilities to enable them to make significant contributions to the world.

VIPS
योगः कर्मसु कौशलम्
IN PURSUIT OF PERFECTION

VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS
Grade A++ Accredited Institution by NAAC
NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC.
Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE
An ISO 9001:2015 Certified Institution
SCHOOL OF ENGINEERING & TECHNOLOGY

INDEX

Experiment Name	Date	Marks			Remark	Updated Marks	Faculty Signature
		Laboratory Assessment (15 Marks)	Class Participation (5 Marks)	Viva (5 Marks)			
Intro & installation of WEKA	5/8/24	15	5	5			Checked 23/8/24
Data preprocessing, cleaning, integration transformation on ARFF files using WEKA	12/8/24	13	4	4			Checked 9/9/25
Apply association rule mining on dataset using WEKA	12/8/24	14	4	4			Checked 20/9/24
Apply visualisation on data using WEKA	9/9/24	15	4	5			Checked 20/9/24
Apply clustering on dataset using WEKA	9/9/24	14	4	4			Checked 23/9/24
Apply classification on dataset using WEKA.	23/9/24	15	4	4			Checked 23/9/24
Evaluate the performance of classification techniques using diff parameters.	23/9/24	14	4	5			Checked 5/11/24

VIPS
योगः कर्मसु कौशलम्
IN PURSUIT OF PERFECTION

VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES - TECHNICAL CAMPUS
Grade A++ Accredited Institution by NAAC
NBA Accredited for MCA Programme; Recognized under Section 2(f) by UGC;
Affiliated to GGSIP University, Delhi; Recognized by Bar Council of India and AICTE
An ISO 9001:2015 Certified Institution
SCHOOL OF ENGINEERING & TECHNOLOGY

INDEX

Experiment Name	Date	Marks			Remark	Updated Marks	Faculty Signature
		Laboratory Assessment (15 Marks)	Class Participation (5 Marks)	Viva (5 Marks)			
Implementation of Bagging and boosting techniques in ARFF files using JDCKA	4/11/24	14	4	5		8	
Apply the concept of voting ensemble method to ARFF files and Compare the result with single classifiers.	4/11/24	13	4	4			
Study of DBMiner tool	8/11/24	14	4	3			
Project: E-commerce customer retention: A Data driven Analysis	8/11/24	14	4	5			

EXPERIMENT-1

AIM: Introduction & Installation of WEKA tools.

Theory: Weka environment for Knowledge analysis (WEKA)

It is a collection of machine learning & data ~~science~~ analysis free software licensed under the GNU General Public License.

In other words, WEKA is a collection of ML algorithms for data mining tasks.

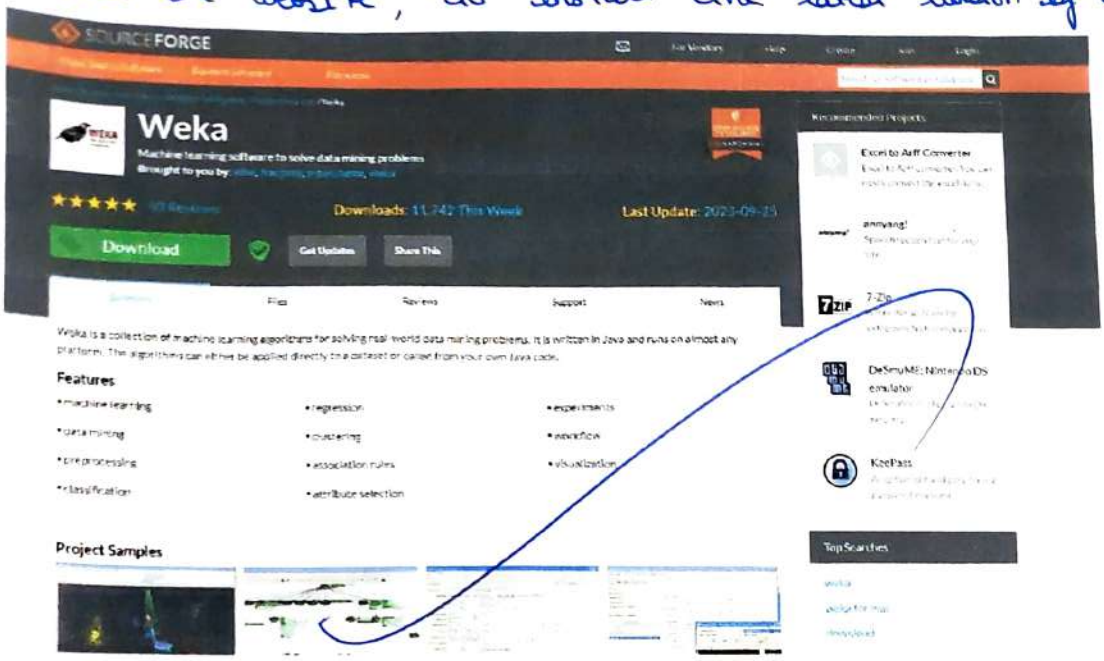
It contains tools for data pre-processing, classification, regression, clustering association rules & visualisation.

Weka provides the implementation of several algorithms one would select an algorithm of our choice, set the desired parameters and run it on dataset.

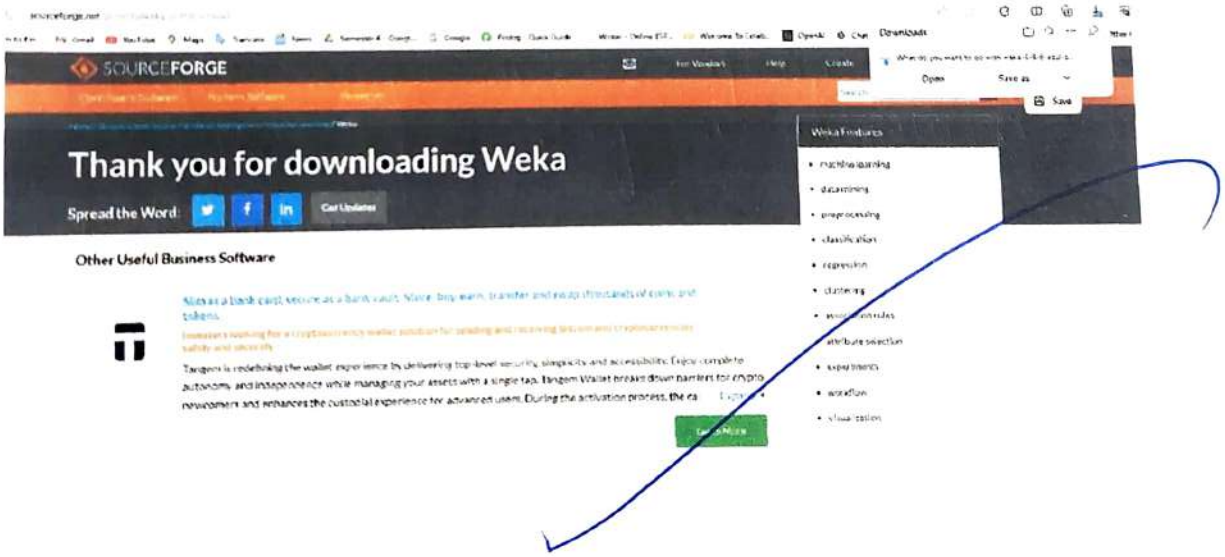
Thus, the use of WEKA results in a quick development of machine learning algorithms on the whole.

⇒ WEKA offers a graphical user interface (GUI) that allows users to interact with the software without needing to write the code, making it accessible to the non-programmers.

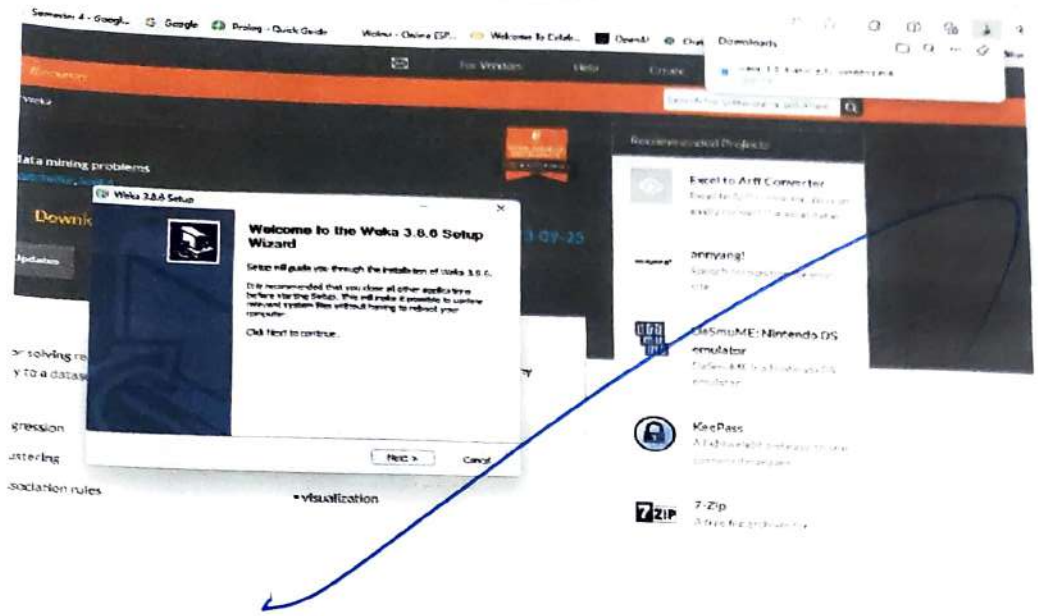
1. Visit the website, to install the latest version of WEKA.



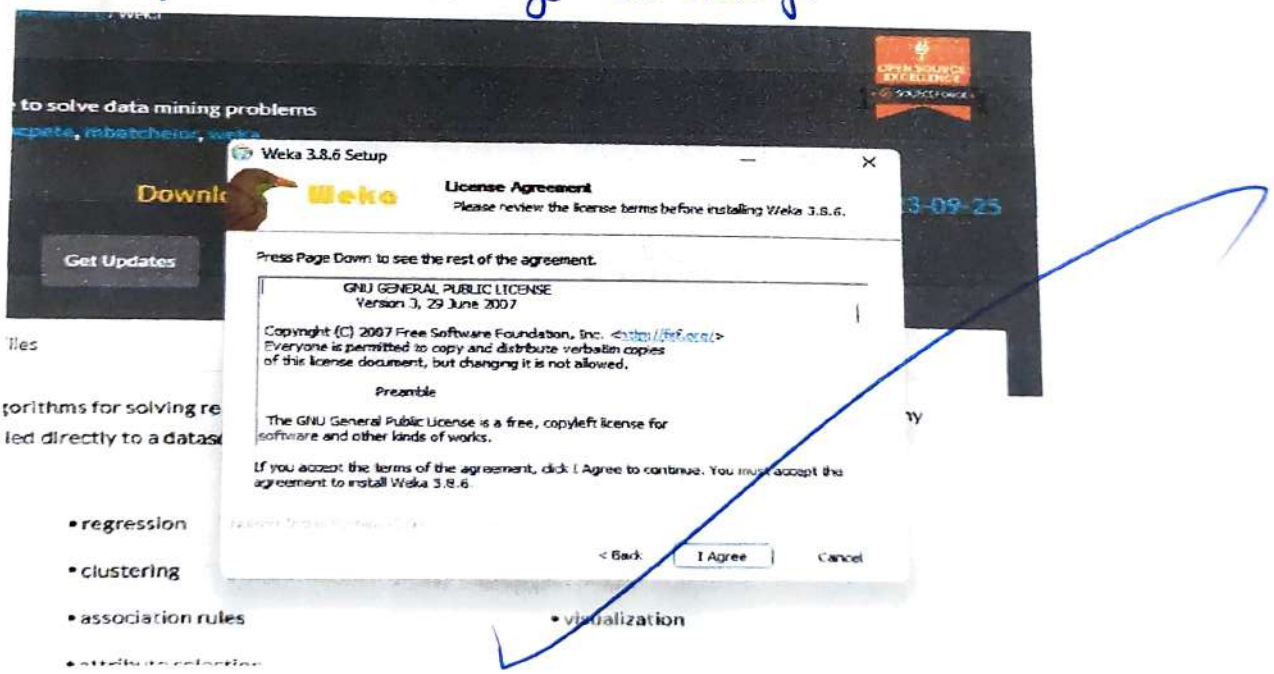
2. Initiate the WEKA setup installation wizard.



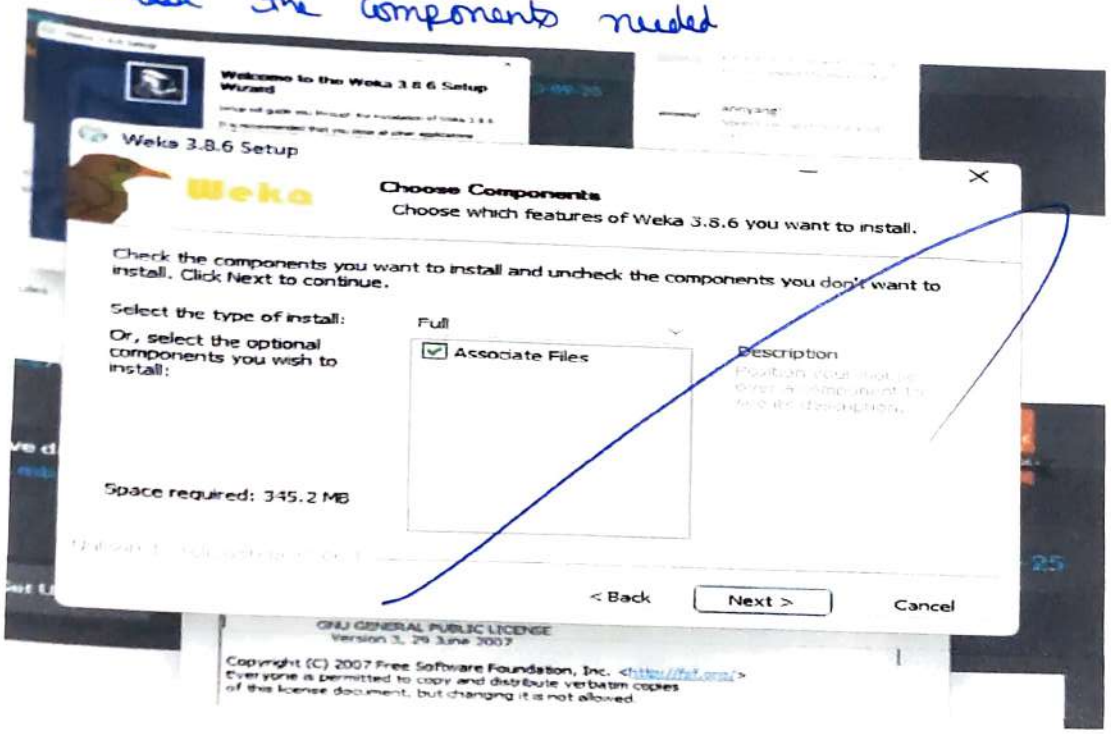
3. Accept the necessary terms & conditions



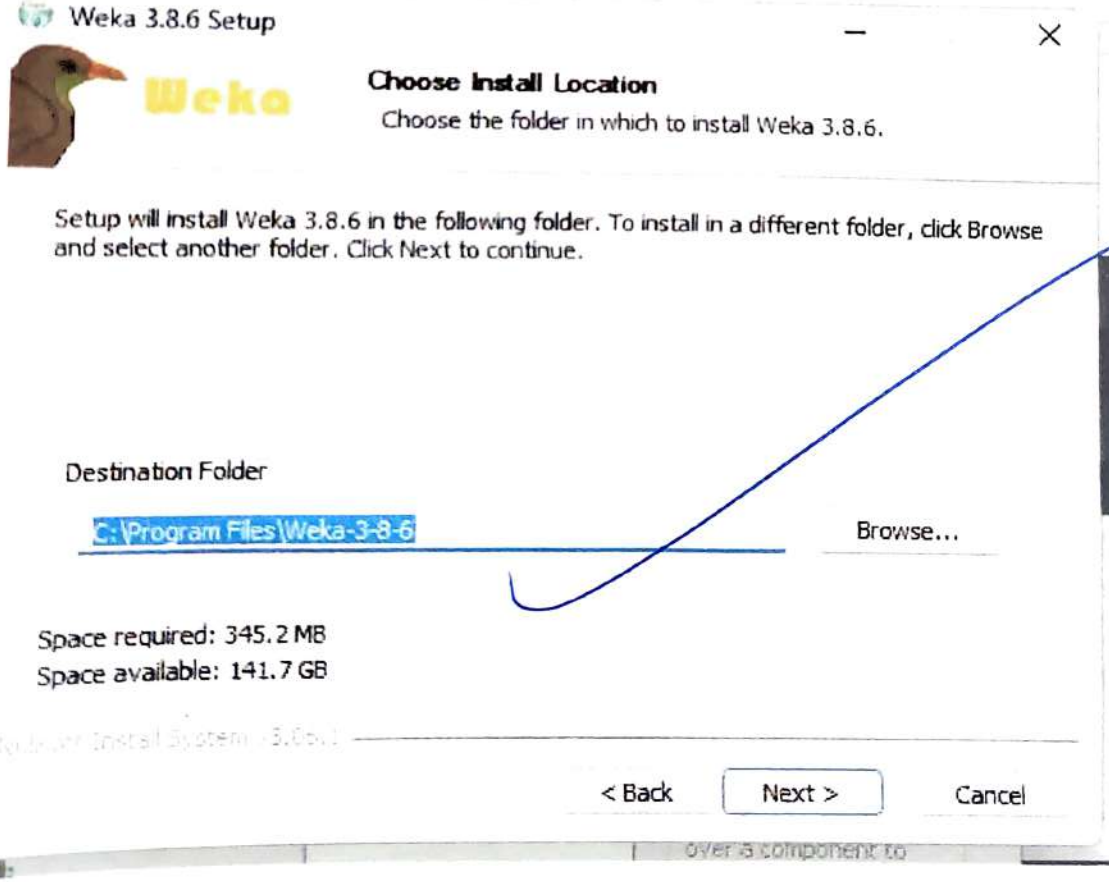
4. Setup WEKA storage directory.



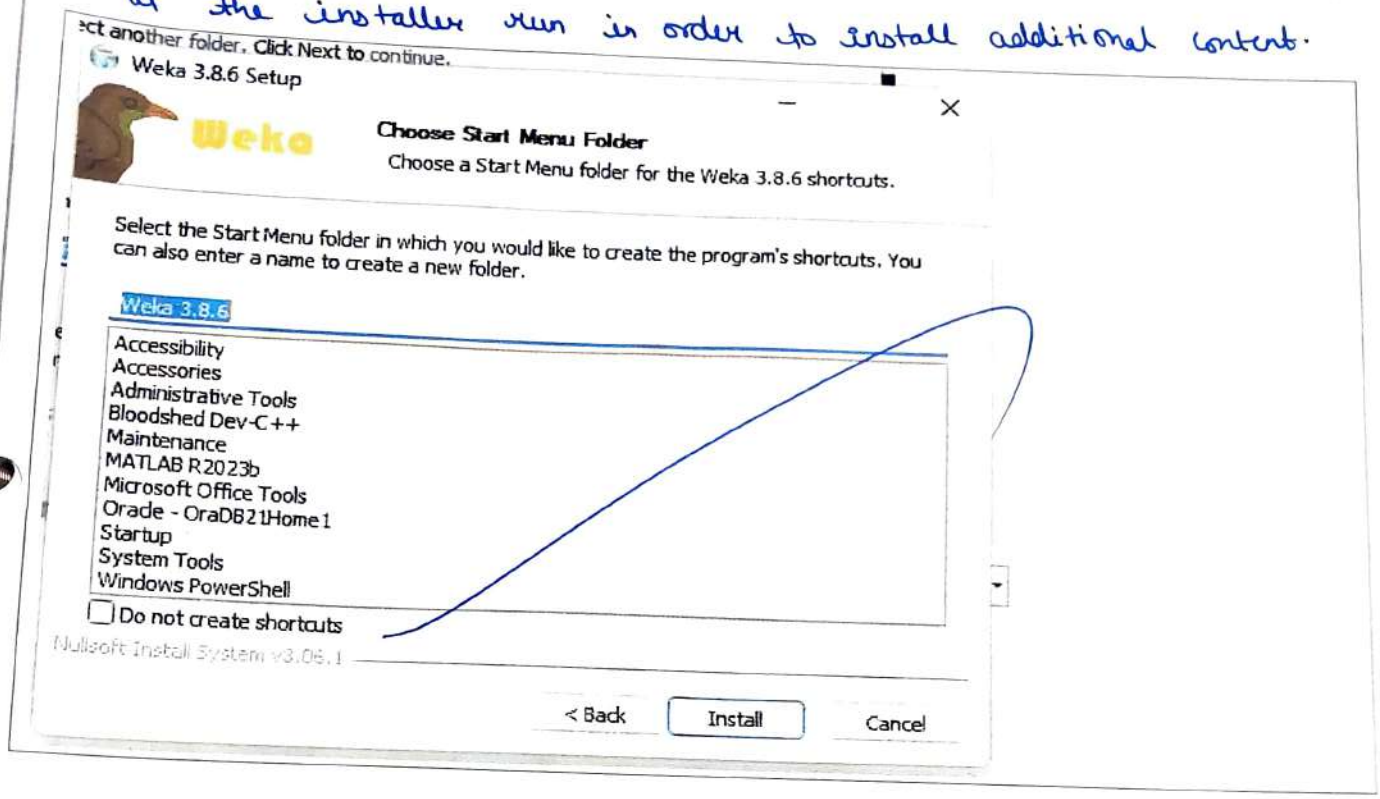
5. Choose the components needed



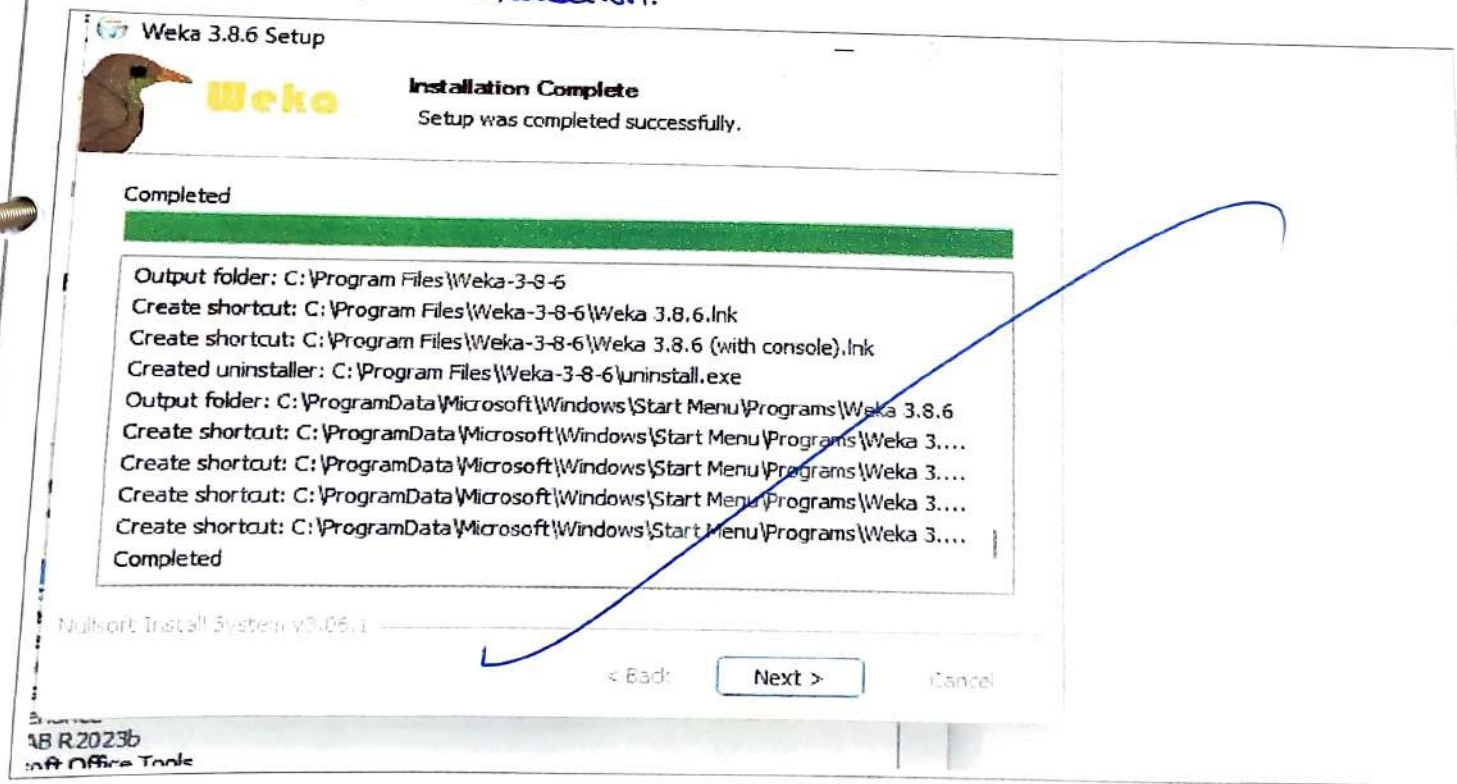
6. Choose the installation location



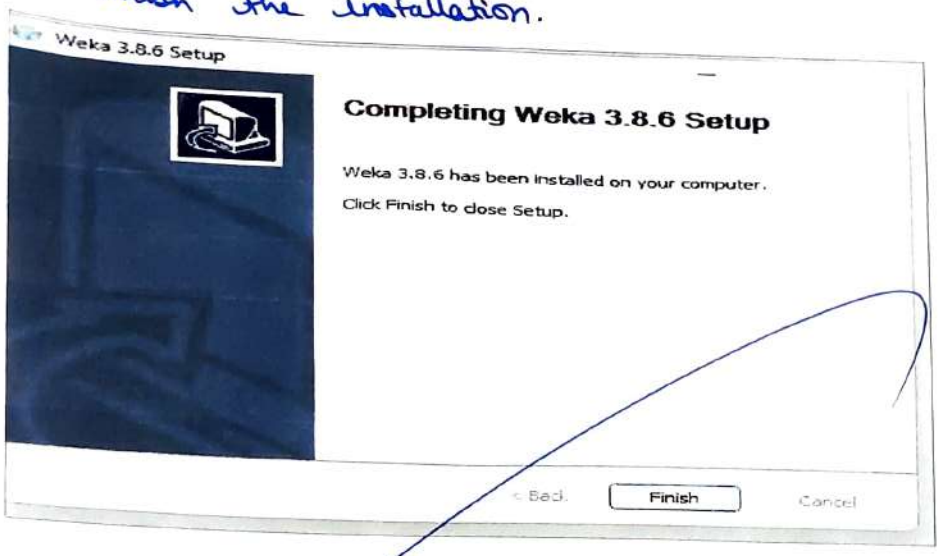
7. let the installer run in order to install additional content.



8. finish the installation.



9. finish the installation.



10. WEKA is installed.

Learning Outcomes:

→ successfully installed & understood the installation procedure of WEKA.

→ learnt about the functions of WEKA on machine learning.

9/2/24

Experiment 2

Aim: Perform data pre-processing including cleaning, integration and transformation on ARFF files using WEKA.

Theory:

Data Preprocessing: It is the process of preparing raw data for analysis by handling missing values, outliers & noise to ensure high-quality input for models. It also helps improve accuracy and reliability of data set.

Data Cleaning: It is the process of correcting or deleting inaccurate damaged, improperly formatted, duplicated or insufficient data from a dataset. Data cleaning lowers errors & raised caliber of the data. It is crucial to construct of a model

Data Integration: It is the process of merging data from several disparate sources. Data integration is a record preprocessing method that includes merging data from the couple of heterogeneous data sources into coherent data to retain and provide a unified perspective of data.

Data Reduction: Data reduction techniques ensure the integrity of data while reducing the data. Data reduction is a process that reduces the volume of original data & represents it in a much smaller volume. Data reduction are used to obtain a reduced representation of dataset.

Data Transformation: Converts data into a suitable format or structure for analysis including normalisation, scaling and feature engineering. This process ensures the data is compatible with machine learning models.

.xlxs File

	Name	Marks
2	Ram	89
3	shyam	90
4	lakhan	92
5	makan	67
6	lakhshman	45
7	shivam	43
8	rohan	67

.arff File

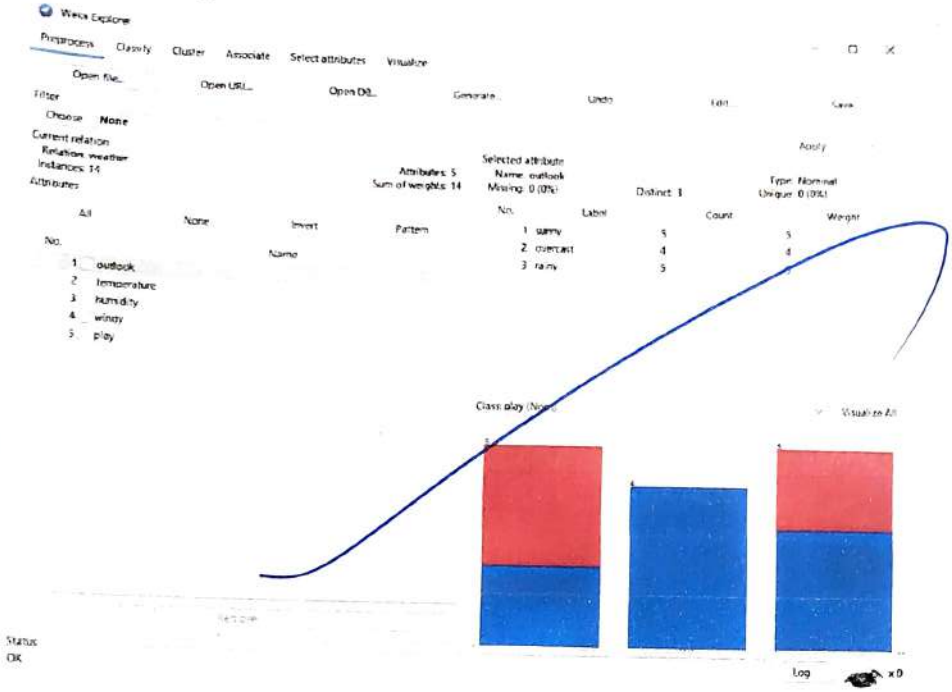
.text File

```
File    Edit    View

@relation weka2

@attribute Name string
@attribute Marks numeric

@data
Sita,67
Ram,78
```

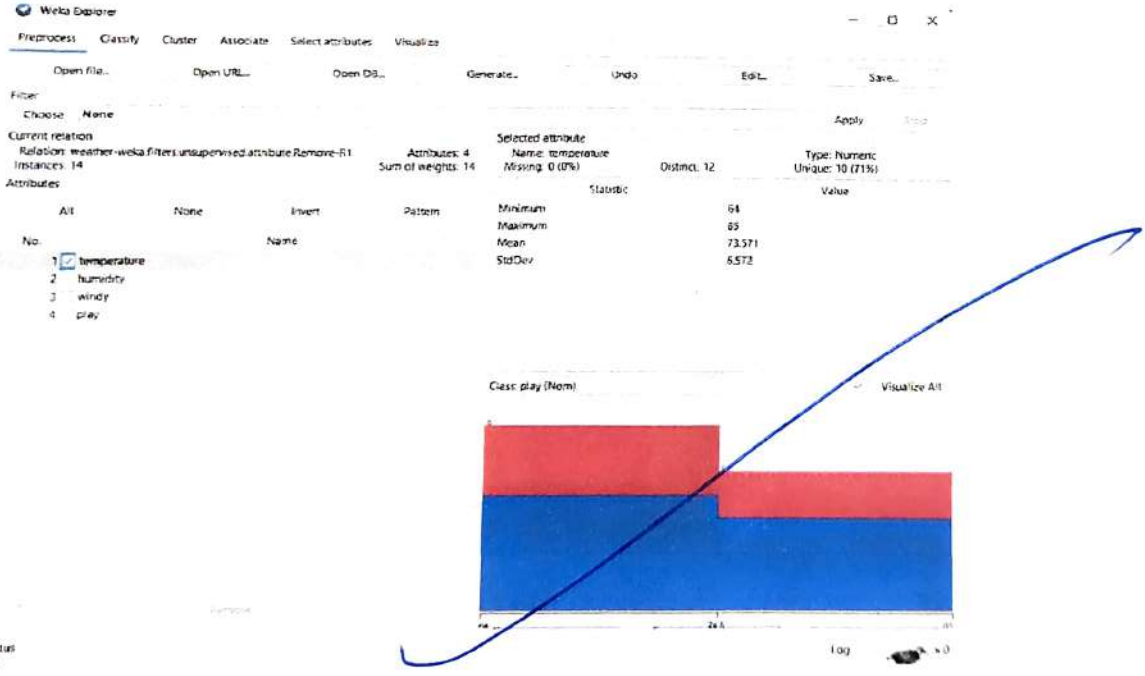
.arff File

Data Cleaning

Before Cleaning

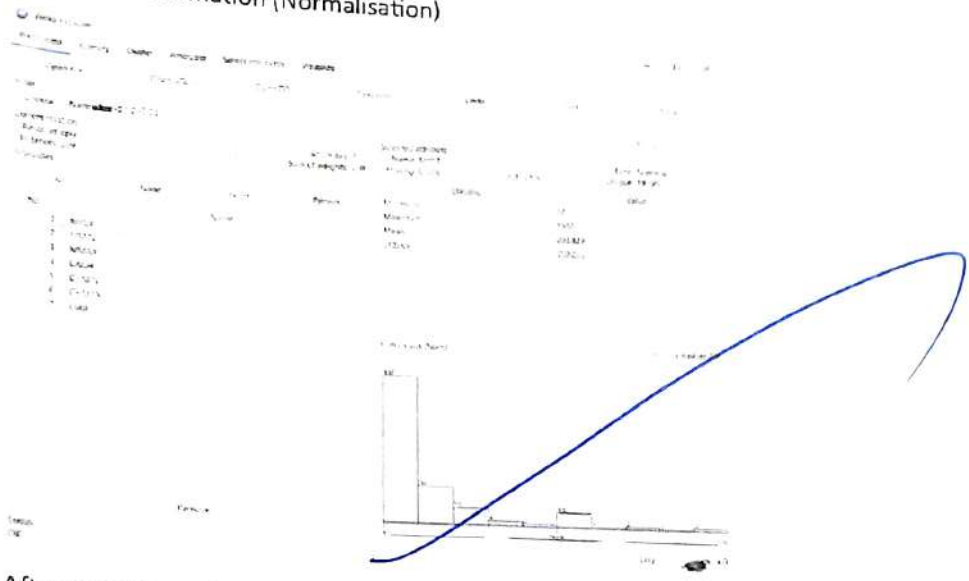


After Cleaning

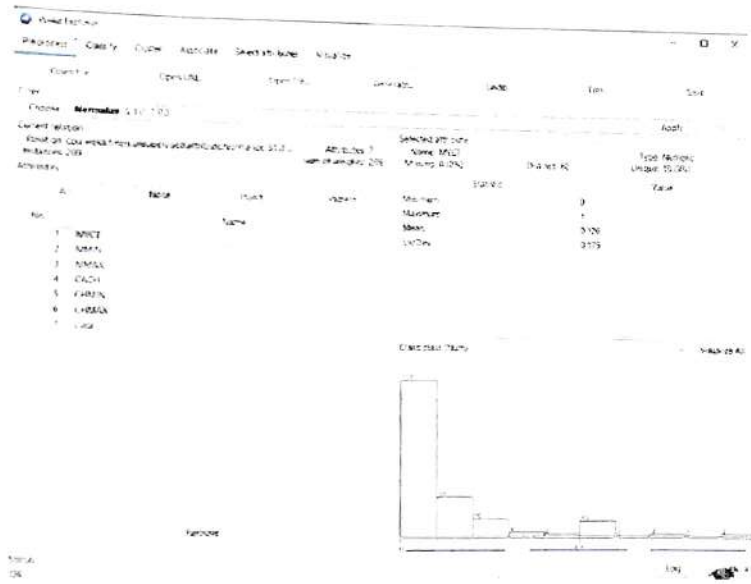


Data Transformation

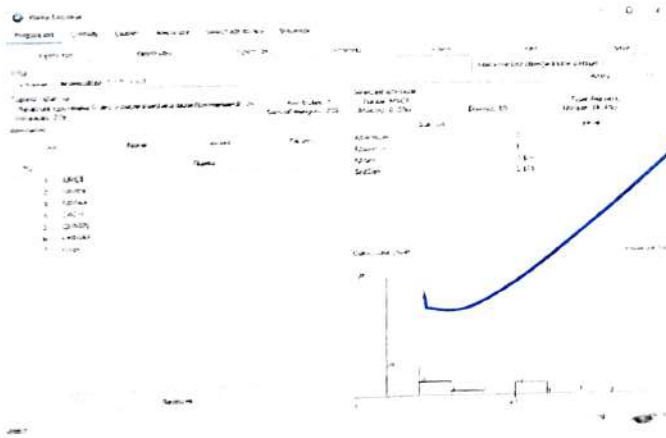
Before Transformation (Normalisation)



After Transformation

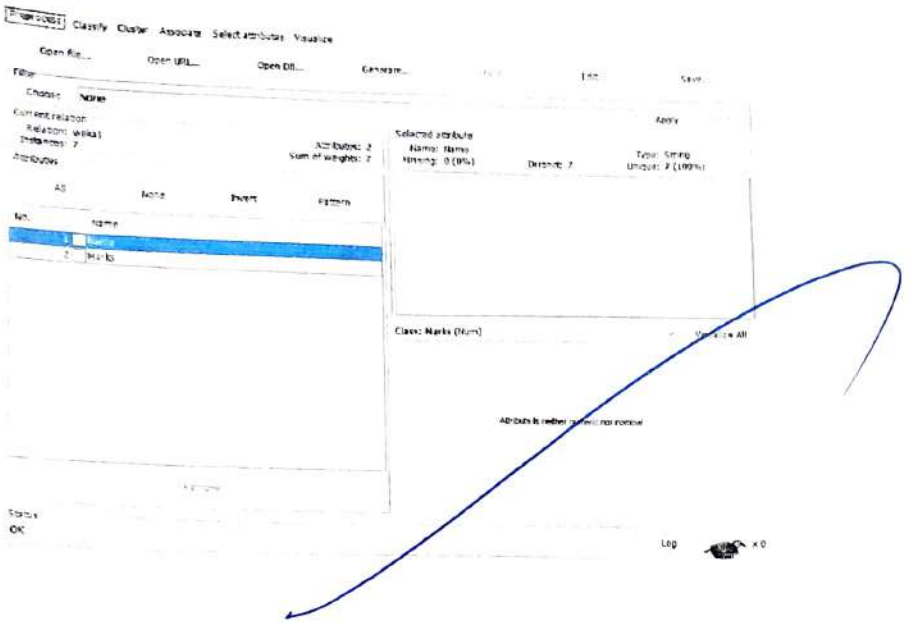


Undo

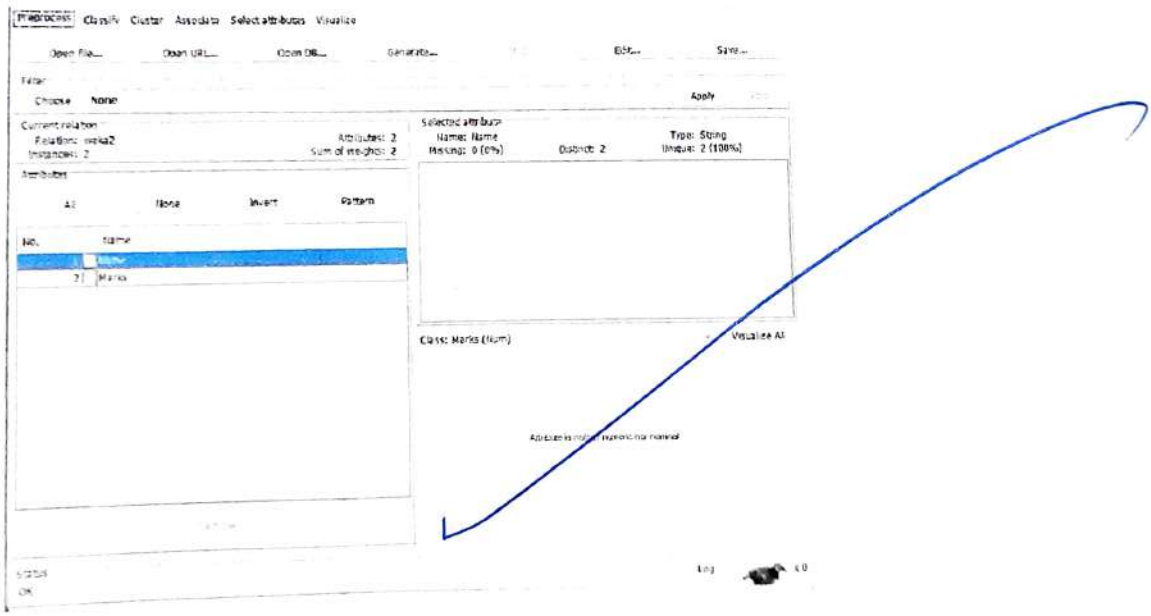


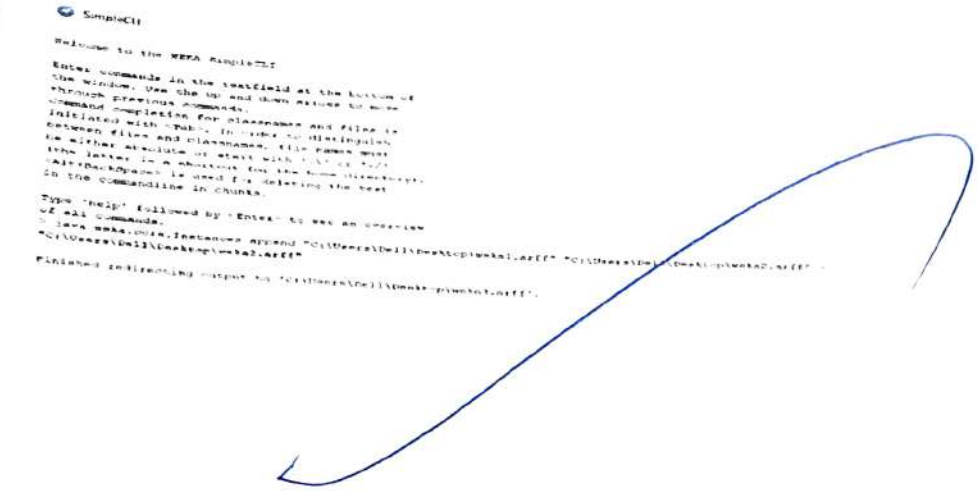
Data Integration

File 1 (Weka1.arff)



File 2 (Weka2.arff





Learning Outcome:

→ learnt about applying various machine learning techniques on weka such as data cleaning, data reduction, integration, etc.

→ learnt converting .xlxs file onto .arff file. + .txt file to .arff file

Harshita
9/8/24

Experiment 3

Aim: Apply Association Rule Mining on a dataset using Weka.

Theory: Association Rule, is a data mining technique used to identify relationships between variables in large datasets.

→ It uncovers patterns in the form of 'If-then' statements, showing how the occurrence of one item is associated with another.

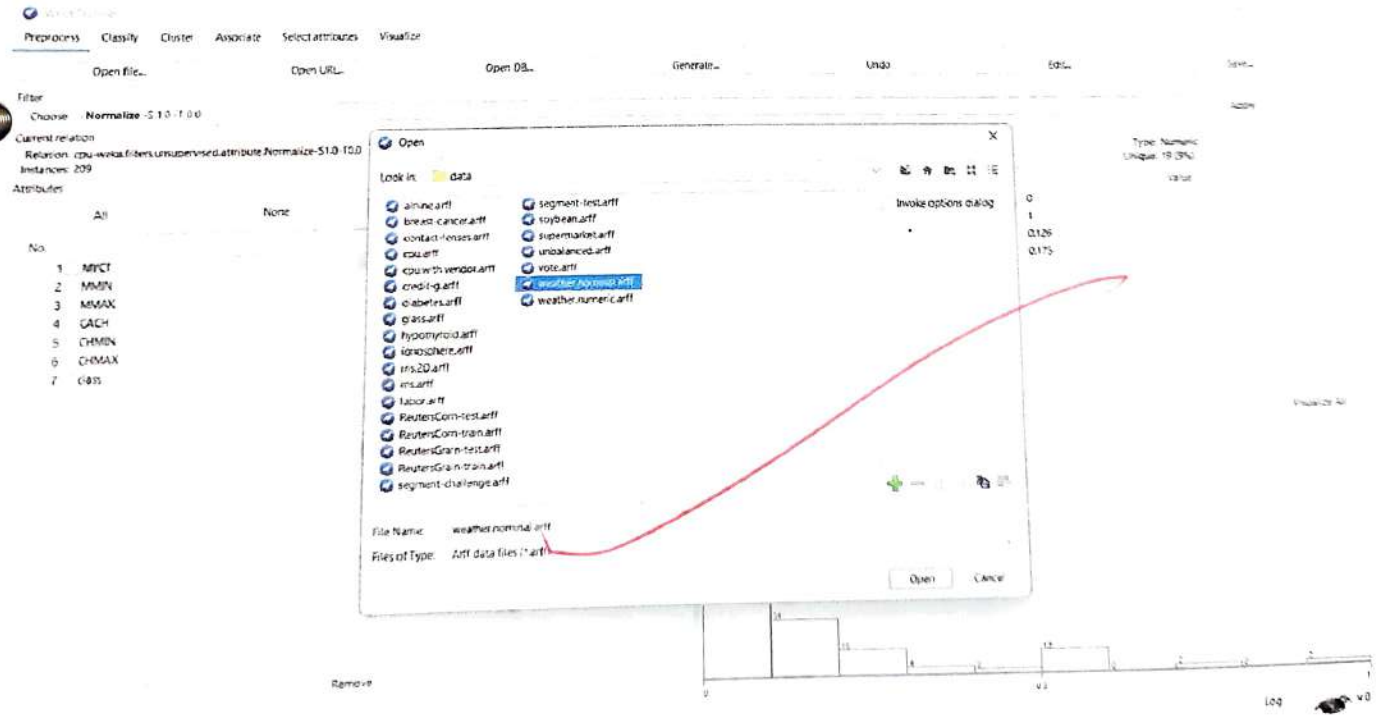
→ These rules are commonly used in market basket analysis to discover product associations, like "if a customer buys bread, they are likely to buy butter".

→ Support: measures how frequently an itemset appears in dataset.

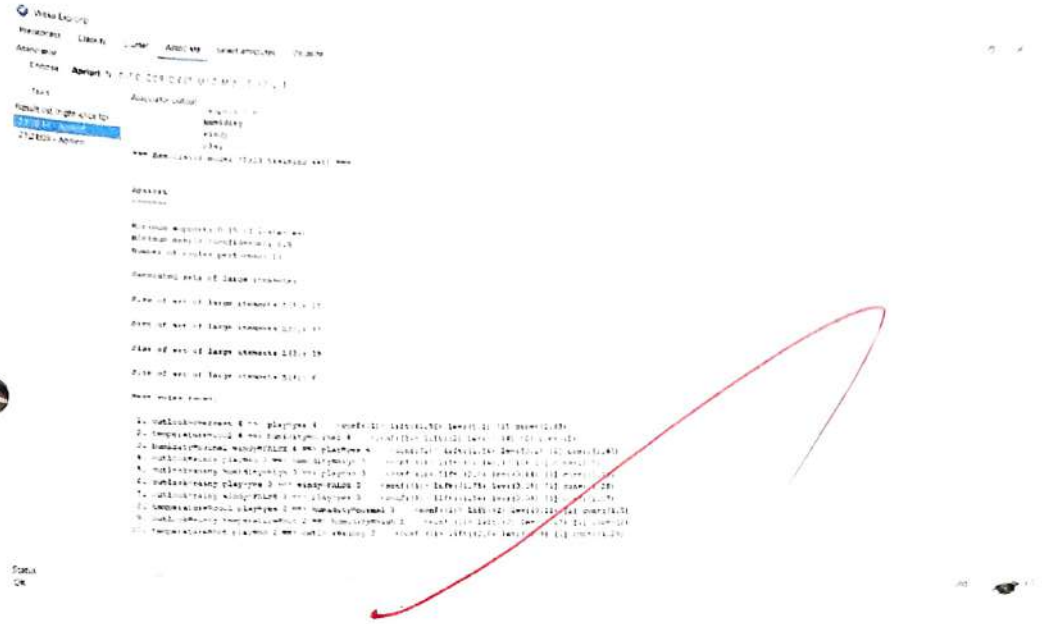
→ Confidence: indicates the likelihood that an item B will be bought if item A is bought.

→ Lift :- shows how much more likely item B is to be bought with item A.

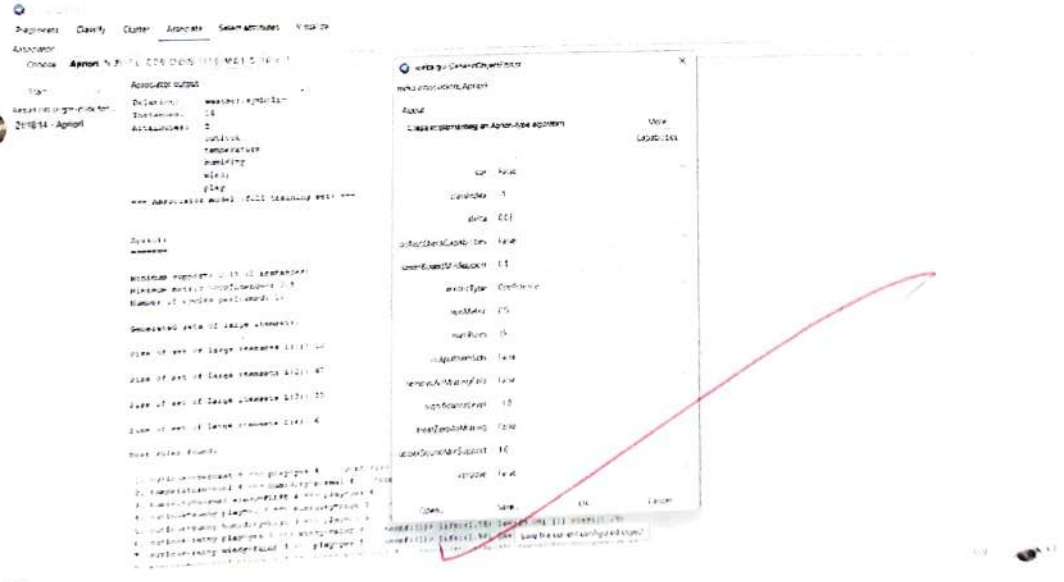
Opening Data:



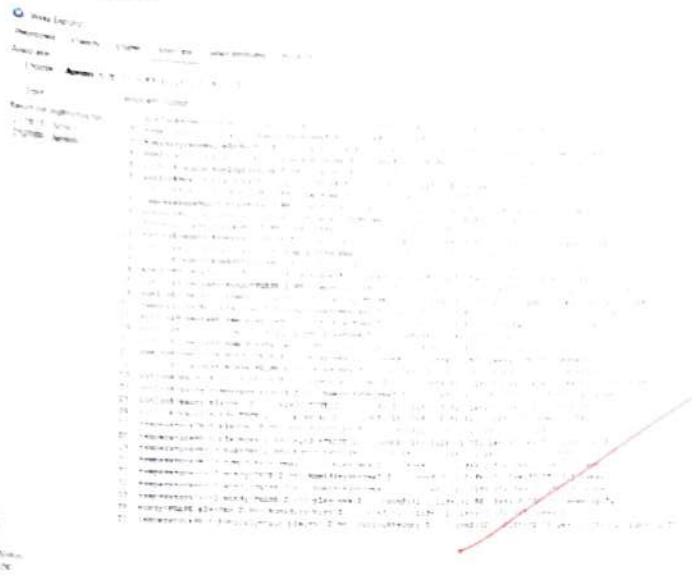
Applying Apriori Algorithm



Changing num Rules



New Rules



Learning Outcome:

- Understood the concept of Association Rule.
- Learnt how to apply various association Rules on WEKA.
- Learnt how to set parameters of an algorithm on WEKA.

Checked 20/09/22

Experiment 4

Aim: Apply Visualization on a dataset using Weka.

Theory: Data visualization is an important feature for ~~exploration~~ explaining and understanding datasets before applying ML algorithms.

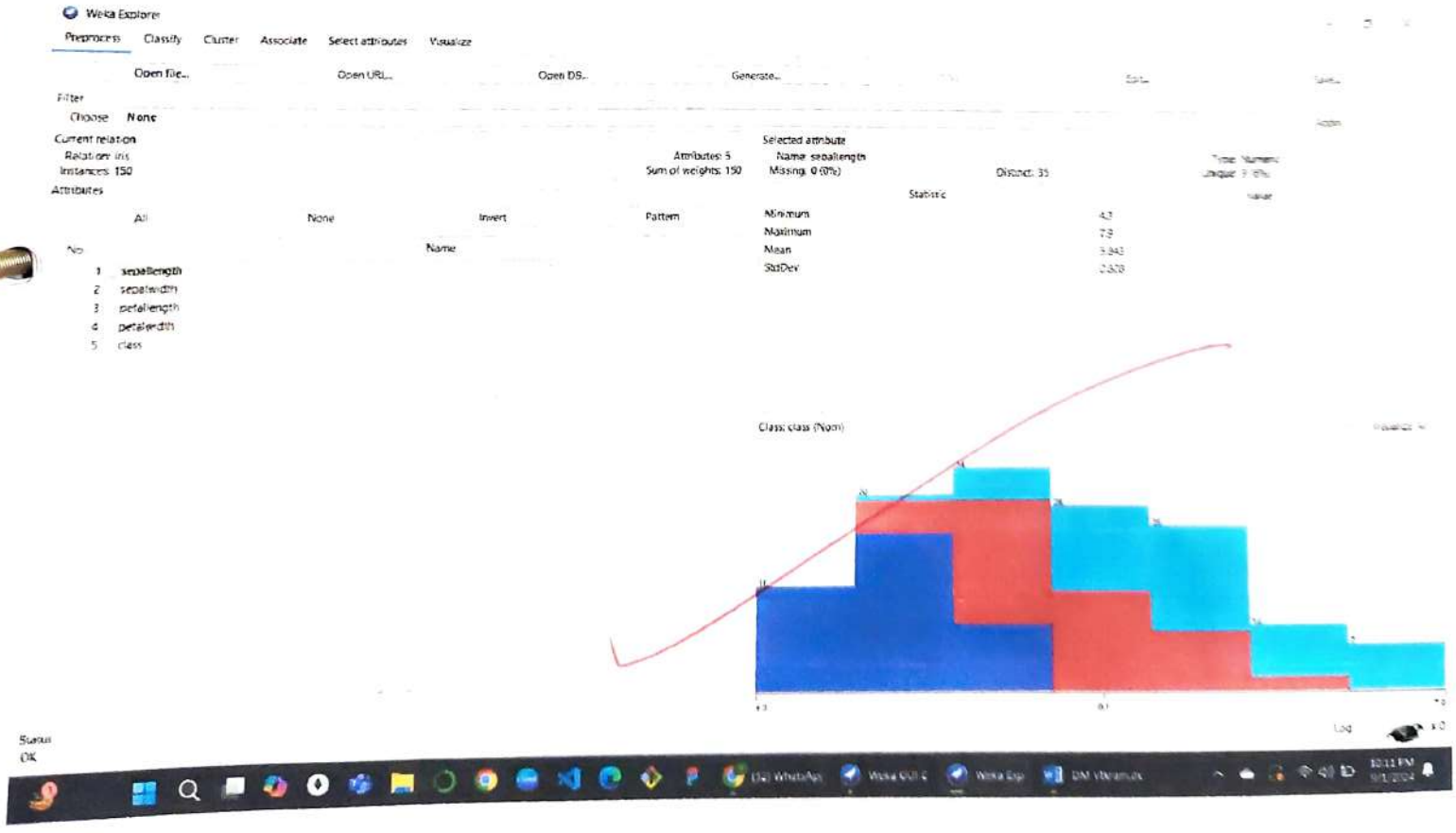
- WEKA provides several tools and methods for visualizing the data, which can help in tasks such as data preprocessing, feature selection, and model evaluation.

* Types of visualisation in WEKA:

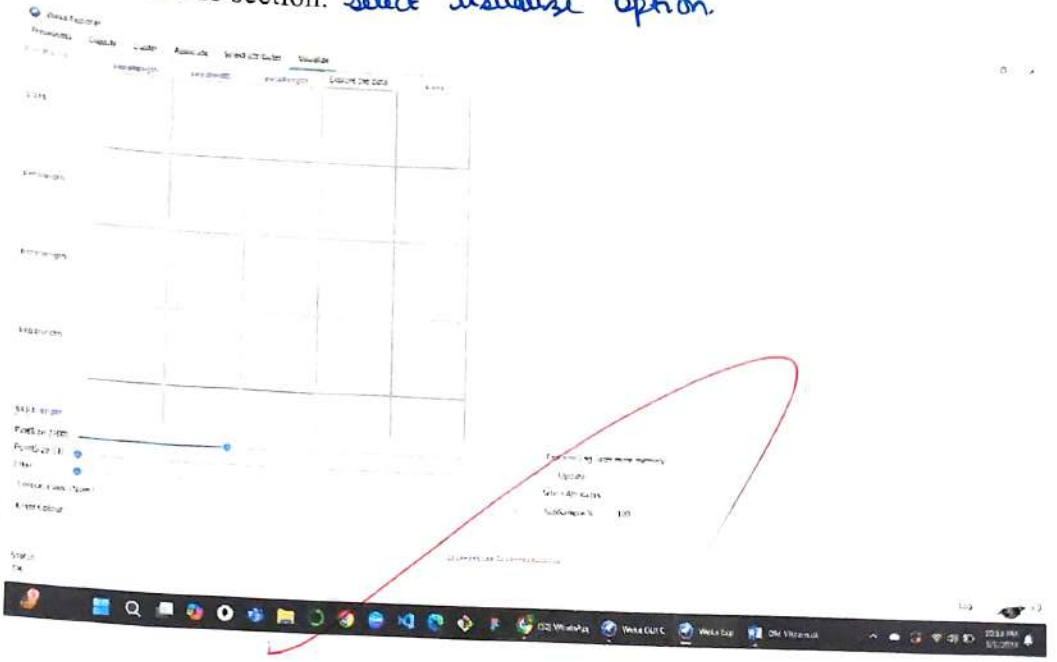
1. Data Explorer: Histogram, Box Plot, Scatter Plot.
2. Visualisation tools: Attribute selection, Class distribution.
3. Model evaluation: ROC curves, Confusion Matrix.

Open Iris Dataset:

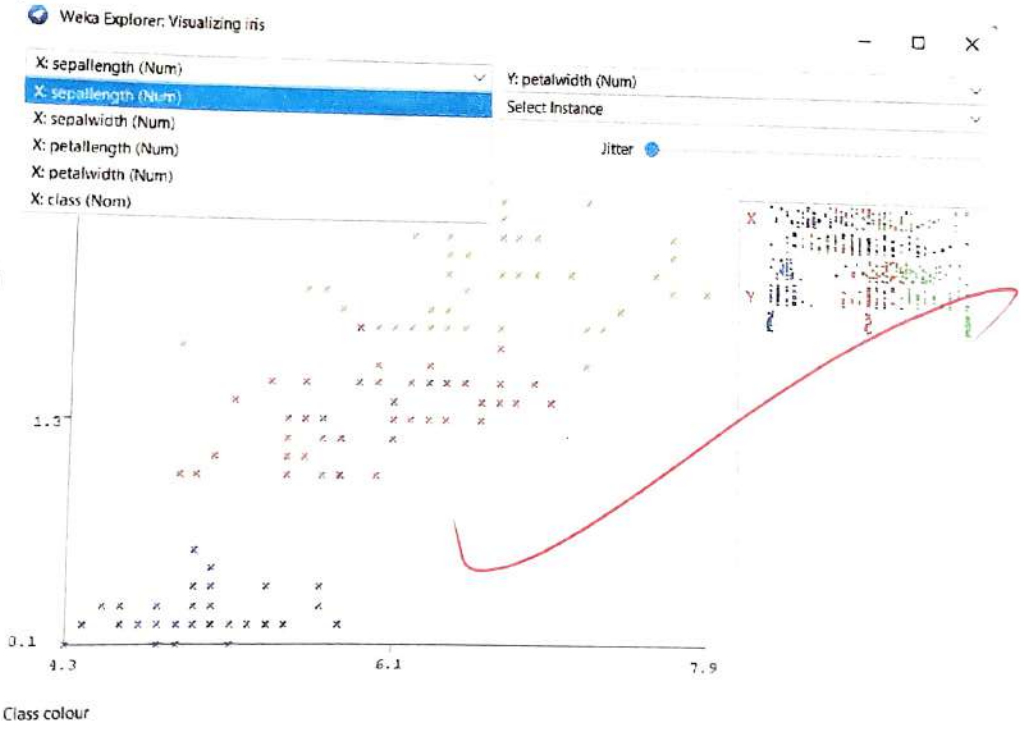
choose any numeric dataset from program files > weka > data > "any data"



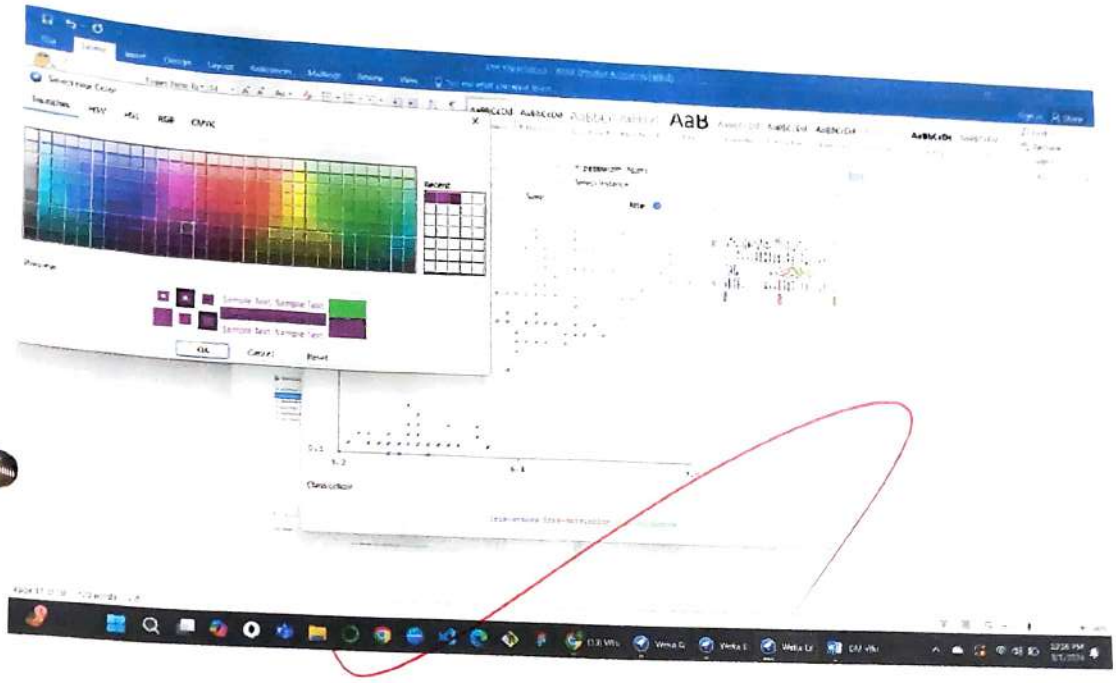
Open visualize section: select visualize option.



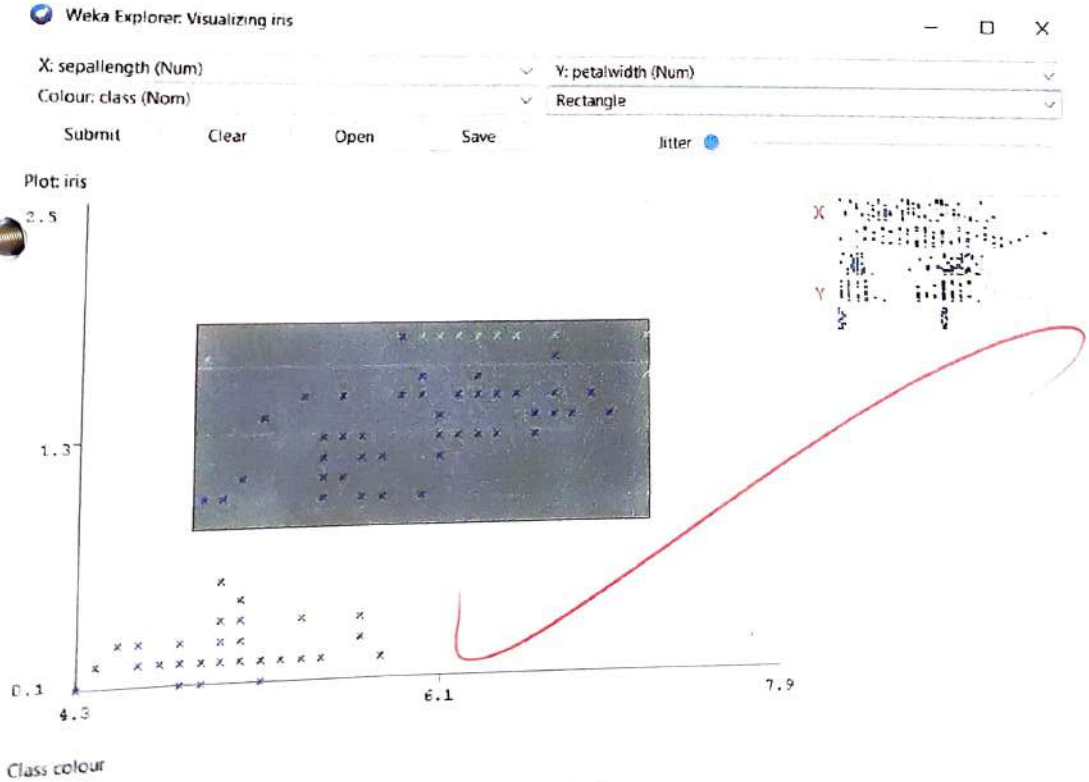
Visualizing dataset in different scenarios: Dataset visualization . x: sepalength & y: petal width.



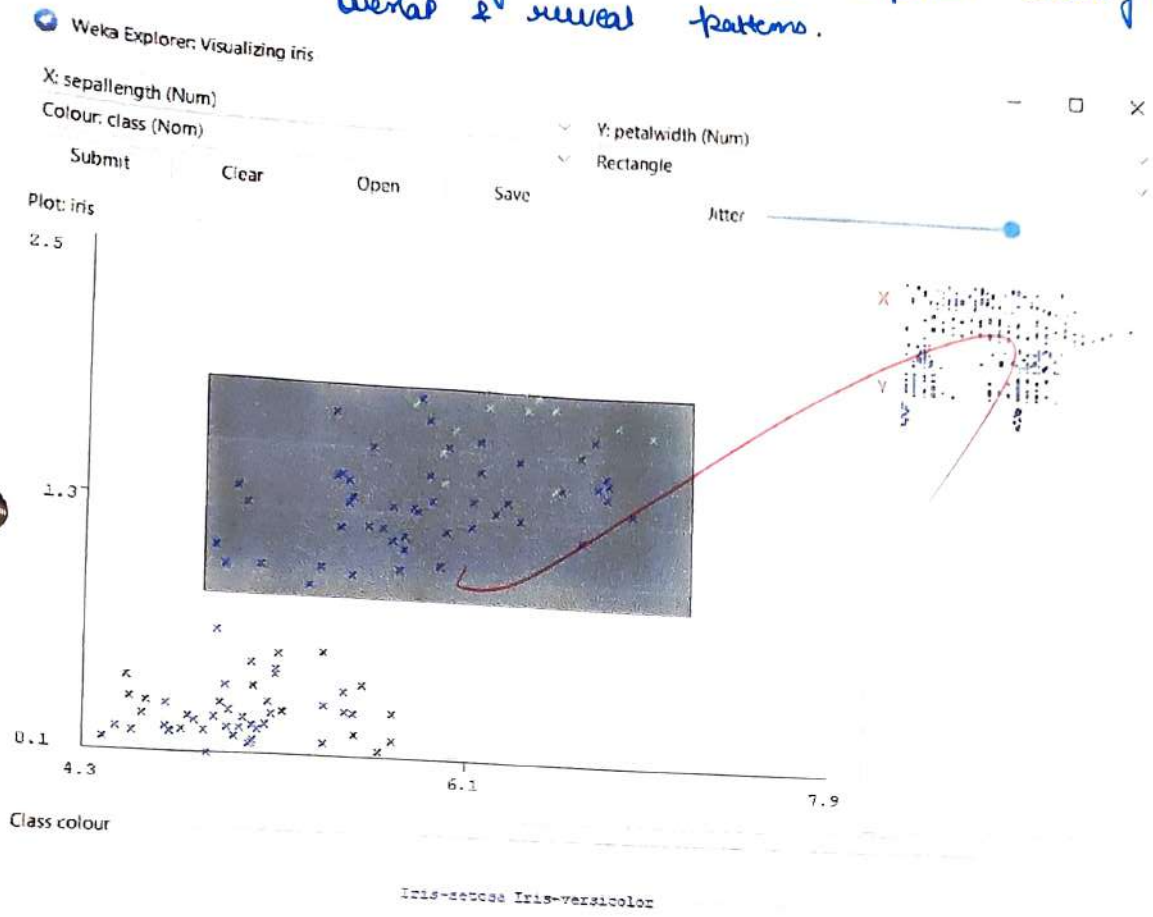
Exploration of colour tool:



Selecting Instances: select "Rectangle" Option.



Changing the jitter: increasing jitter, we can improve clarity, Reduce overlap & reveal patterns.



Learning Outcome:

1. understood the concept of Data visualization in Data Mining.
2. learnt the process of data visualization in WEKA.

Satisfied 20/09/24

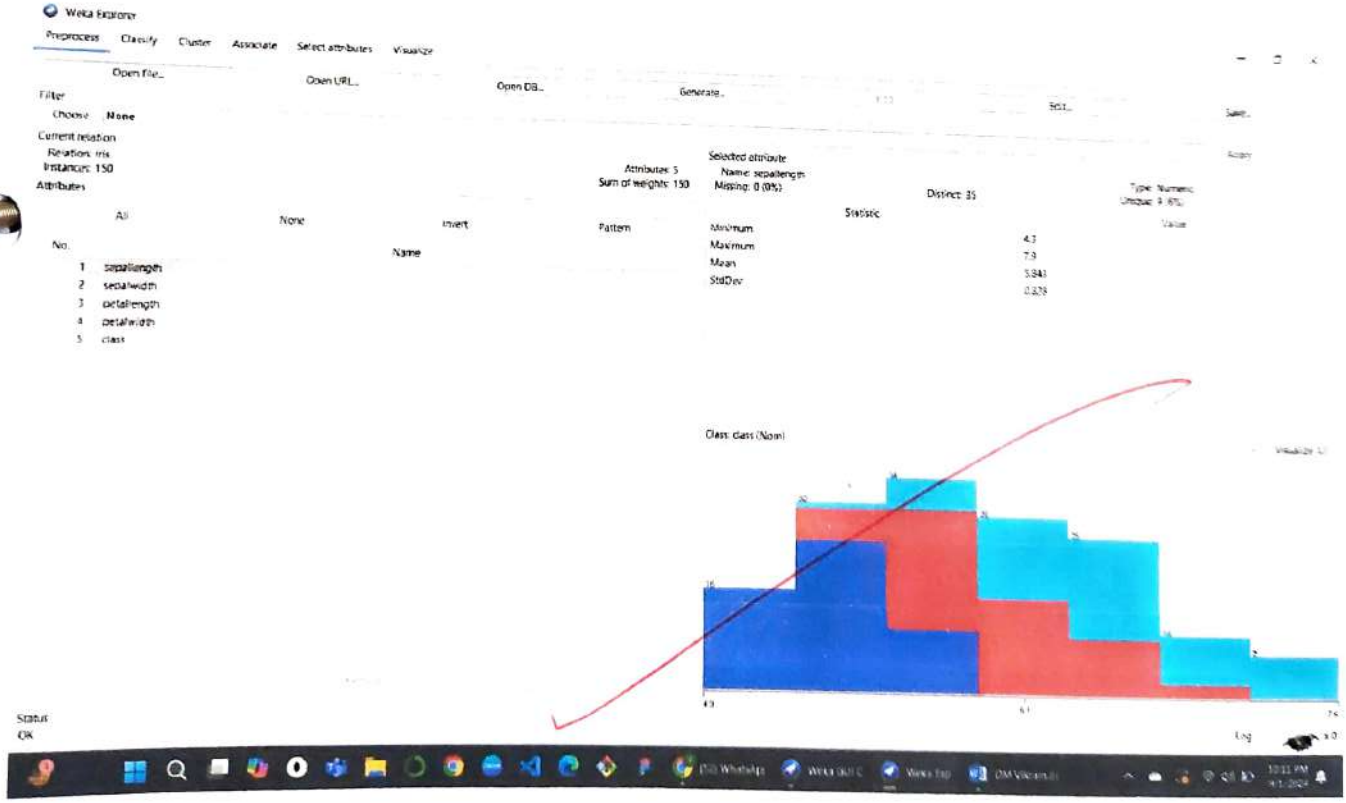
Experiment 5

Aim: Apply Clustering on a dataset using Weka.

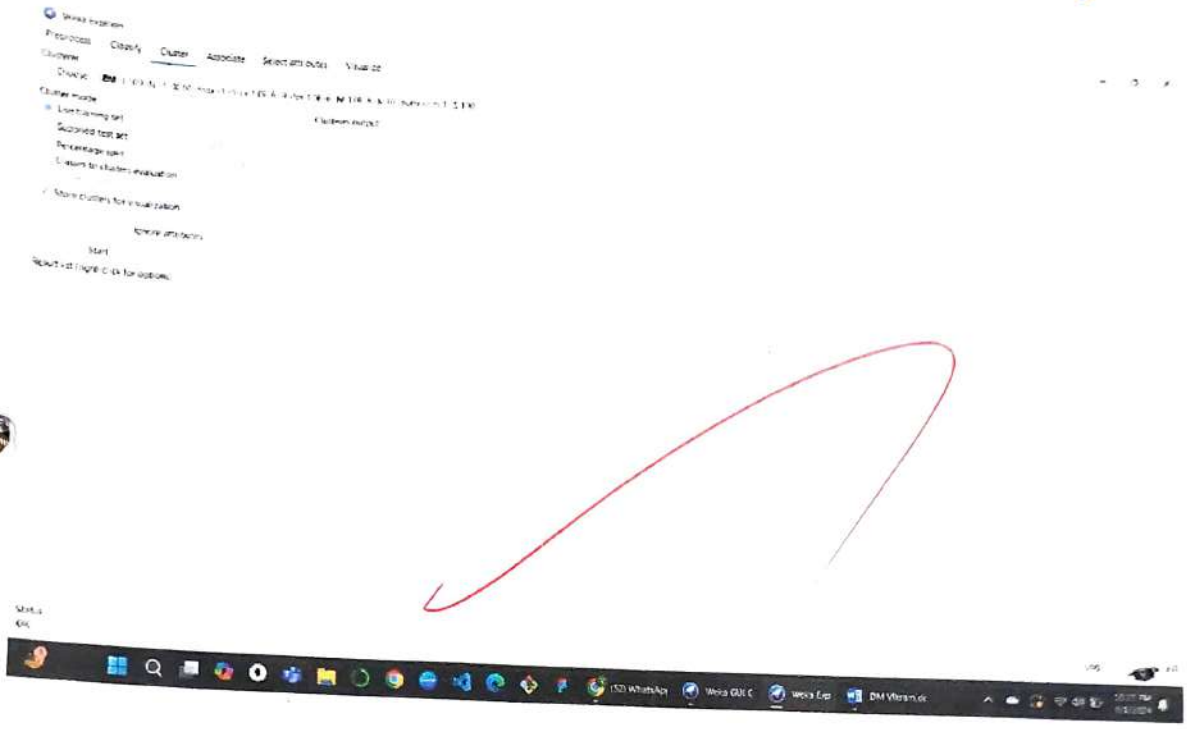
Theory: Clustering is a technique in data analysis and machine learning that groups a set of objects into clusters based on their similarities. Objects within the same cluster are more similar to each other than to those in other clusters.

- Algorithms for clustering :-
  - → K-means
  - → Hierarchical clustering
  - → DBSCAN
- Purpose
  - → Pattern Discovery
  - → Data Reduction.

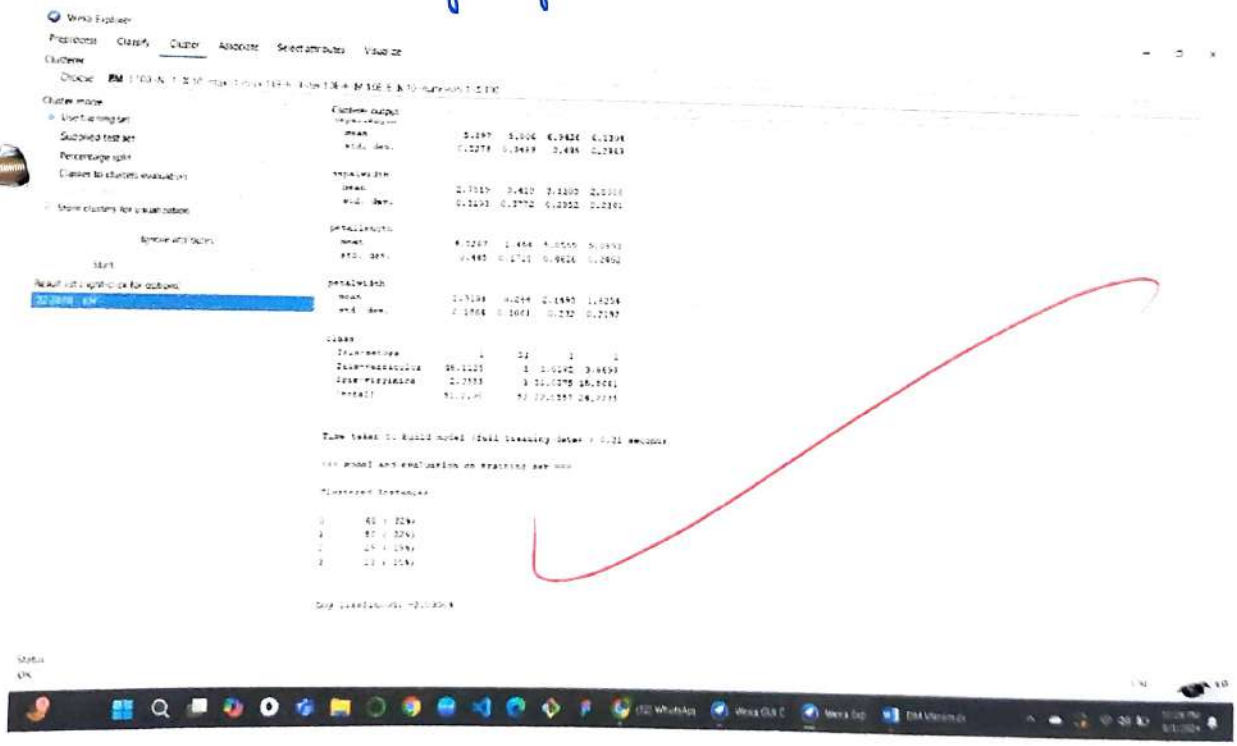
Open Iris Dataset:



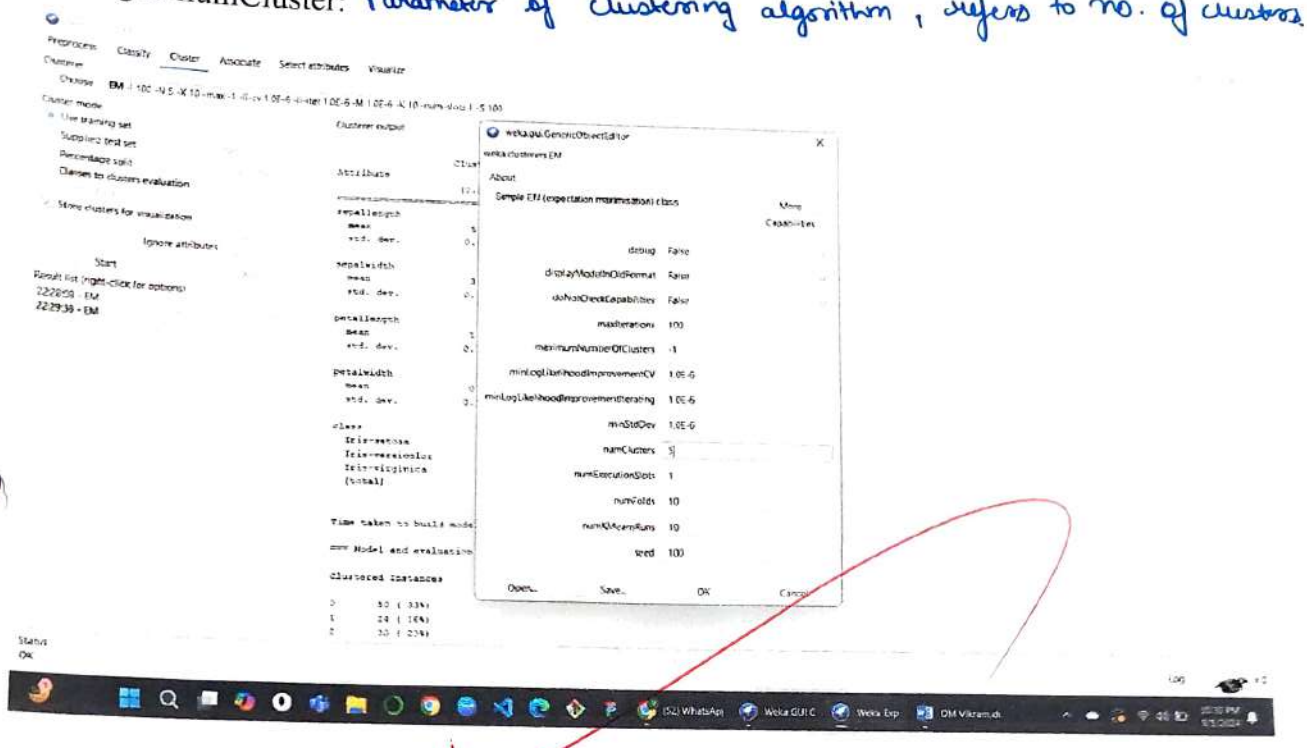
Selecting cluster option: at the top of Explorer dashboard of WEKA.



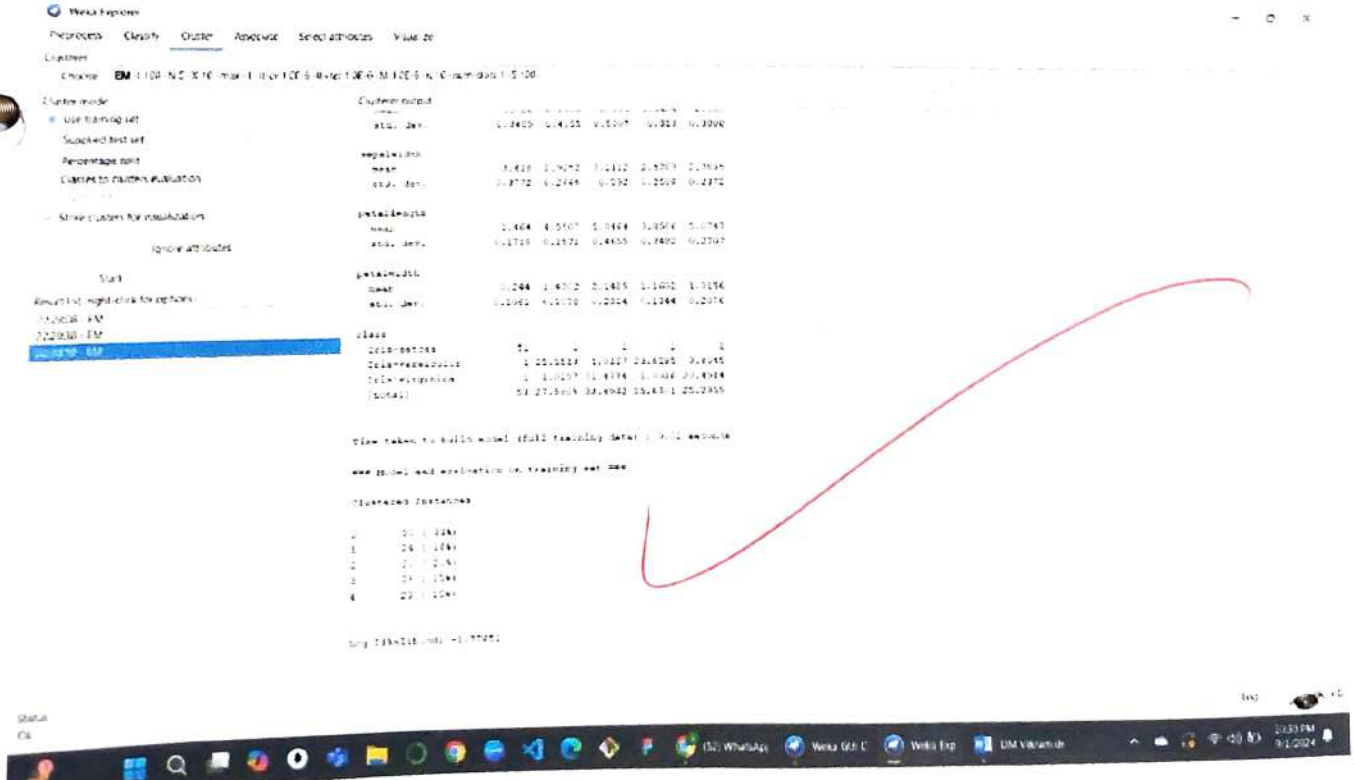
Applied clustering: From 'Choose' selected clustering option, & 'Start' button to apply algorithm to dataset.



Changed numCluster: Parameter of clustering algorithm, refers to no. of clusters

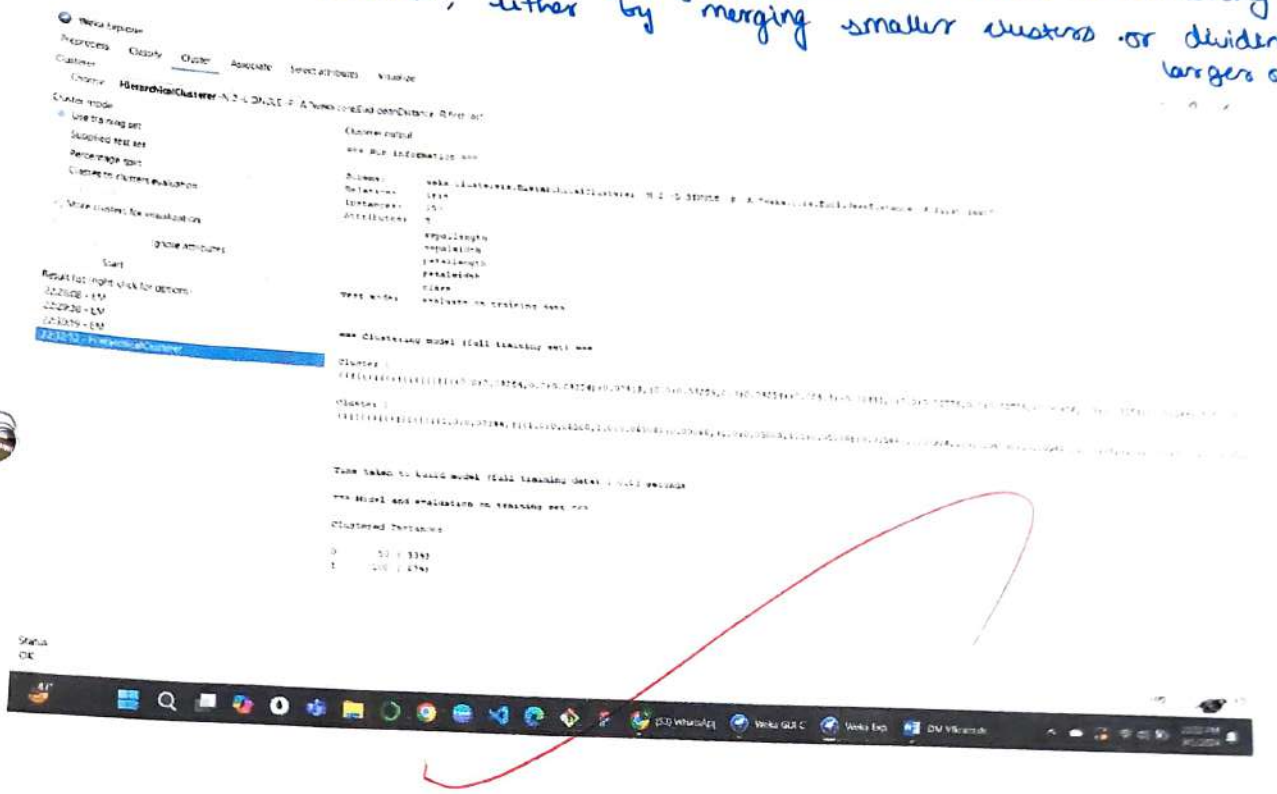


Output after changing numcluster:

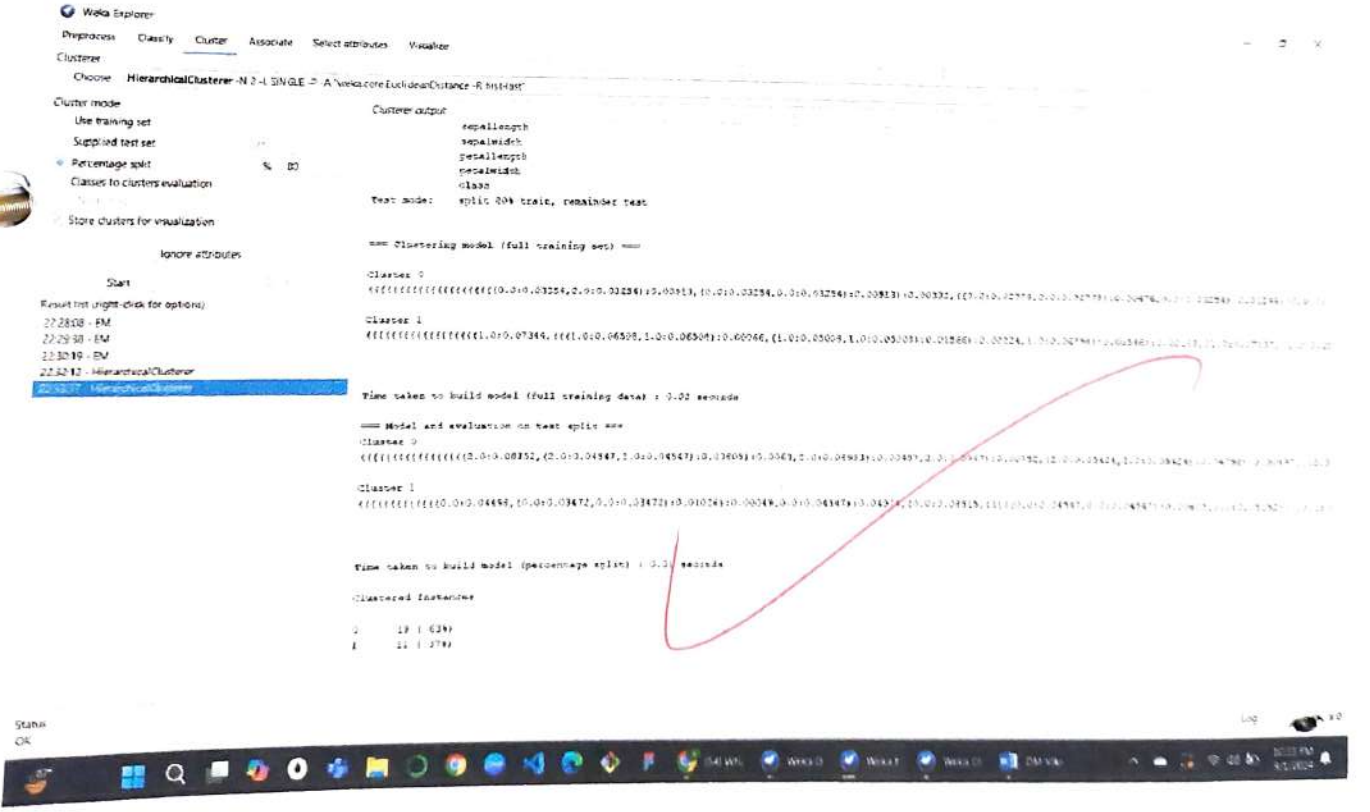


Applied Hierarchial Cluster:

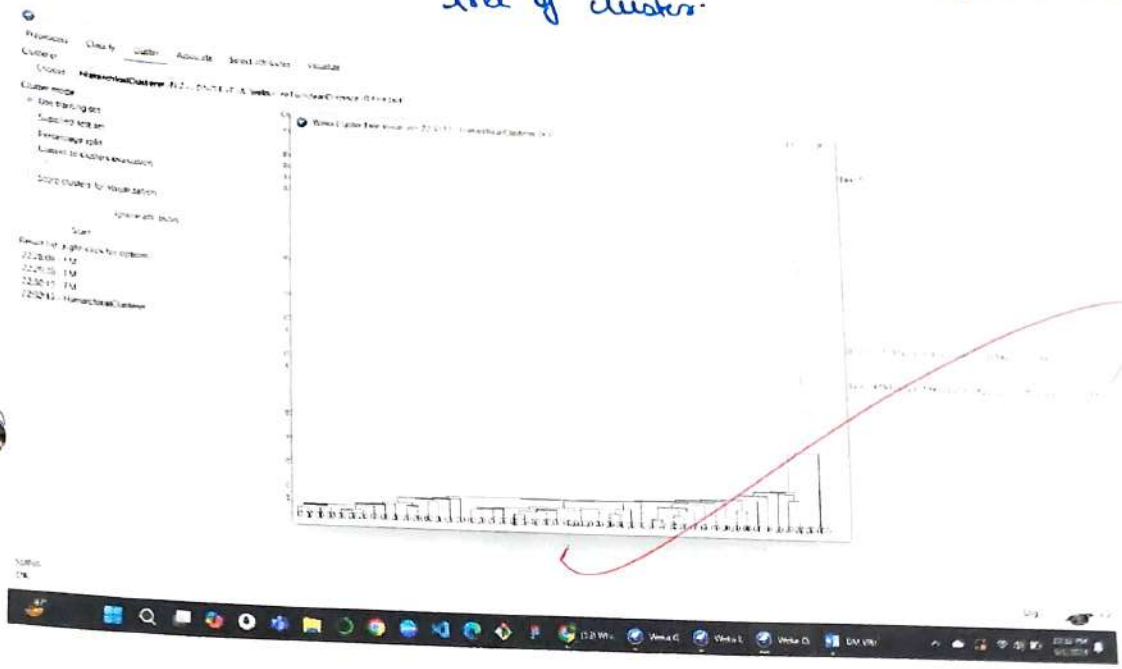
Clustering algorithm which creates a hierarchy of clusters, either by merging smaller clusters or dividing larger ones



Output of Hierarchial Cluster:



Visualization of Hierarchial Cluster Tree: choose tree visualize, to visualize tree of clusters.



Learning Outcome:

→ understood the clustering concept in data mining.

→ learnt applying different clustering options on dataset in Weka.

→ Visualize the clustering tree.

Successful 23/09/24

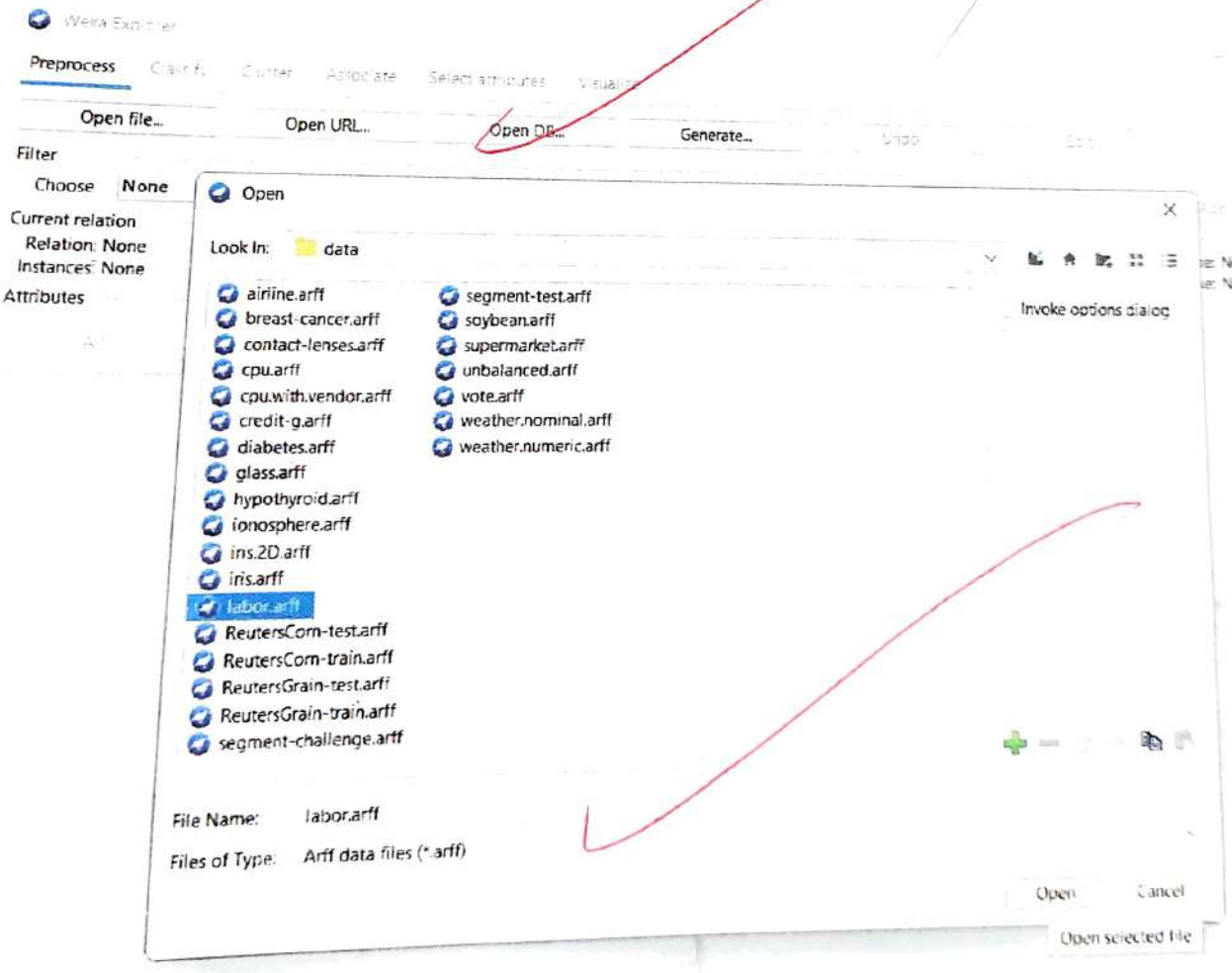
Experiment 6

Aim: Apply Classification on a dataset using Weka.

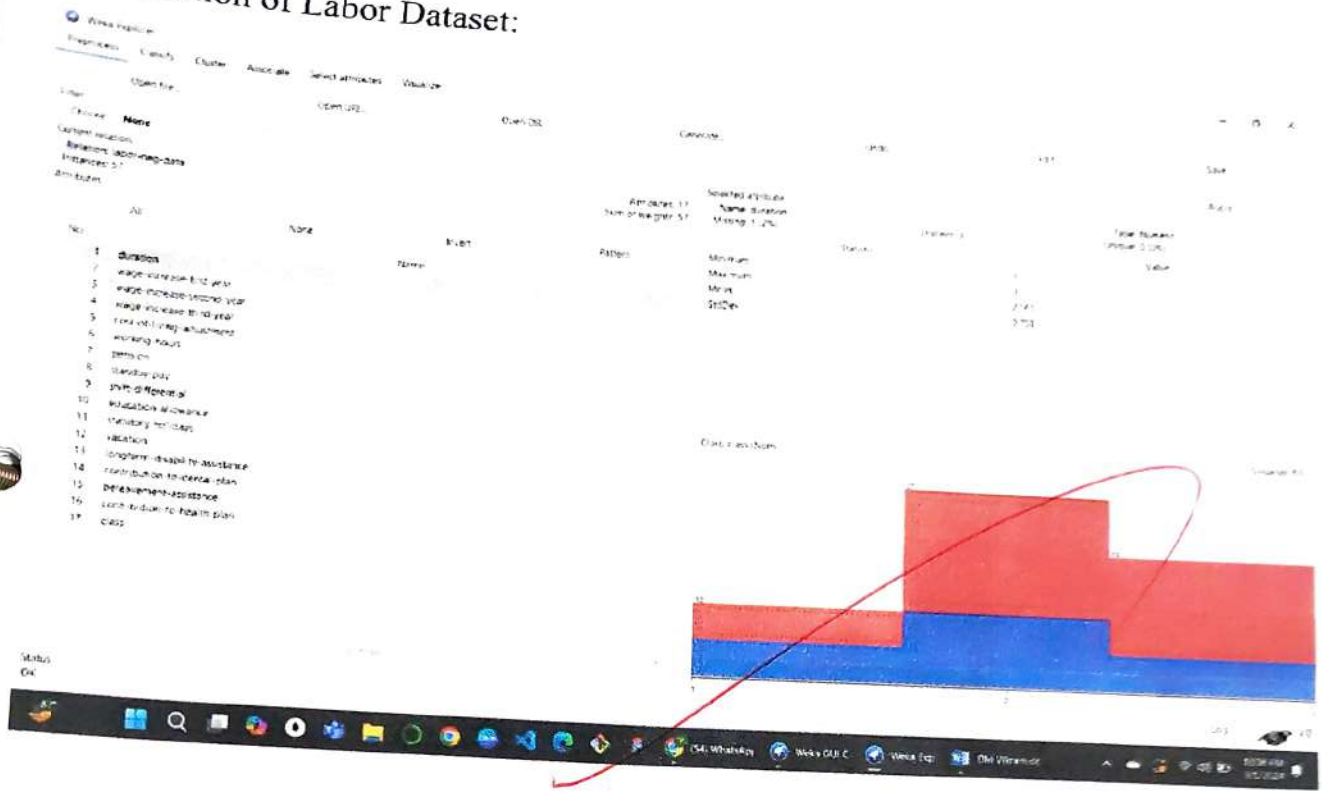
Theory: Classification is a supervised ML technique used to predict the category or class of a given data point based on its features. It involves training model on labled data to learn patterns and relationships, which can then be used to classify new, unseen data points.

- Common algorithms are:-
  - → Decision Tree
  - → Naive Bayes
  - → Support Vector Machine
  - → K-Nearest Neighbour
  - → logistic regression

Selection of Labor dataset:

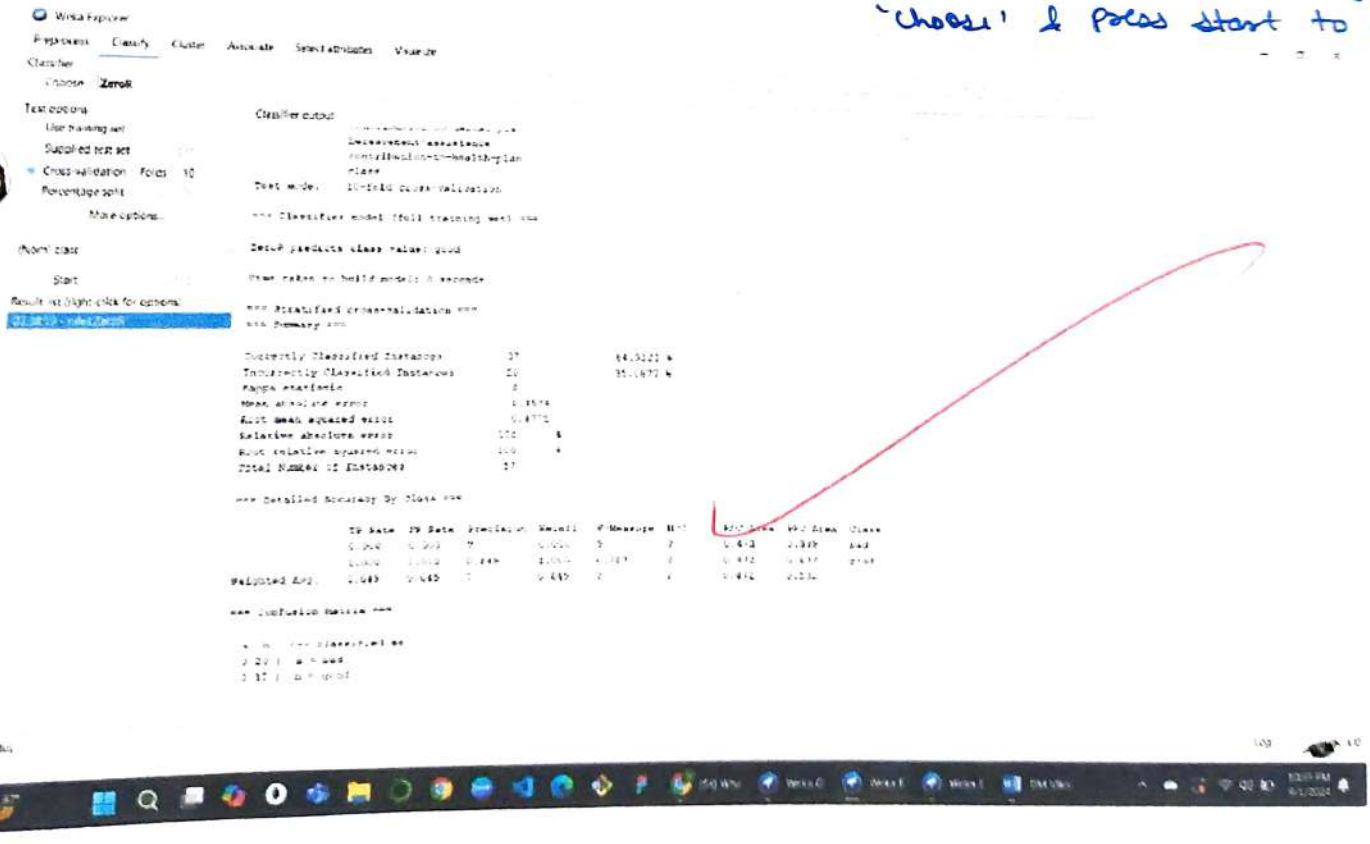


Analyzation of Labor Dataset:

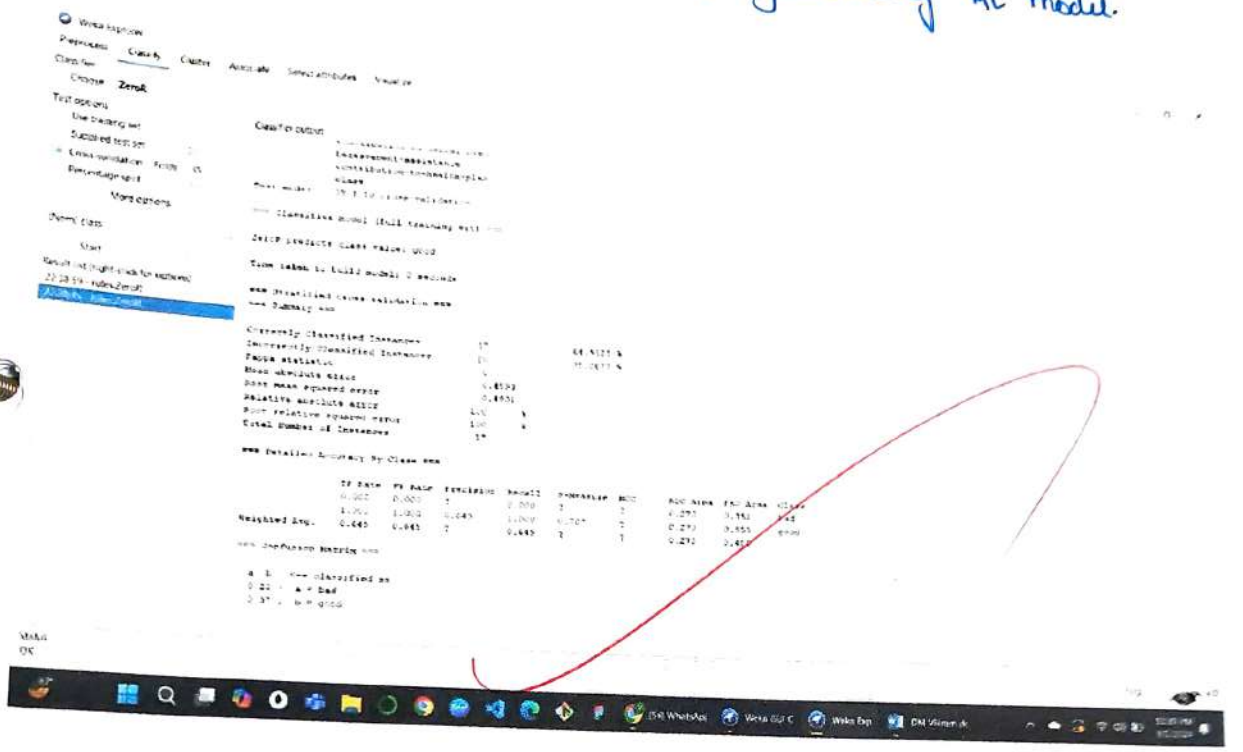


Applying ZeroR Classification on Labor dataset:

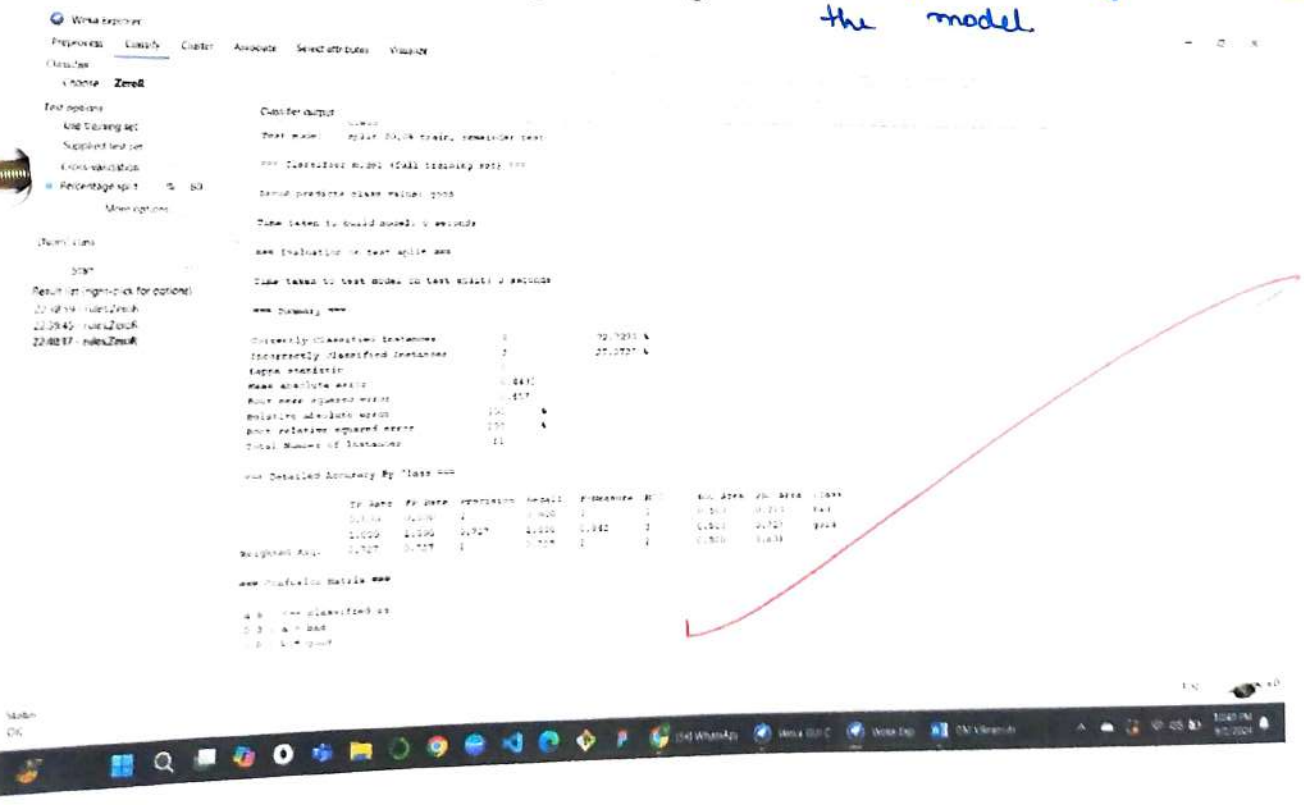
select ZeroR classification from 'choose' & press start to apply.



Changing number of Cross-Validation Folds: refers to subsets of a datasets to evaluate the performance of ML model.

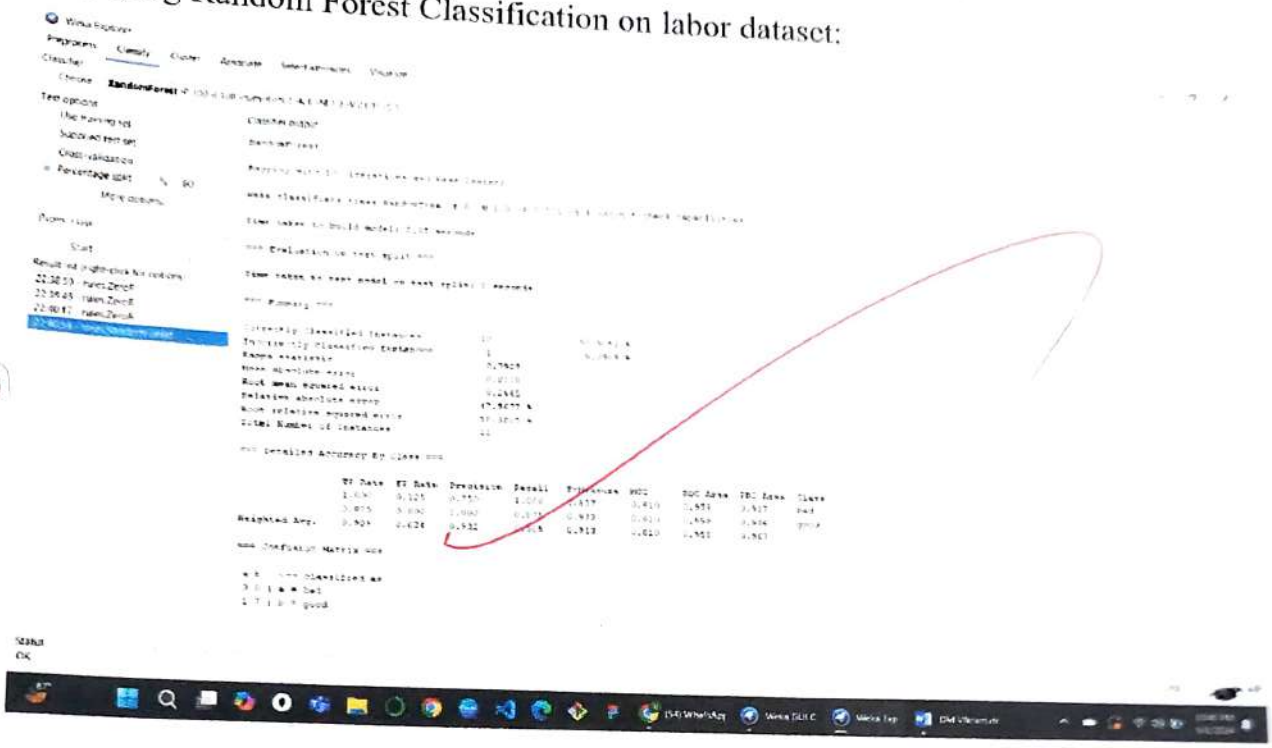


Splitting the dataset into training and testing sets: 80% for training & 20% for testing the model.

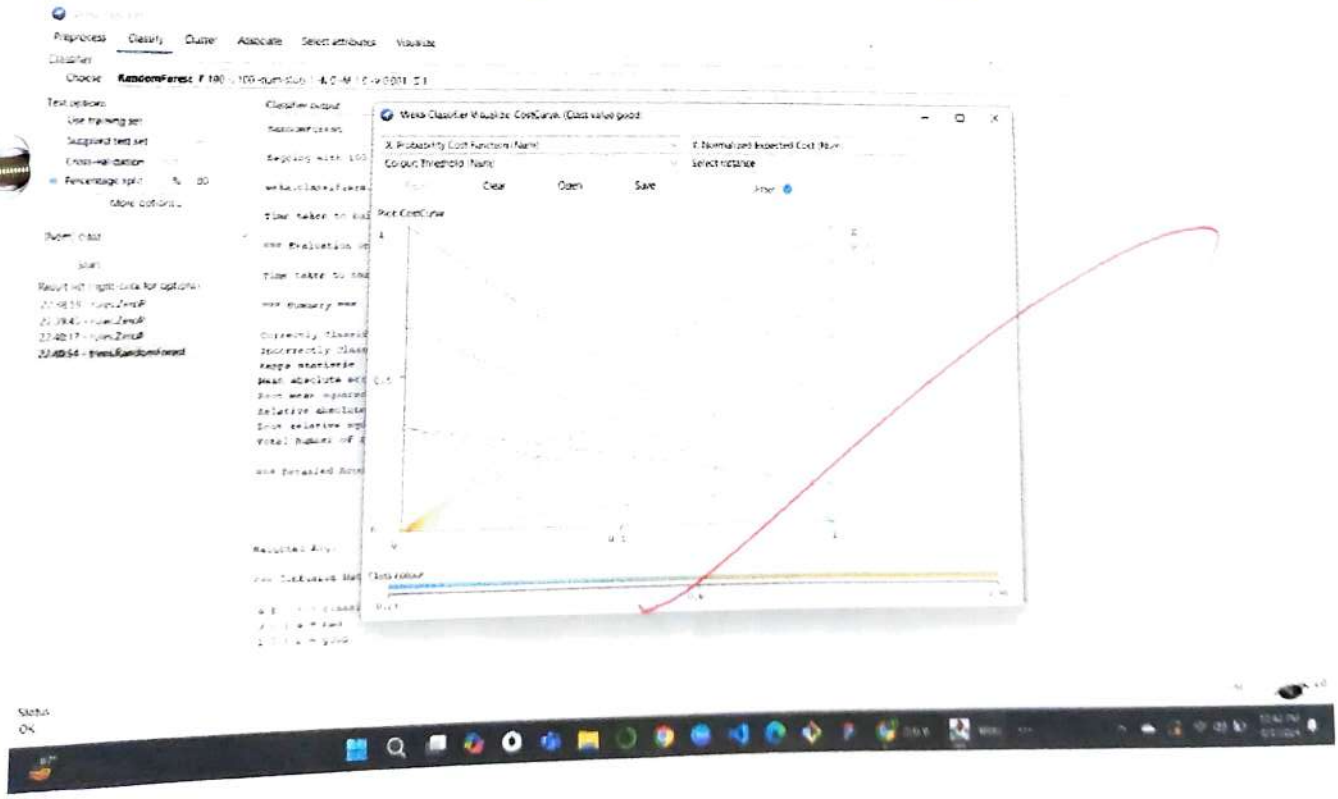


We can choose multiple classifiers in WEKA, Random forest is one of them, which is an ensemble learning method.

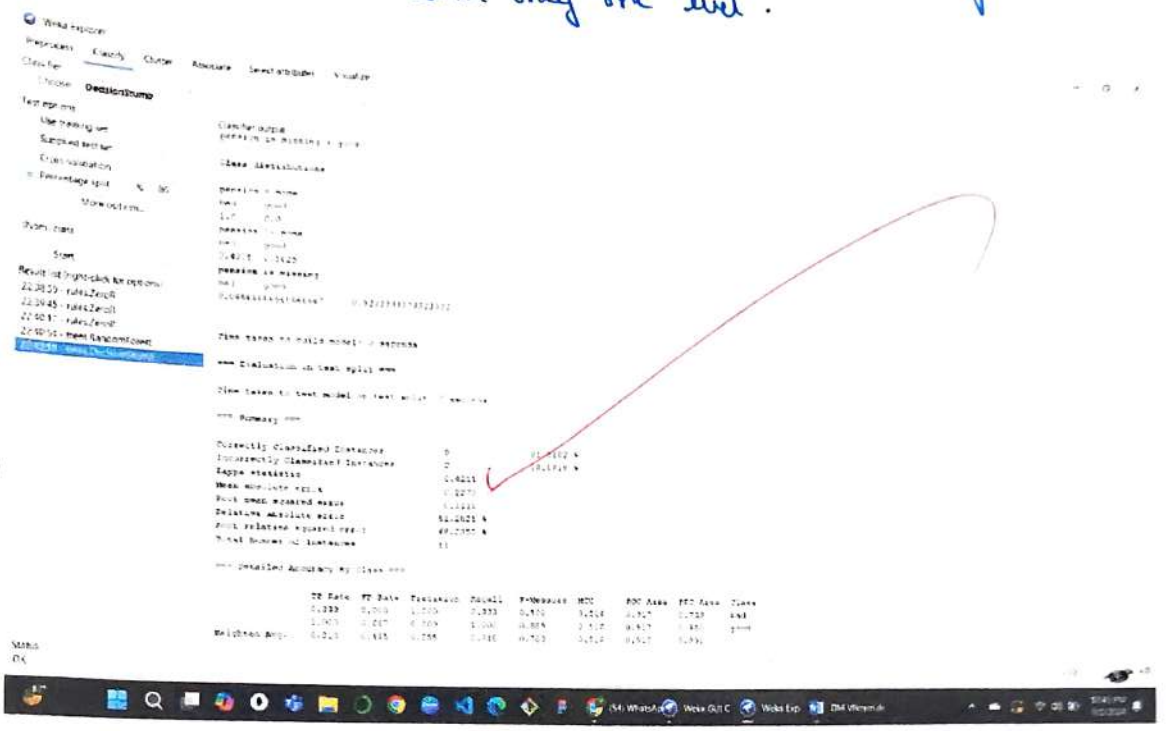
Applying Random Forest Classification on labor dataset:



Visualization of cost curve: Performance of model analysis.



Applying Decision Stump Classification on labor dataset: uses only one decision tree with only one level.



Learning Outcome:

- understood the concepts of classification.
- learnt how to classify datasets and visualize performance of model in WEKA.

Lakshita
23/09/24

Experiment 7

Objective: Evaluate the performance of classification techniques using different parameters.

Theory:

Classification is a supervised learning technique in data Mining and machine learning where a model is trained on labeled data to categorize new instances into classes.

In this experiment, we aim to explore and compare different classification techniques such as decision trees, Naive Bayes, Svm and KNN.

Key classification techniques

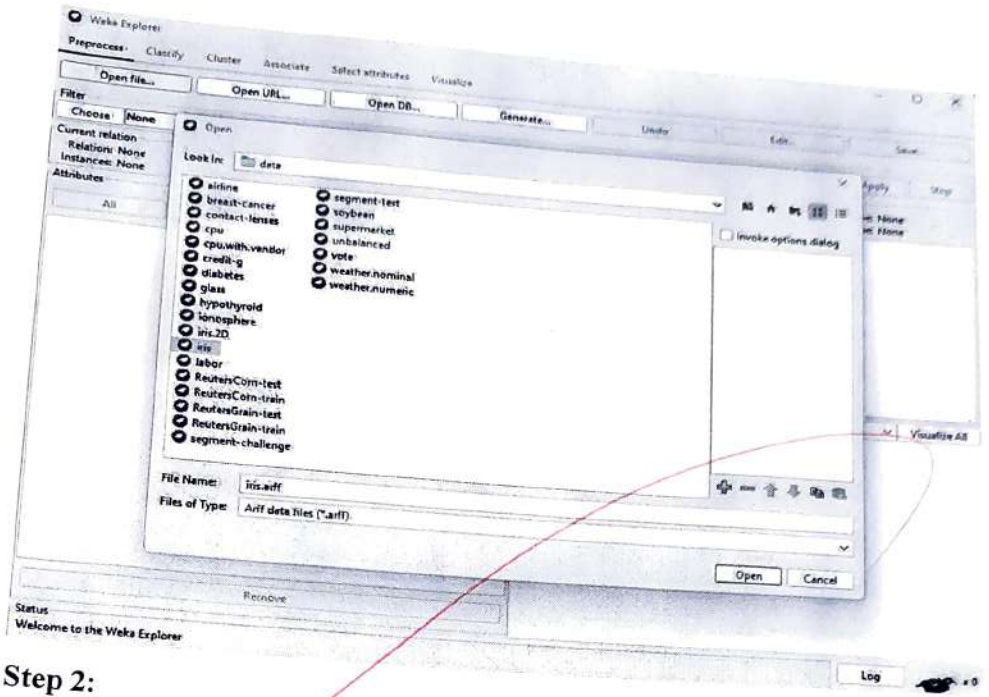
(I) Decision Trees :- A tree structured classified where each node represents a feature, each branch represents a decision rule and each leaf node represents an outcome

II) Naive Bayes :- A probilistic classification based on Baye's theorem assuming Independence between feature.

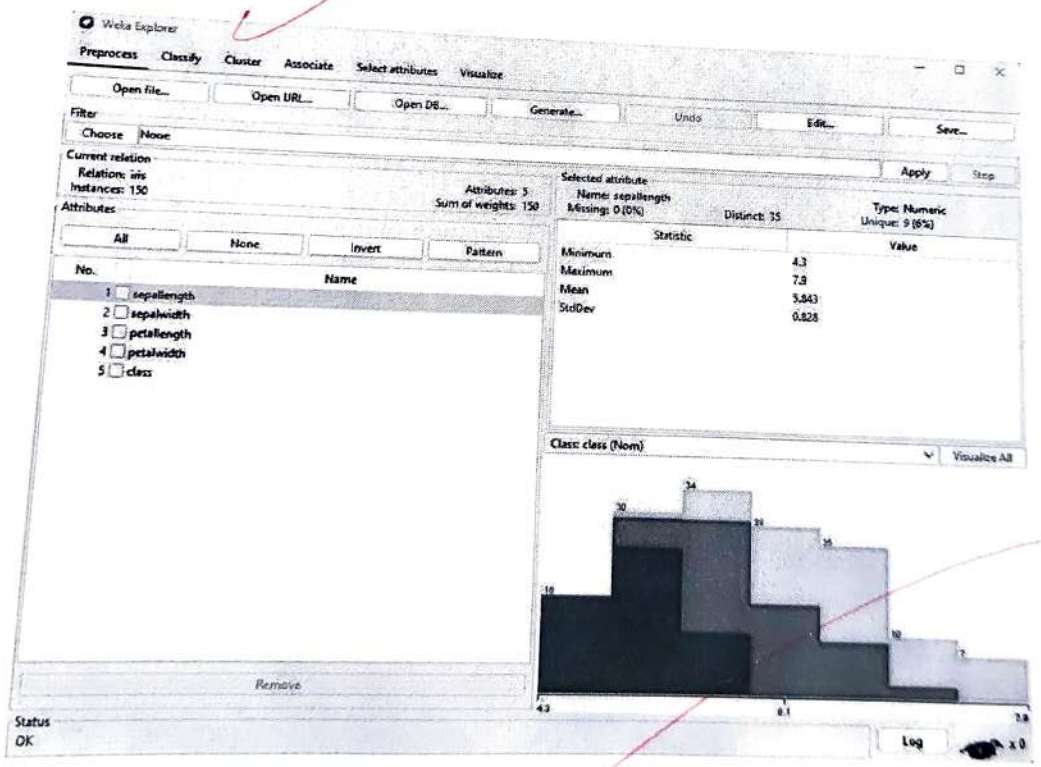
II) Support Vector Machine (SVM) :- SVM finds a hyperplane that seperates classes. It is particularly powerful for binary classification.

) K-Nearest neighbour :- A non-parametric method where classification is based on the majority label of the nearest neighbour.

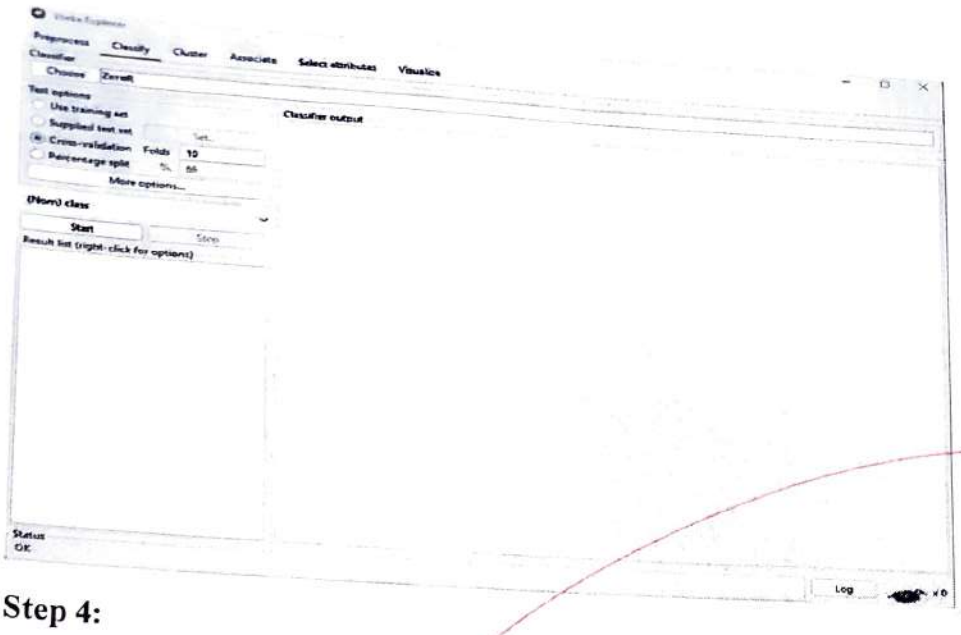
Step 1:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Current relation
Relation: None
Instances: None
Attributes
All
Open
Look in: data
airline
breast-cancer
contact-lenses
cpu
cpu.with.vendor
credit-g
diabetes
glass
hypothyroid
ionosphere
iris.2D
iris
labor
ReutersCorn-test
ReutersCorn-train
ReutersGrain-test
ReutersGrain-train
segment-challenge
segment-test
soybean
supermarket
unbalanced
vote
weather.nominal
weather.numeric
Invoke options dialog
File Name: iris.arff
Files of Type: Arff data files (*.arff)
Open
Cancel
Remove
Status
Welcome to the Weka Explorer
Log
x 0

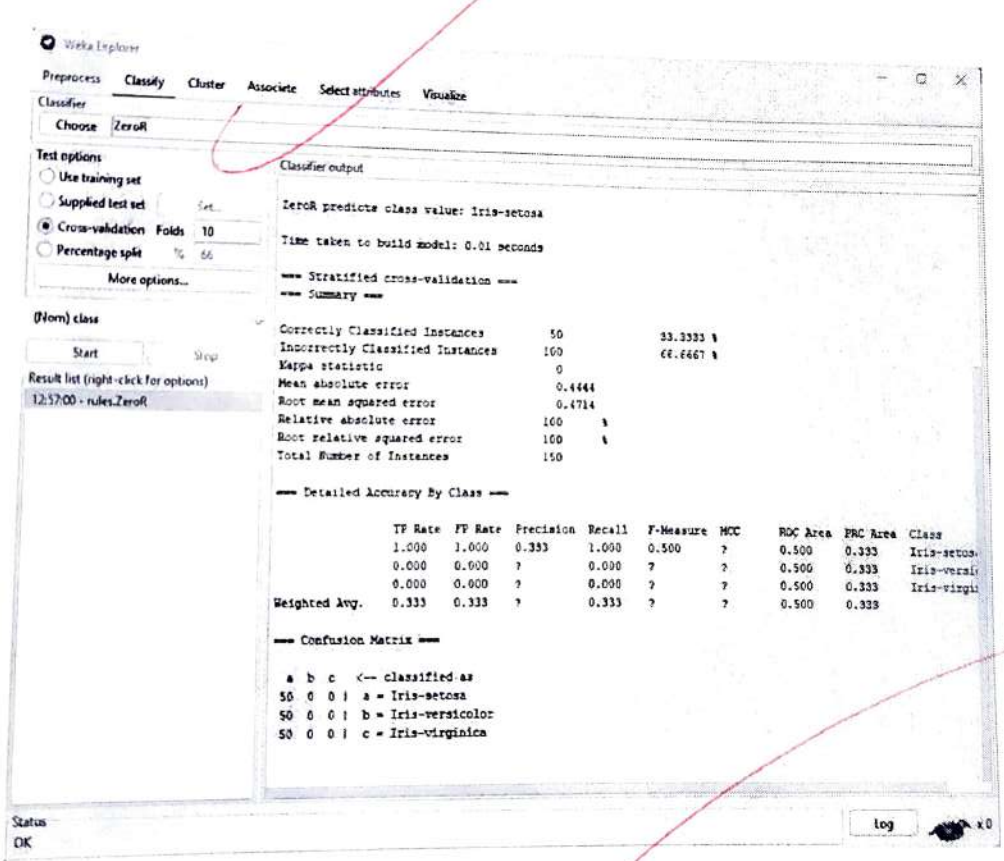
Step 2:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Current relation
Relation: iris
Instances: 150
Attributes: 5
Sum of weights: 150
Attributes
All
None
Invert
Pattern
No.
Name
1 sepallength
2 sepalwidth
3 petallength
4 petalwidth
5 class
Remove
Selected attribute
Name: sepallength
Missing: 0 (0%)
Distinct: 35
Type: Numeric
Unique: 9 (6%)
Statistic
Value
Minimum 4.3
Maximum 7.9
Mean 5.843
StdDev 0.828
Class: class (Nom)
Visualize All
Status
OK
Log
x 0

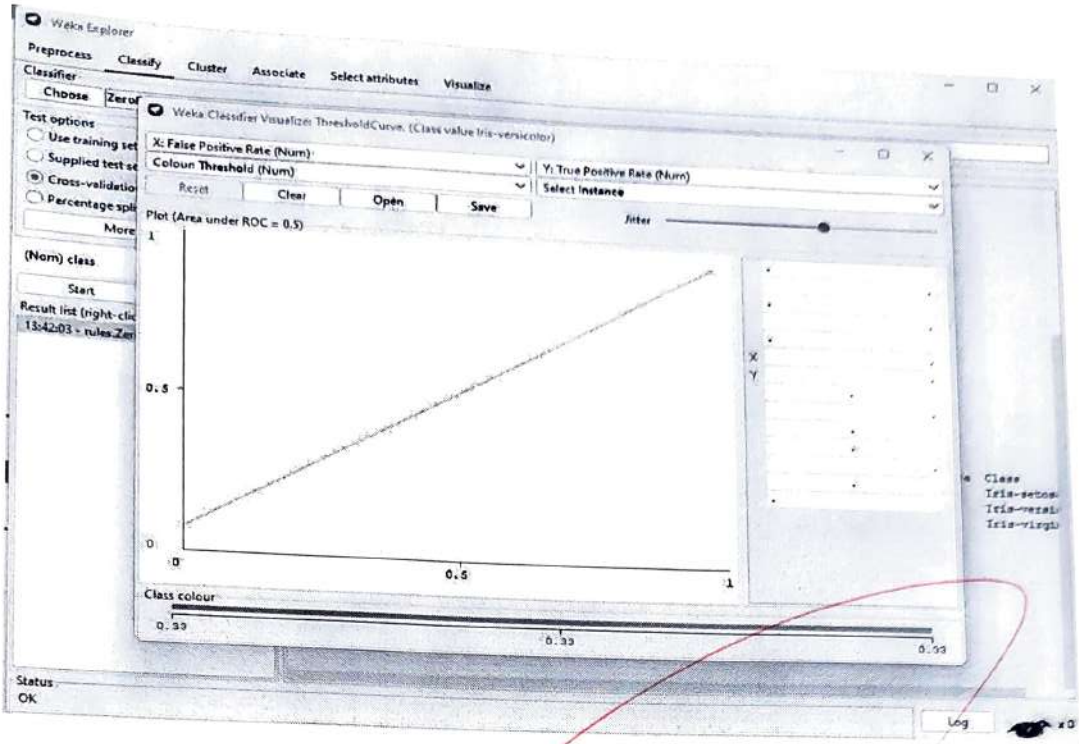
Step 3:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
ZeroR
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
Classifier output
(Nom) class
Start
Stop
Result list (right-click for options)
Status
OK
Log
x 0

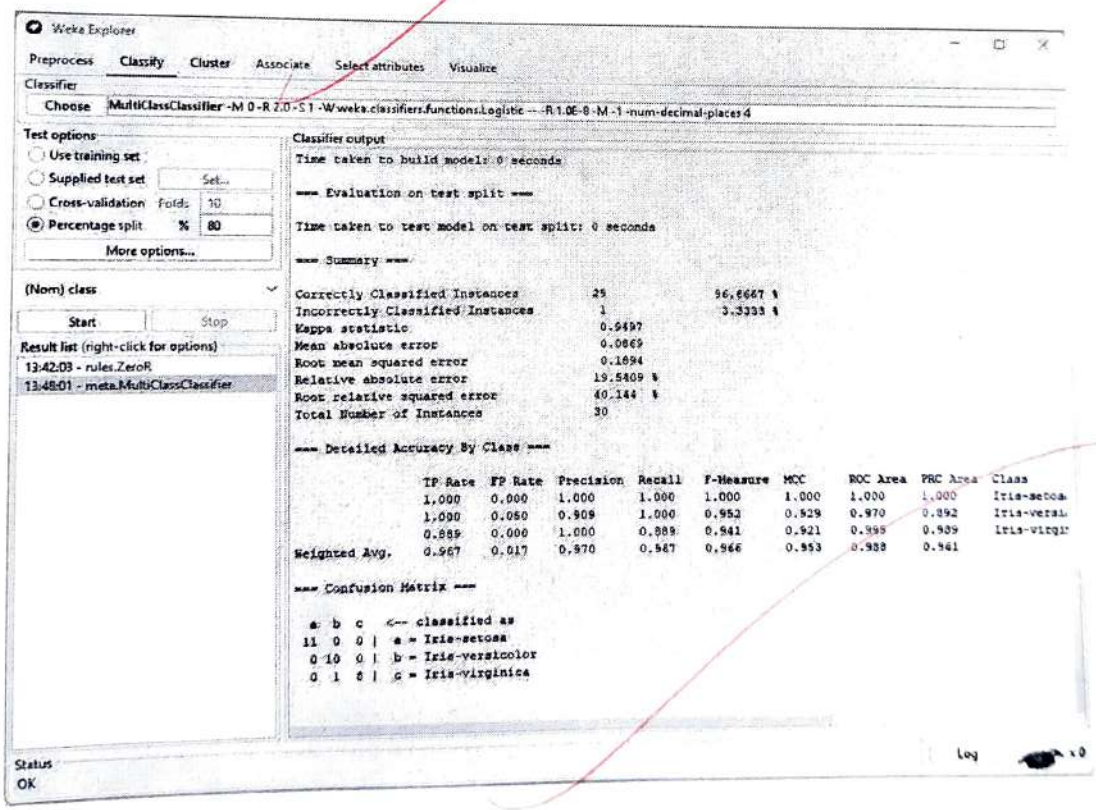
Step 4:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
ZeroR
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
Classifier output
(Nom) class
Start
Stop
Result list (right-click for options)
12:57:00 - rules.ZeroR
ZeroR predicts class value: Iris-setosa
Time taken to build model: 0.01 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances 50 33.3333 %
Incorrectly Classified Instances 100 66.6667 %
Kappa statistic 0
Mean absolute error 0.4444
Root mean squared error 0.4714
Relative absolute error 100 %
Root relative squared error 100 %
Total Number of Instances 150
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
1.000 1.000 0.333 1.000 0.500 ? 0.500 0.333 Iris-setos
0.000 0.000 ? 0.000 ? ? 0.500 0.333 Iris-versi
0.000 0.000 ? 0.000 ? ? 0.500 0.333 Iris-virgi
Weighted Avg. 0.333 0.333 ? 0.333 ? ? 0.500 0.333
=== Confusion Matrix ===
a b c <-- classified as
50 0 0 | a = Iris-setosa
50 0 0 | b = Iris-versicolor
50 0 0 | c = Iris-virginica
Status
OK
log
x 0

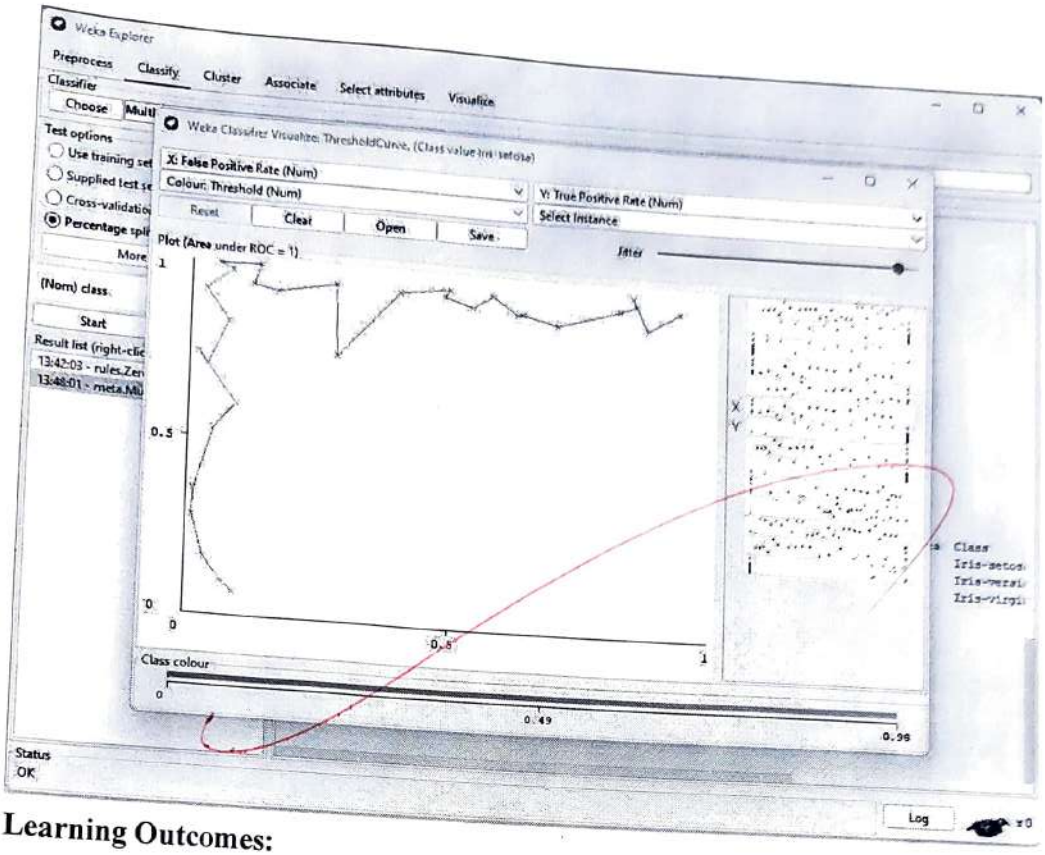
Step 5 :


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Weka Classifier Visualize: ThresholdCurve: (Class value Iris-versicolor)
X: False Positive Rate (Num)
Y: True Positive Rate (Num)
Colour
Threshold (Num)
Select Instance
Reset
Clear
Open
Save
Jitter
Plot (Area under ROC = 0.5)
Class colour
Status
OK
Log

Step 6:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
MultiClassClassifier -M 0 -R 2.0 -S 1 -W weka.classifiers.functions.Logistic -- -R 1.0E-8 -M -1 -num-decimal-places 4
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
80
More options...
(Nom) class
Start
Stop
Result list (right-click for options)
13:42:03 - rules.ZeroR
13:48:01 - meta.MultiClassClassifier
Classifier output
Time taken to build model: 0 seconds
=== Evaluation on test split ===
Time taken to test model on test split: 0 seconds
=== Summary ===
Correctly Classified Instances 25 96.6667 %
Incorrectly Classified Instances 1 3.3333 %
Kappa statistic 0.9497
Mean absolute error 0.0869
Root mean squared error 0.1694
Relative absolute error 19.5409 %
Root relative squared error 40.144 %
Total Number of Instances 30
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
1,000 0,000 1,000 1,000 1,000 1,000 1,000 1,000 Iris-setosa
1,000 0,050 0,909 1,000 0,952 0,929 0,970 0,892 Iris-versi
0,889 0,000 1,000 0,889 0,941 0,921 0,995 0,989 Iris-virgi
Weighted Avg. 0,967 0,017 0,970 0,967 0,966 0,953 0,988 0,961
=== Confusion Matrix ===
a b c <-- classified as
11 0 0 | a = Iris-setosa
0 10 0 | b = Iris-versicolor
0 1 8 | c = Iris-virginica
Status
OK
Log

Step 7:



Learning Outcomes:

→ understood various classification techniques like svm, decision trees, Naive Bayes & KNN.

→ learnt to implement & evaluate performance of classification techniques using different parameters.

Experiment 8

Objective: Implementation of Bagging and Boosting techniques on ARFF files using WEKA.

Theory:

Bagging :- It's an ensemble learning method that improves the stability and accuracy of ml algorithms. It involves creating multiple subsets of training data by random sampling, training a seperate model on each subset and then averaging the prediction for regression and majority voting for classification.

Boosting :- Boosting is another ensemble technique that aims to reduce bias and improve predictive performance by sequentially training models.

ARFF files :- [Attribute. Relation file format] files stored in a format, readable by the WEKA.

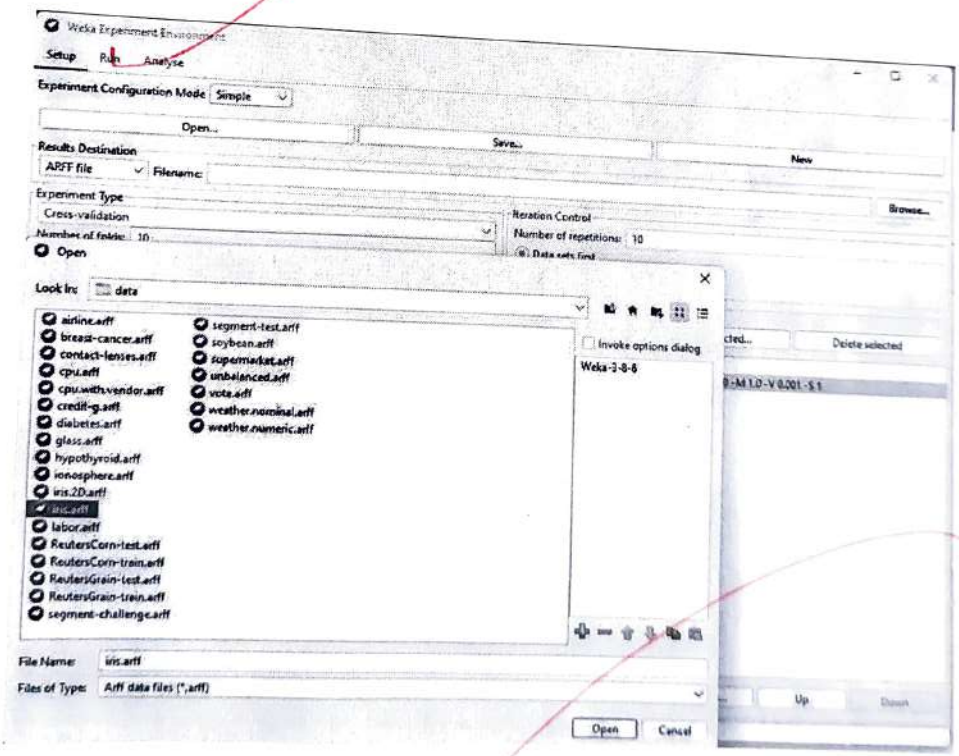
Procedure:-

1) Load ARFF file
2) Choose bagging / boosting under classify tab, select an algorithm like decision trees as a meta classifier.
3) Adjust parameters such as no. of iteration or sample size.
4) Execute the algorithm and review the performance.

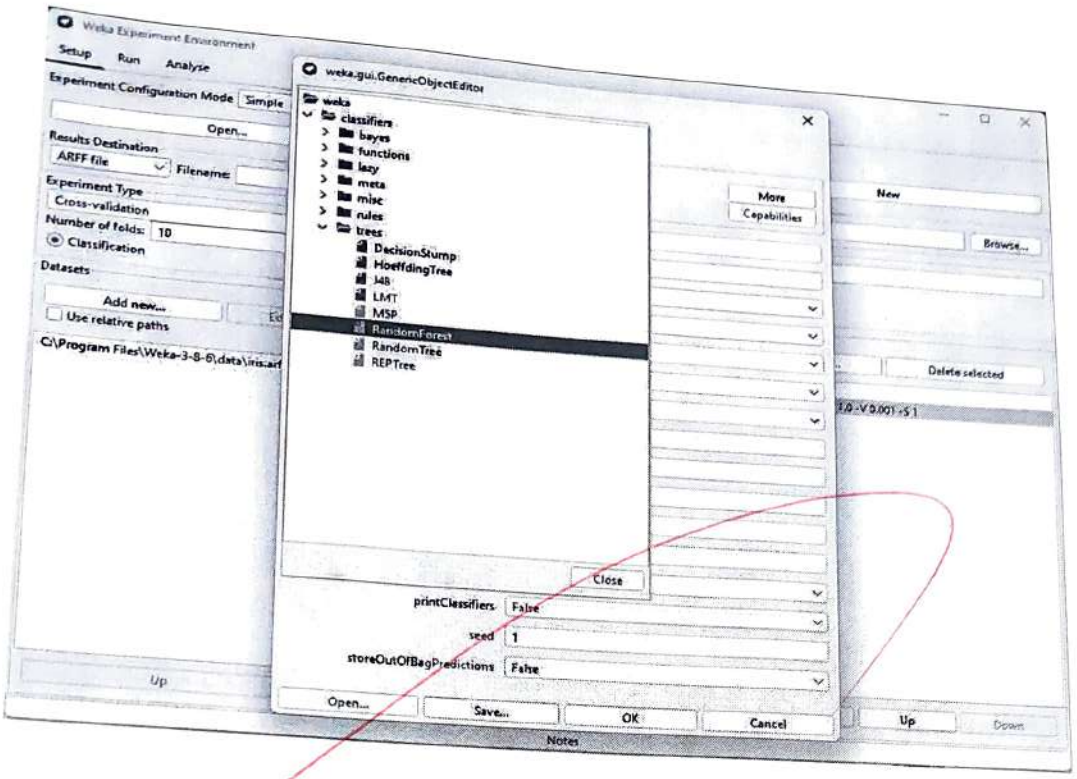
Step 1:


Weka Experiment Environment
Setup
Run
Analyse
Experiment Configuration Mode
Simple
Open...
Results Destination
ARFF file
Filename:
Experiment Type
Cross-validation
Number of folds:
Classification
Regression
Datasets
Add new...
Edit selected...
Delete selected
Use relative paths
Iteration Control
Number of repetitions:
Data sets first
Algorithms first
Algorithms
Add new...
Edit selected...
Delete selected
Up
Down
Load options...
Save options...
Notes

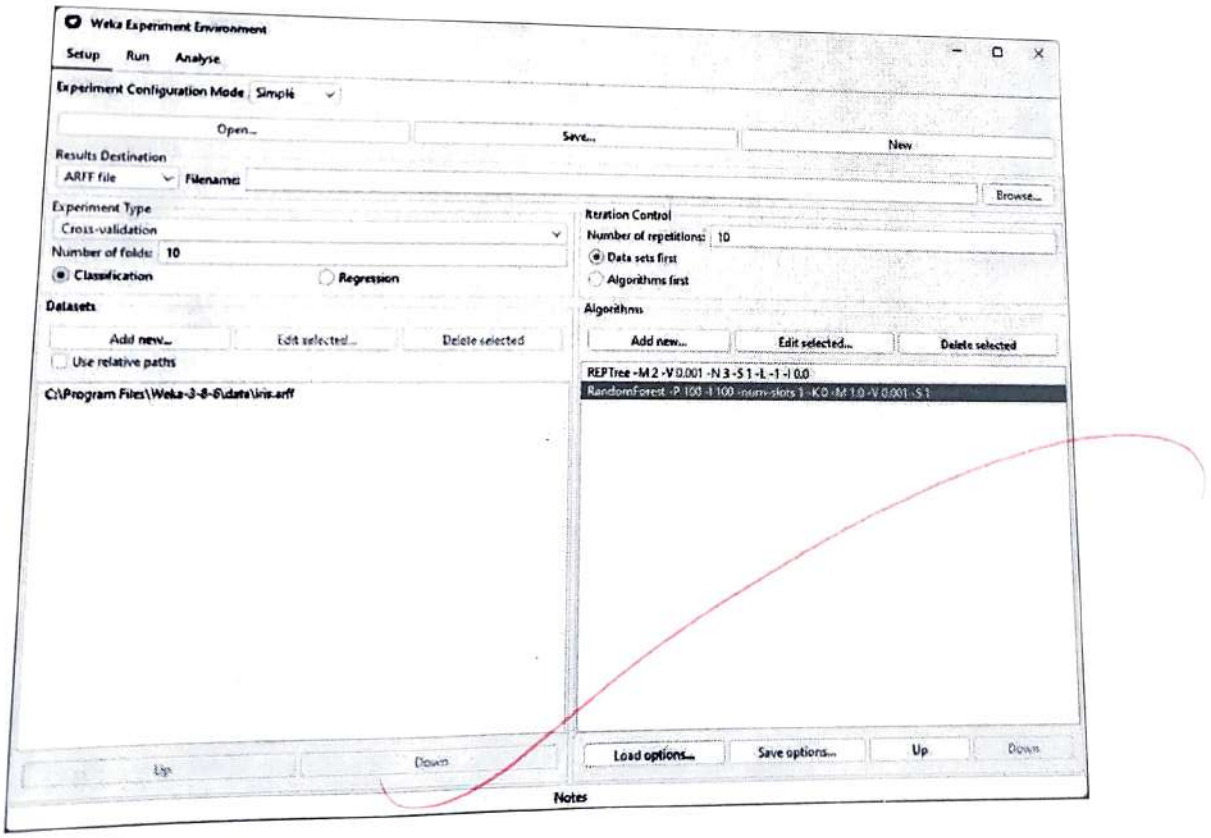
Step 2:


Weka Experiment Environment
Setup
Run
Analyse
Experiment Configuration Mode
Simple
Open...
Save...
New
Results Destination
ARFF file
Filename:
Experiment Type
Cross-validation
Iteration Control
Number of repetitions: 10
Open
Look in:
data
airline.arff
breast-cancer.arff
contact-lenses.arff
cpu.arff
cpu.with.vendor.arff
credit-g.arff
diabetes.arff
glass.arff
hypothyroid.arff
ionosphere.arff
iris.2D.arff
iris.arff
labor.arff
ReutersCorn-test.arff
ReutersCorn-train.arff
ReutersGrain-test.arff
ReutersGrain-train.arff
segment-challenge.arff
segment-test.arff
soybean.arff
supermarket.arff
unbalanced.arff
vote.arff
weather.nominal.arff
weather.numeric.arff
Invoke options dialog
Weka-3-8-6
File Name:
iris.arff
Files of Type:
Arff data files (*.arff)
Open
Cancel
Delete selected
Up
Down

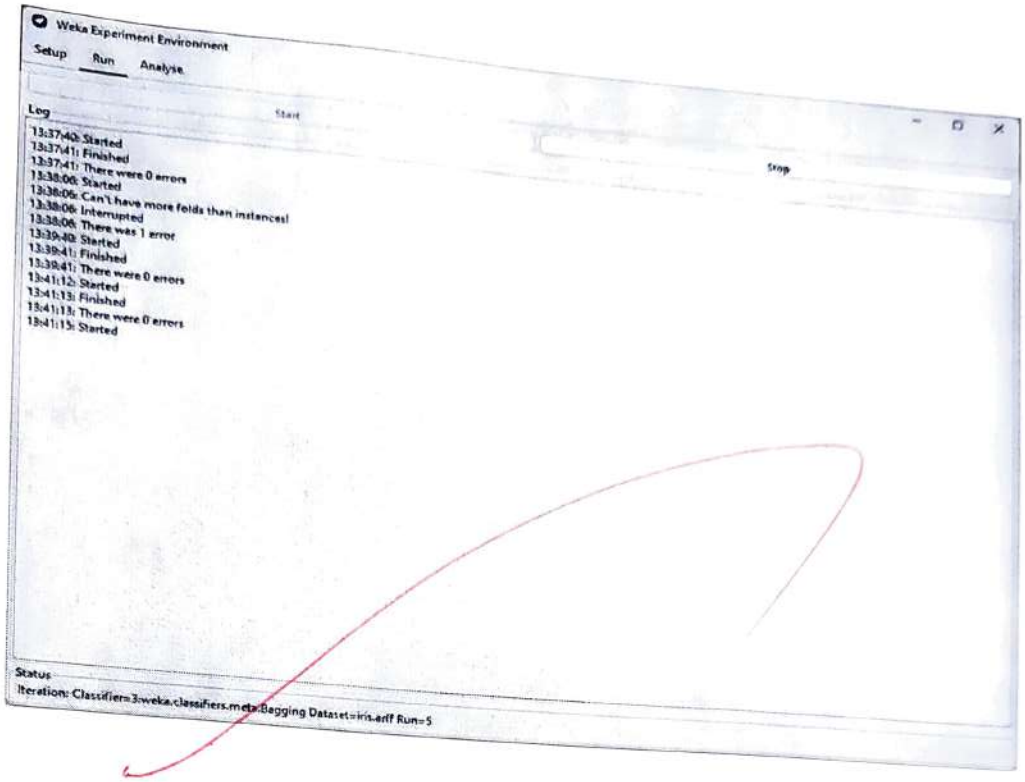
Step 3:


Weka Experiment Environment
Setup
Run
Analyse
Experiment Configuration Mode
Simple
Open...
Results Destination
ARFF file
Filename:
Experiment Type
Cross-validation
Number of folds:
10
Classification
Datasets
Add new...
Use relative paths
C:\Program Files\Weka-3-8-6\data\iris.arff
Up
weka.gui.GenericObjectEditor
weka
classifiers
bayes
functions
lazy
meta
misc
rules
trees
DecisionStump
HoeffdingTree
J48
LMT
M5P
RandomForest
RandomTree
REPTree
Close
More
Capabilities
printClassifiers
False
seed
1
storeOutOfBagPredictions
False
Open...
Save...
OK
Cancel
Notes
New
Browse...
Delete selected
Up
Down

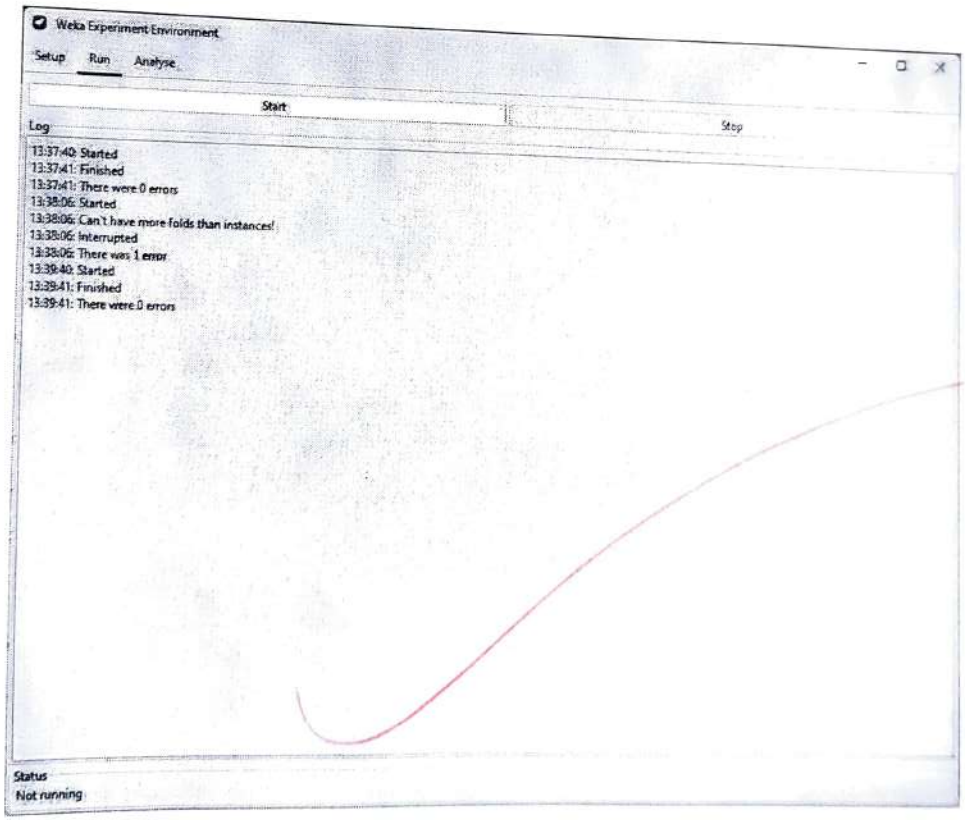
Step 4:


Weka Experiment Environment
Setup
Run
Analyse
Experiment Configuration Mode
Simple
Open...
Save...
New
Results Destination
ARFF file
Filename:
Browse...
Experiment Type
Cross-validation
Number of folds:
10
Classification
Regression
Datasets
Add new...
Edit selected...
Delete selected
Use relative paths
C:\Program Files\Weka-3-8-6\data\iris.arff
Iteration Control
Number of repetitions:
10
Data sets first
Algorithms first
Algorithms
Add new...
Edit selected...
Delete selected
REPTree -M 2 -V 0.001 -N 3 -S 1 -L -1 -I 0.0
RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1
Up
Down
Load options...
Save options...
Up
Down
Notes

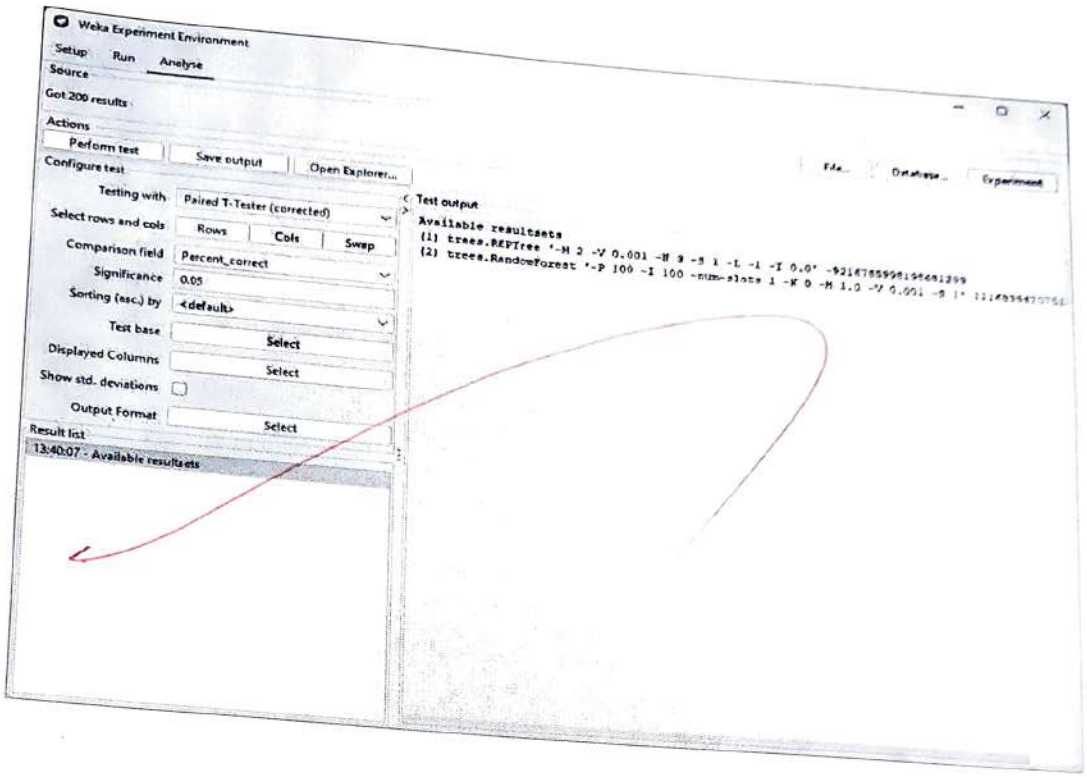
Step 5:


Weka Experiment Environment
Setup
Run
Analyse
Start
Stop
Log
13:37:40: Started
13:37:41: Finished
13:37:41: There were 0 errors
13:38:06: Started
13:38:06: Can't have more folds than instances!
13:38:06: Interrupted
13:38:06: There was 1 error
13:39:40: Started
13:39:41: Finished
13:39:41: There were 0 errors
13:41:12: Started
13:41:13: Finished
13:41:13: There were 0 errors
13:41:15: Started
Status
Iteration: Classifier=3:weka.classifiers.meta.Bagging Dataset=iris.arff Run=5

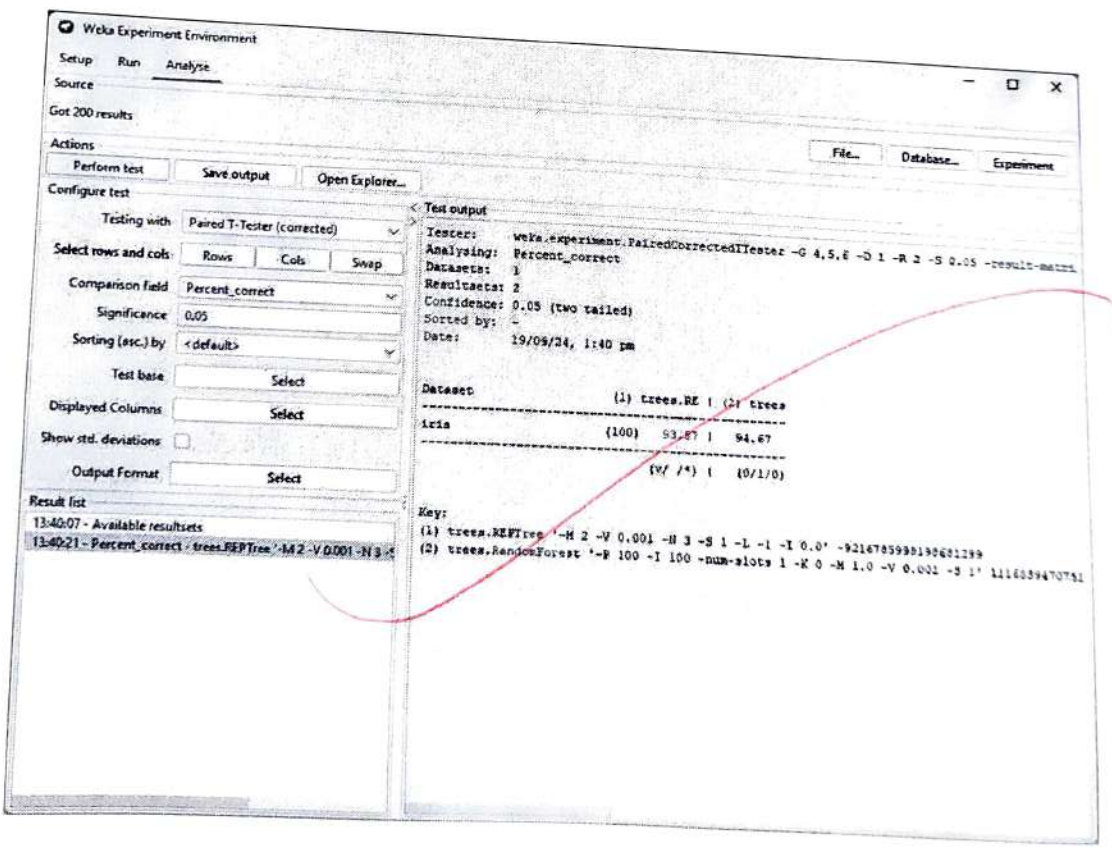
Step 6:


Weka Experiment Environment
Setup
Run
Analyse
Start
Stop
Log
13:37:40: Started
13:37:41: Finished
13:37:41: There were 0 errors
13:38:06: Started
13:38:06: Can't have more folds than instances!
13:38:06: Interrupted
13:38:06: There was 1 error
13:39:40: Started
13:39:41: Finished
13:39:41: There were 0 errors
Status
Not running

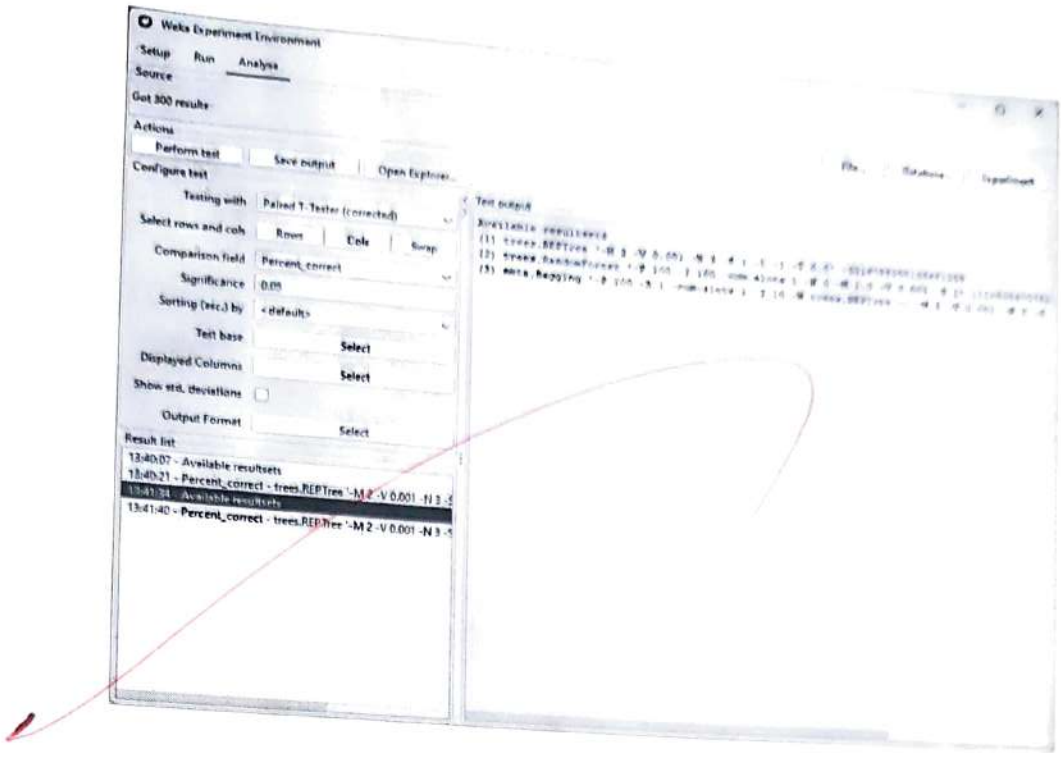
Step 7:


Weka Experiment Environment
Setup
Run
Analyse
Source
Got 200 results
Actions
Perform test
Save output
Open Explorer...
Configure test
Testing with
Paired T-Tester (corrected)
Select rows and cols
Rows
Cols
Swap
Comparison field
Percent_correct
Significance
0.05
Sorting (asc.) by
<default>
Test base
Select
Displayed Columns
Select
Show std. deviations
Output Format
Select
Result list
13:40:07 - Available resultsets
Test output
Available resultsets
(1) trees.REPTree '-M 2 -V 0.001 -N 3 -S 1 -L -1 -I 0.0' -9216785998198681299
(2) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 1116039470751
File...
Database...
Experiment

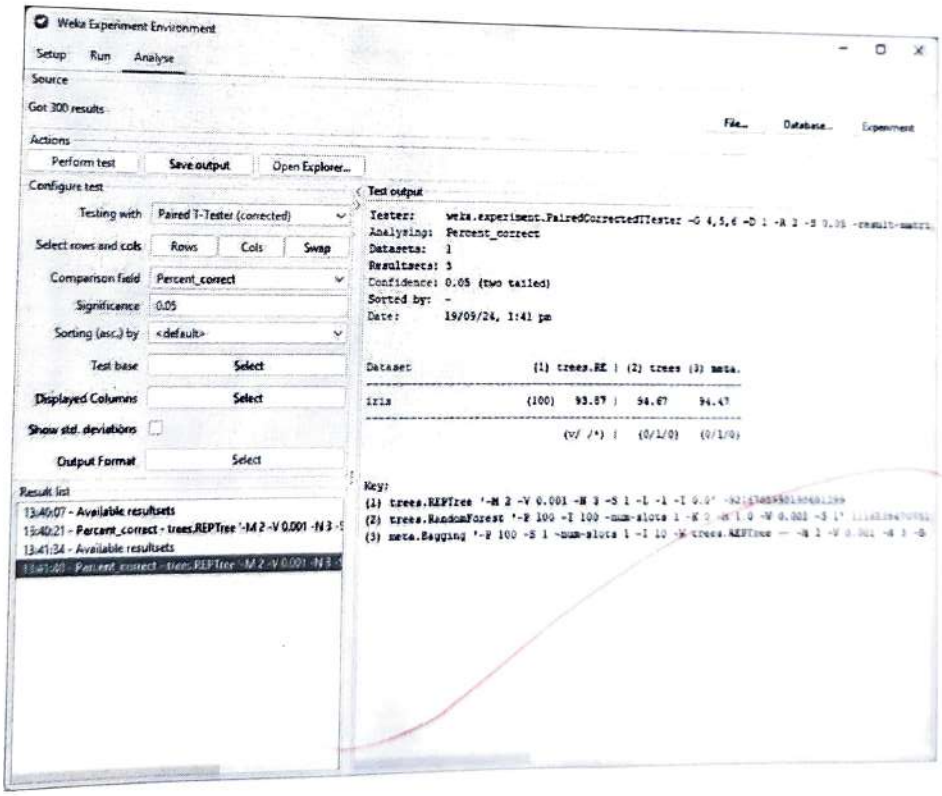
Step 8:


Weka Experiment Environment
Setup
Run
Analyse
Source
Got 200 results
Actions
Perform test
Save output
Open Explorer...
Configure test
Testing with
Paired T-Tester (corrected)
Select rows and cols
Rows
Cols
Swap
Comparison field
Percent_correct
Significance
0.05
Sorting (asc.) by
<default>
Test base
Select
Displayed Columns
Select
Show std. deviations
Output Format
Select
Result list
13:40:07 - Available resultsets
13:40:21 - Percent_correct - trees.REPTree '-M 2 -V 0.001 -N 3 -S
Test output
Tester: weka.experiment.PairedCorrectedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matrix
Analysing: Percent_correct
Datasets: 1
Resultsets: 2
Confidence: 0.05 (two tailed)
Sorted by: -
Date: 19/09/24, 1:40 pm
Dataset (1) trees.RE | (2) trees
iris (100) 93.87 | 94.67
(V/ /*) | (0/1/0)
Key:
(1) trees.REPTree '-M 2 -V 0.001 -N 3 -S 1 -L -1 -I 0.0' -9216785998198681299
(2) trees.RandomForest '-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1' 1116039470751
File...
Database...
Experiment

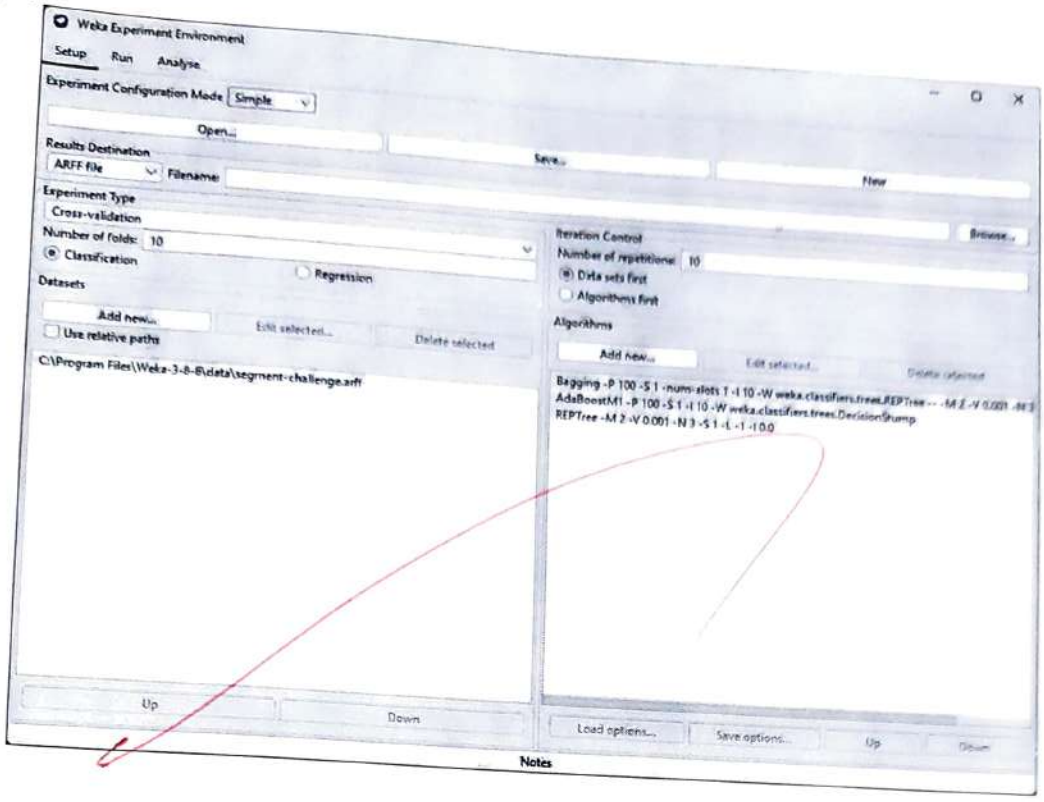
Step 9:



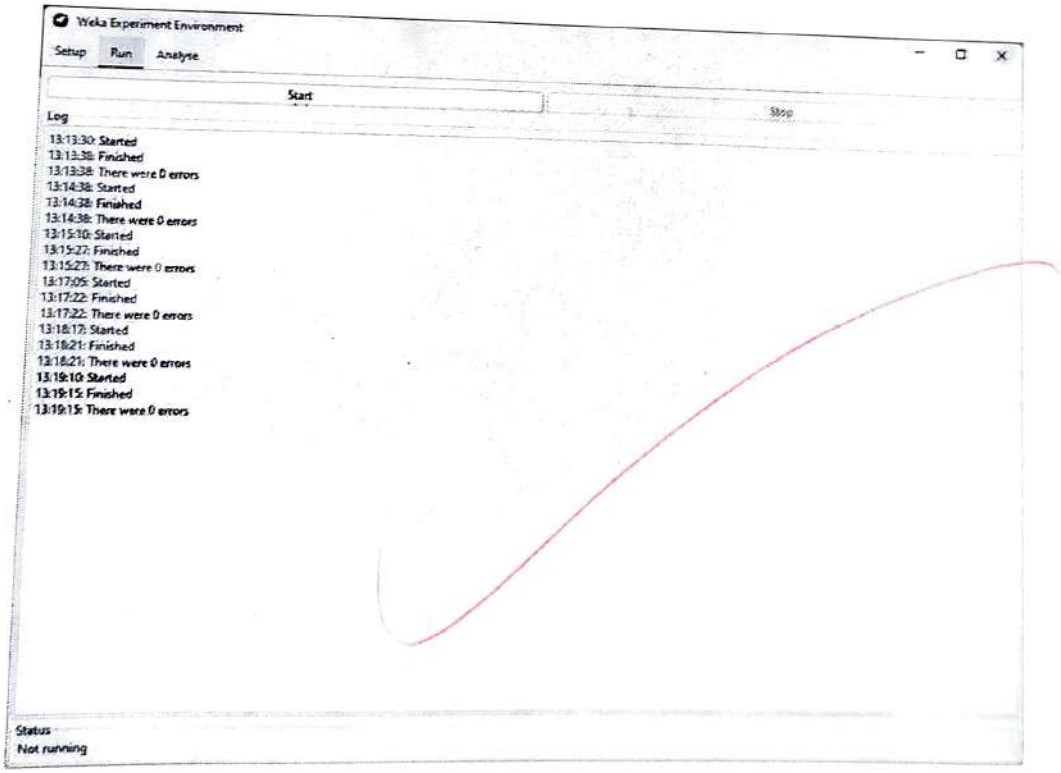
Step 10:



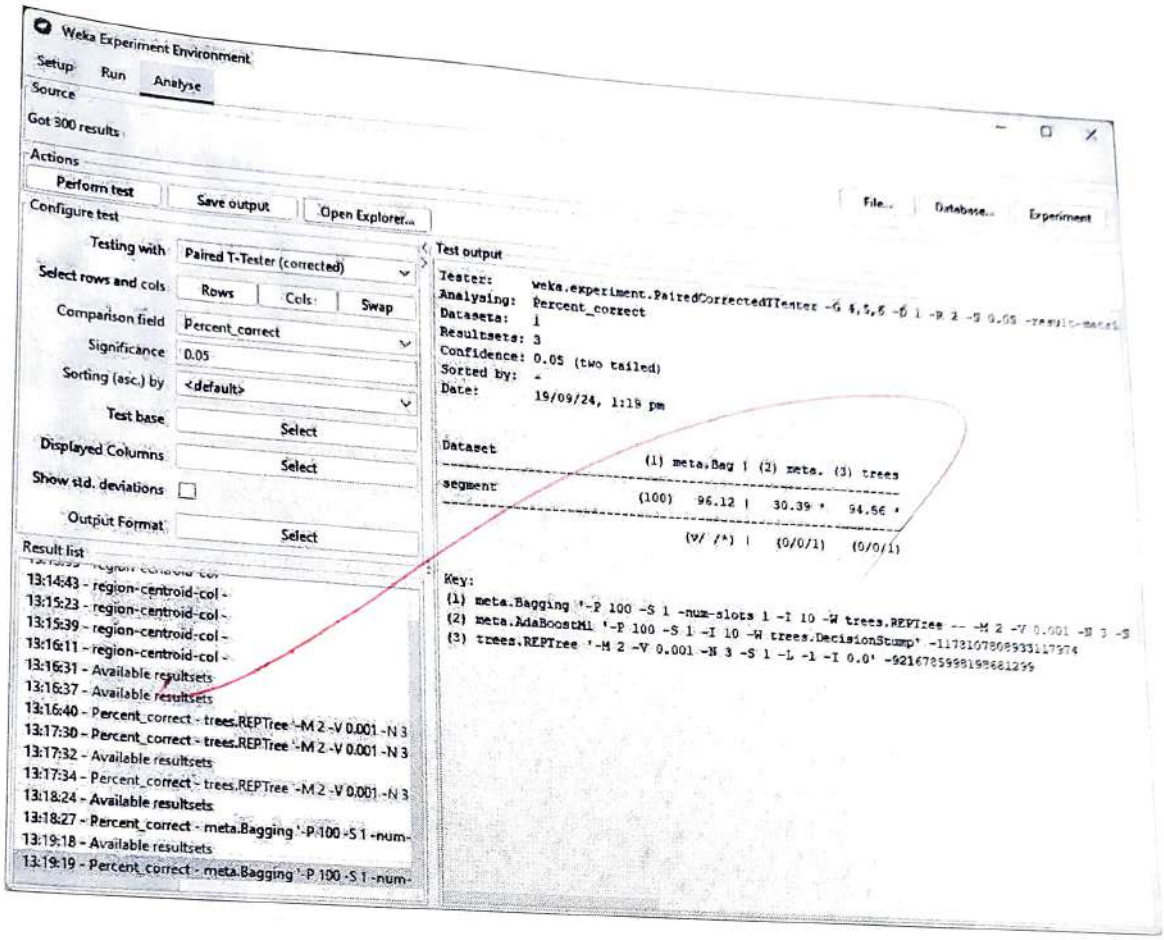
Step 11:



Step 12:



Step 13:



Learning Outcomes:

learnt how ensemble techniques can improve classification accuracy, reduce variance (bagging) and address misclassification bias (boosting).

Varshita 8/11/24

Experiment 9

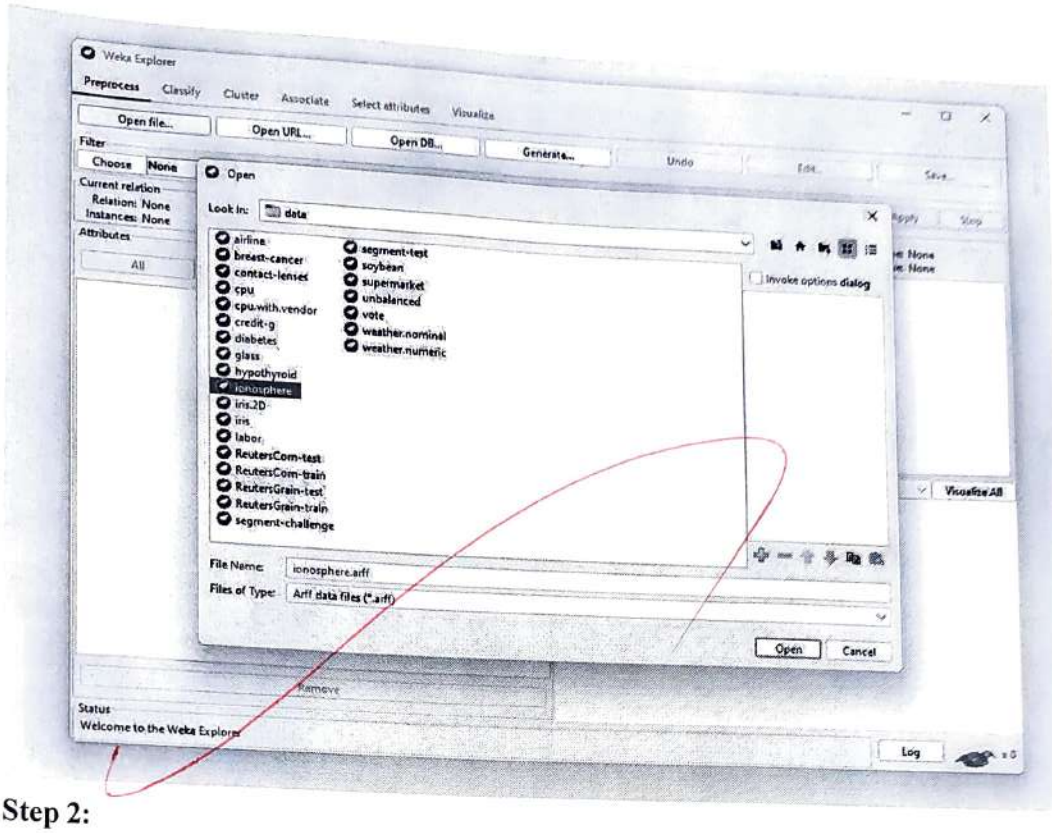
Objective: Apply the concept of Voting ensemble method to ARFF files and compare the results with single classifiers.

Theory:

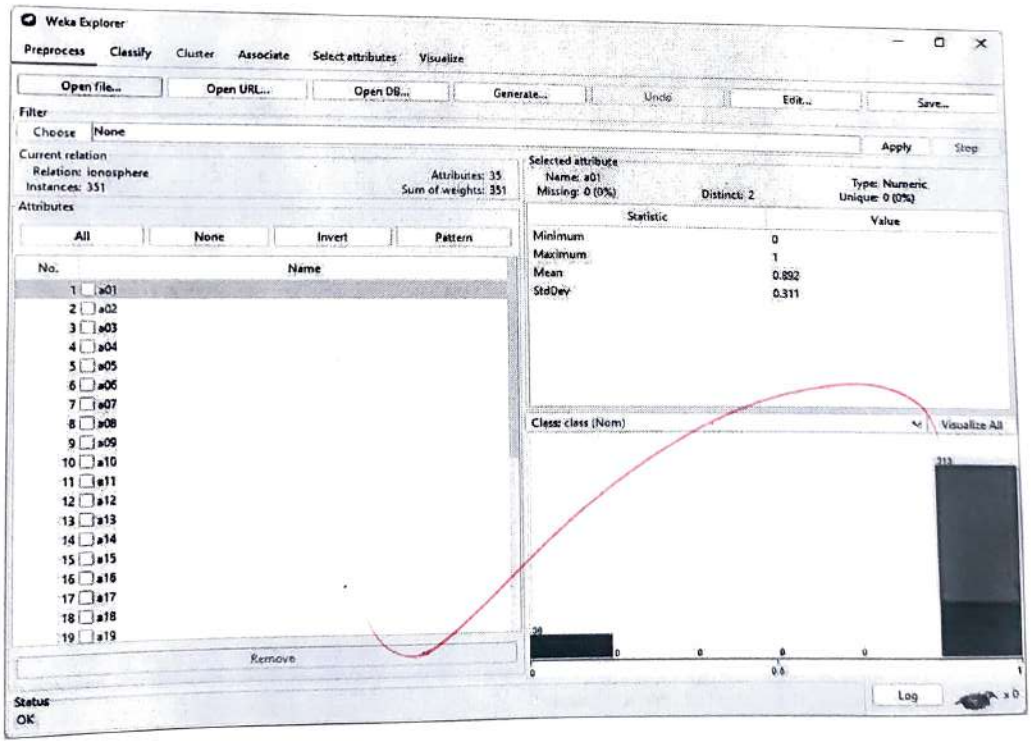
Voting Ensemble method :- The voting ensemble technique where multiple ML models (classifiers) are combined to improve prediction accuracy. In this approach several models like decision trees, Naive Bayes and SVM are trained on the same dataset and their predictions aggregated by voting for classification. Voting can be either majority voting or weighted voting.

Single classifier vs Voting ensemble :- Single classifier such as decision trees make predictions based on one model's perspective, which may lead to bias or variance issue, in contrast the voting ensemble method combine the strengths of multiple classifier leading to higher accuracy.

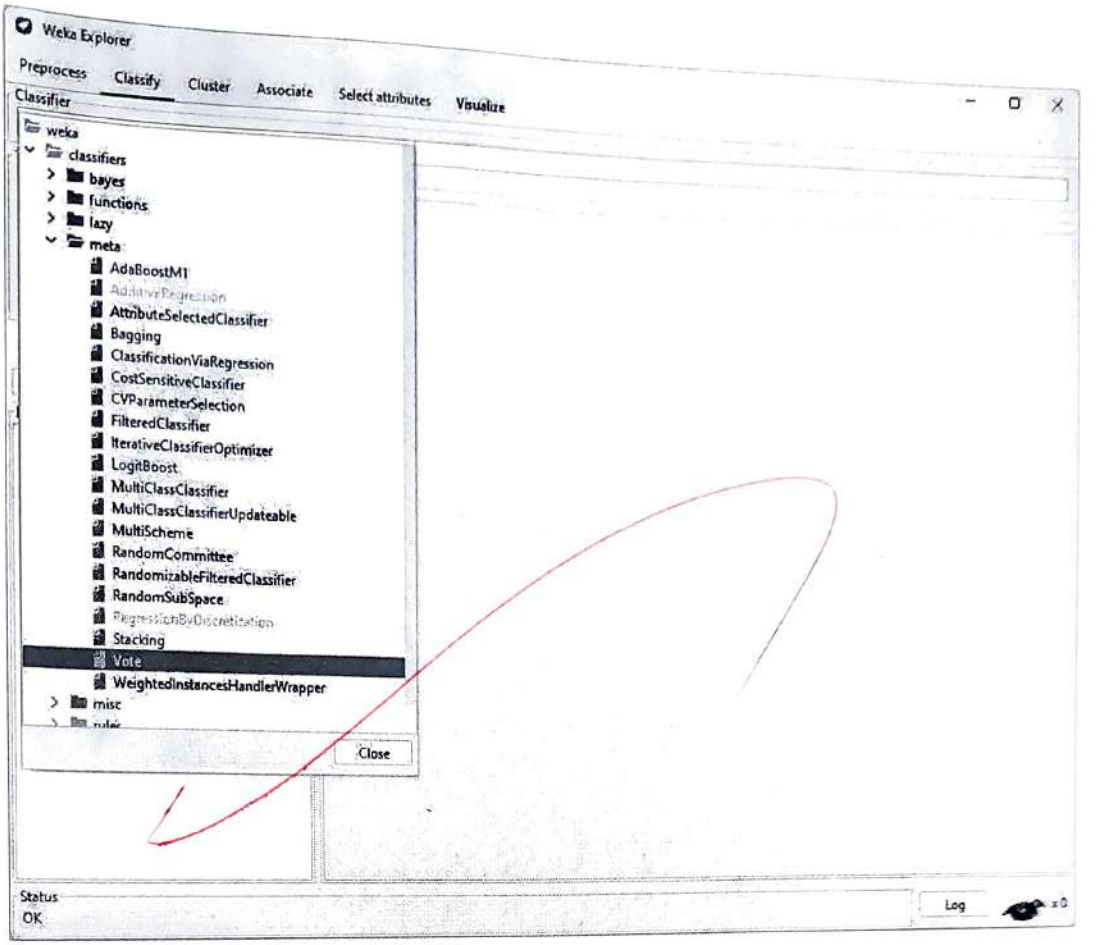
Step 1:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Current relation
Relation: None
Instances: None
Attributes
All
Open
Look In:
data
airline
breast-cancer
contact-lenses
cpu
cpu.with.vendor
credit-g
diabetes
glass
hypothyroid
ionosphere
iris.2D
iris
labor
ReutersCorn-test
ReutersCorn-train
ReutersGrain-test
ReutersGrain-train
segment-challenge
segment-test
soybean
supermarket
unbalanced
vote
weather.nominal
weather.numeric
Invoke options dialog
File Name:
ionosphere.arff
Files of Type:
Arff data files (*.arff)
Open
Cancel
Remove
Status
Welcome to the Weka Explorer
Log
x 0

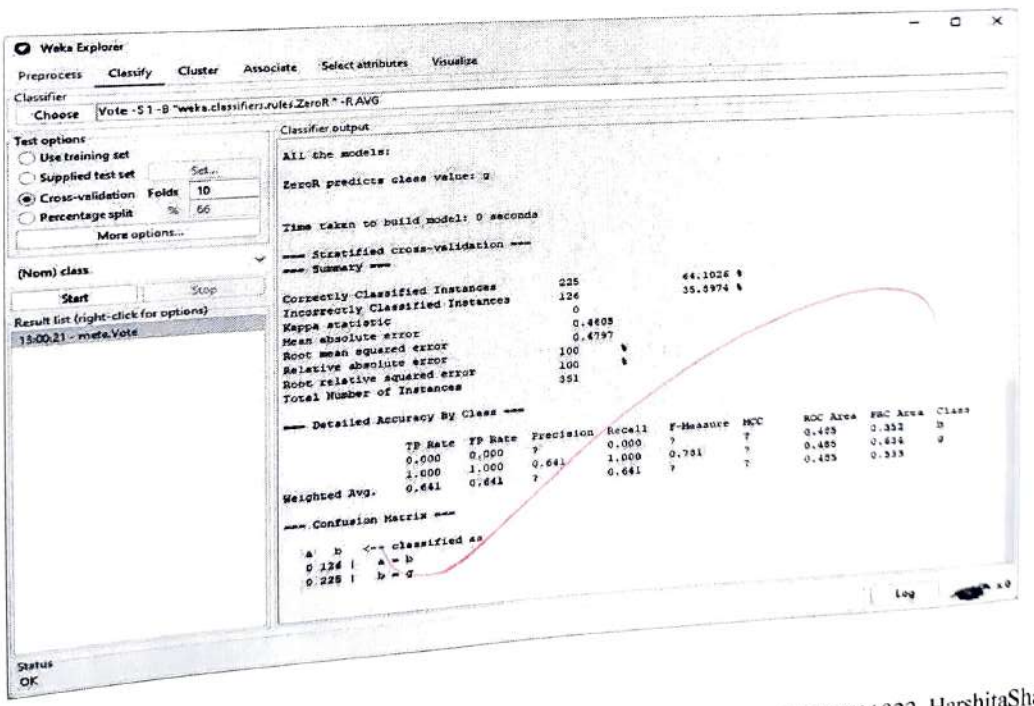
Step 2:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Apply
Stop
Current relation
Relation: ionosphere
Instances: 351
Attributes: 35
Sum of weights: 351
Attributes
All
None
Invert
Pattern
No.
Name
1 a01
2 a02
3 a03
4 a04
5 a05
6 a06
7 a07
8 a08
9 a09
10 a10
11 a11
12 a12
13 a13
14 a14
15 a15
16 a16
17 a17
18 a18
19 a19
Remove
Selected attribute
Name: a01
Missing: 0 (0%)
Distinct: 2
Type: Numeric
Unique: 0 (0%)
Statistic
Value
Minimum
0
Maximum
1
Mean
0.892
StdDev
0.311
Class: class (Nom)
Visualize All
313
Log
x 0
Status
OK

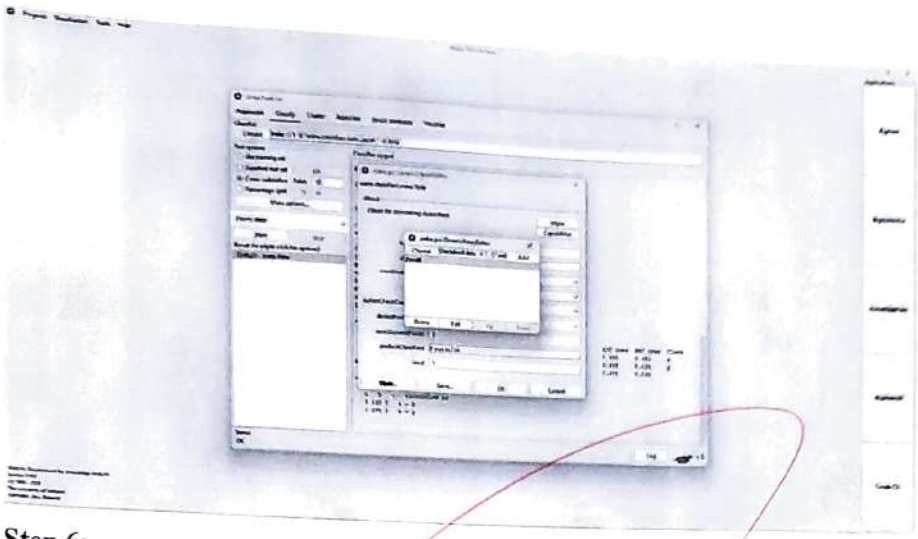
Step 3:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
weka
classifiers
bayes
functions
lazy
meta
AdaBoostM1
AdditiveRegression
AttributeSelectedClassifier
Bagging
ClassificationViaRegression
CostSensitiveClassifier
CVParameterSelection
FilteredClassifier
IterativeClassifierOptimizer
LogitBoost
MultiClassClassifier
MultiClassClassifierUpdateable
MultiScheme
RandomCommittee
RandomizableFilteredClassifier
RandomSubSpace
RegressionByDiscretization
Stacking
Vote
WeightedInstancesHandlerWrapper
misc
rules
Close
Status
OK
Log
x 0

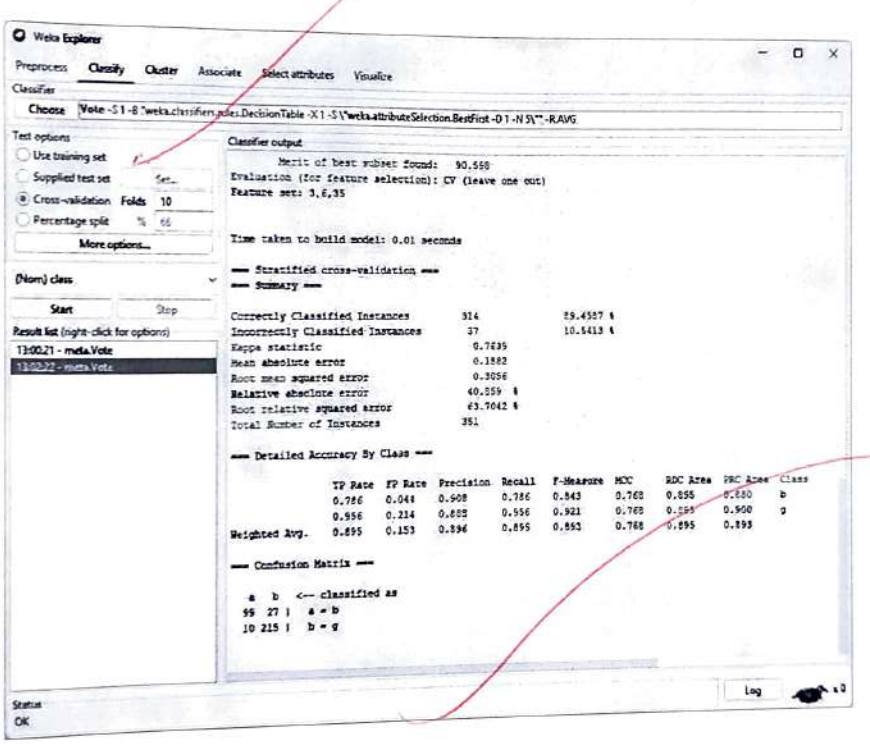
Step 4:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
Vote -S 1 -B "weka.classifiers.rules.ZeroR " -R AVG
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) class
Start
Stop
Result list (right-click for options)
13:00:21 - meta.Vote
Classifier output
All the models:
ZeroR predicts class value: 0
Time taken to build model: 0 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances 225 64.1026 %
Incorrectly Classified Instances 126 35.8974 %
Kappa statistic 0
Mean absolute error 0.4609
Root mean squared error 0.4797
Relative absolute error 100 %
Root relative squared error 100 %
Total Number of Instances 351
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
0.000 0.000 ? 0.000 ? ? 0.429 0.352 b
1.000 1.000 0.641 1.000 0.781 ? 0.485 0.434 a
Weighted Avg. 0.641 0.641 ? 0.641 ? ? 0.455 0.393
=== Confusion Matrix ===
a b <-- classified as
0 126 | a = b
0 225 | b = a
Status
OK
Log
x 0

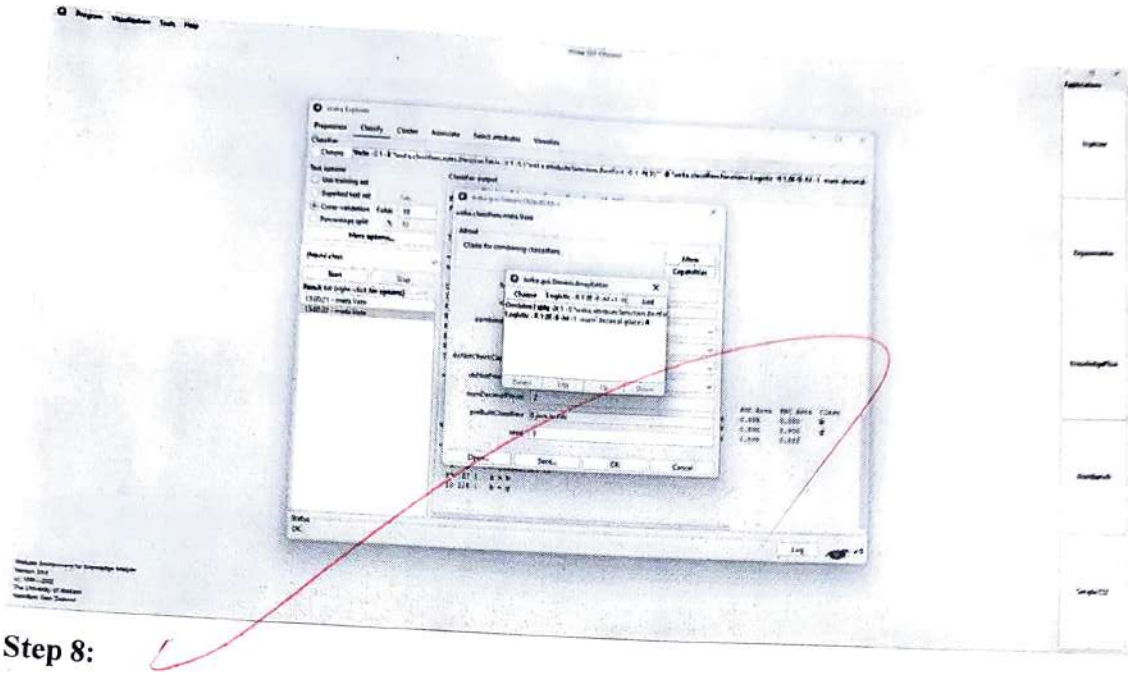
Step 5:



Step 6:


Weka Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Classifier
Choose Vote -S 1 -B "weka.classifiers.rules.DecisionTable -X 1 -S \"weka.attributeSelection.BestFirst -D 1 -N 5\"" -R AVG
Test options
Use training set
Supplied test set Set...
Cross-validation Folds 10
Percentage split % 66
More options...
(Nom) class
Start Stop
Result list (right-click for options)
13:00:21 - meta.Vote
13:02:22 - meta.Vote
Classifier output
Merit of best subset found: 90.568
Evaluation (for feature selection): CV (leave one out)
Feature set: 3,6,35
Time taken to build model: 0.01 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances 314 89.4587 %
Incorrectly Classified Instances 37 10.5413 %
Kappa statistic 0.7635
Mean absolute error 0.1882
Root mean squared error 0.3056
Relative absolute error 40.359 %
Root relative squared error 63.7042 %
Total Number of Instances 351
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
0.786 0.044 0.908 0.786 0.843 0.768 0.895 0.880 b
0.956 0.214 0.885 0.956 0.921 0.768 0.895 0.900 g
Weighted Avg. 0.895 0.153 0.896 0.895 0.893 0.768 0.895 0.893
=== Confusion Matrix ===
a b <-- classified as
99 27 | a = b
10 215 | b = g
Status
OK
Log
x 0

Step 7:



Step 8:

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Classifier

Choose: Vote -S 1 -B "weka.classifiers.rules.DecisionTable -X 1 -S \"weka.attributeSelection.BestFirst -D 1 -N 5\"" -B "weka.classifiers.functions.Logistic -R 1.0E-8 -M -1 -num-decimal-

Test options

- Use training set
- Supplied test set — Set...
- Cross-validation — Folds: 10
- Percentage split — %: 66

More options...

(Nom) class

Start | Stop

Result list (right-click for options)

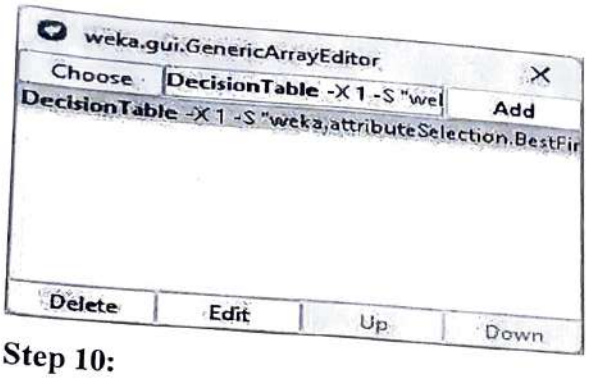
- 13:00:21 - meta.Vote
- 13:02:22 - meta.Vote
- 13:04:08 - meta.Vote

Classifier output

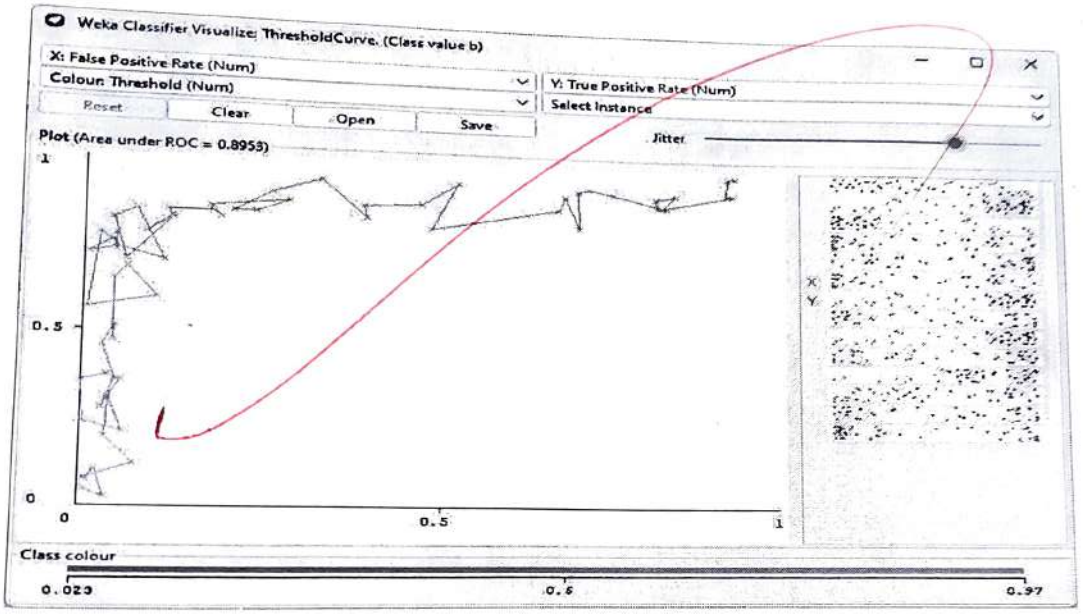
```
a33                0.8734
a34               56.1574


Time taken to build model: 0.06 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         317               90.3134 %
Incorrectly Classified Instances        34                9.6866 %
Kappa statistic                          0.7811
Mean absolute error                      0.1582
Root mean squared error                  0.2636
Relative absolute error                 34.3592 %
Root relative squared error             54.9523 %
Total Number of Instances              351

=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall  F-Measure   MCC      ROC Area  PRC Area  Class
                 0.778     0.027     0.942     0.778     0.852      0.789    0.947     0.946     b
                 0.973     0.222     0.887     0.973     0.928      0.789    0.947     0.945     g
Weighted Avg.    0.903     0.152     0.907     0.903     0.901      0.789    0.947     0.945

=== Confusion Matrix ===

  a   b   <-- classified as
 98  28 |   a = b
  6 219 |   b = g
```

Status: OK — Log — x 0

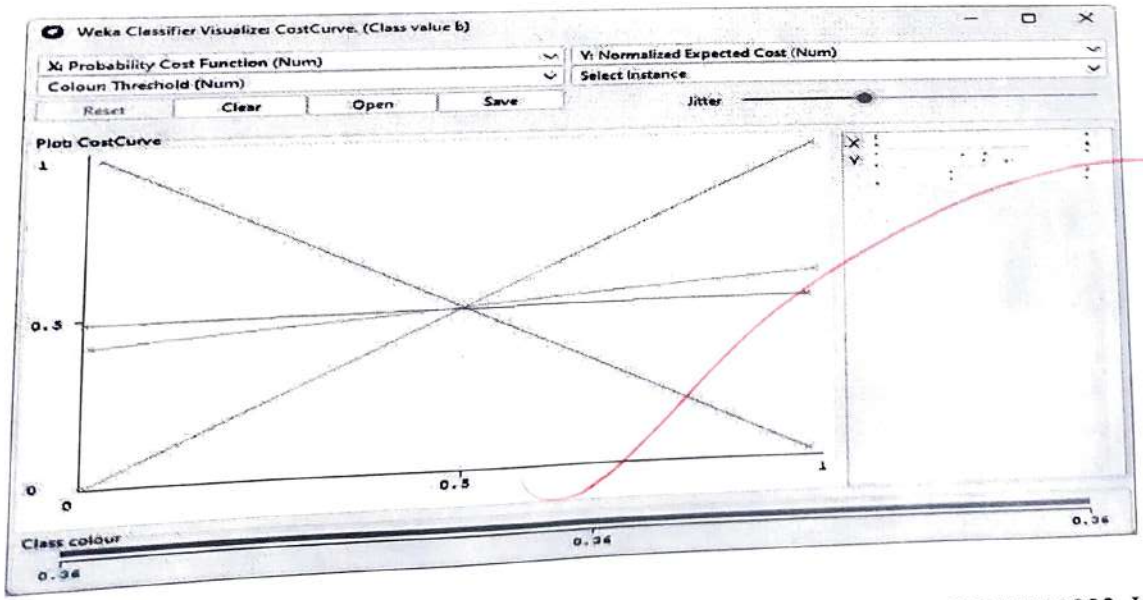
Step 9:



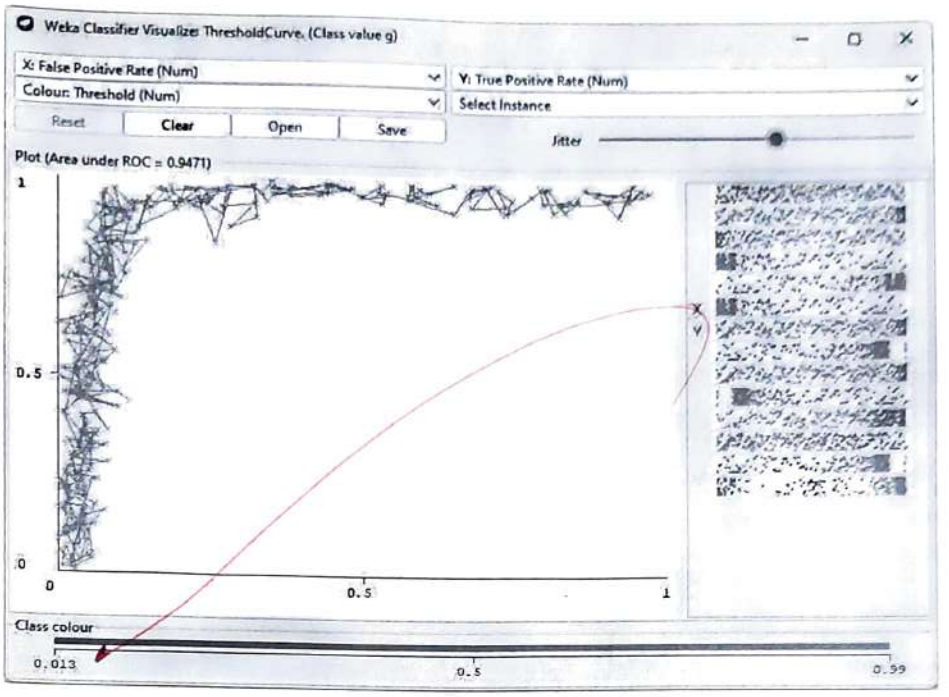
Step 10:



Step 11:



Step 12:



Learning Outcomes:

→ learnt how the voting ensemble method improves accuracy and robustness over single classifiers and gained practical experiancy in applying and comparing ensemble method with - in same class or weka.

Validated

Sign

Experiment 10

Objective: Study of DBMINER tool.

Theory:

DBMiner is an advanced data mining system developed to address the emerging requirements for meaningful knowledge discovery in large relational databases. Since organizations have begun to accumulate enormous amounts of data, traditionally databases designed mainly for storage and query operations are not enough to discover vital insights. DBMiner bridges this gap with a user-friendly yet powerful tool that is capable of multi-dimensional analysis with interactive knowledge discovery. It's primary goal is to enable multi-abstraction-level data exploration allowing organisations to carry out completely transparent analysis of complex datasets. The DBMiner architecture consists of three cores components :-

1) a relational database that serves as the source of information.
2) concept hierarchies so that the user can generalize from one level.
3) the knowledge discovery modules, each of which specializes in specific mining tasks.

→ These modules form the backbone functionality of DBMiner, which involves characterization, comparison, classification, association, prediction and clustering.

Advanced data mining techniques boost the capabilities of DBMiner significantly and make it good for intricate analytical tasks. Multiple-level characterization make use of generalization for showing data at different levels of detail and gives users broad overviews before going into details.

The discriminant rules are yet another technique in DBMiner; it finds different groups of data with contrasting features for analysis and determines which features differentiate one group from another, hence revealing the distinguishing features. DBMiner further supports multiple level association rules not only with a single attribute but also among various attributes. for example:- Intra-attribute associations in a retail database may point to trends in product categories, whereas, inter-attribute associations may provide you insight in terms of relating purchases with the demographics of customers.

Classification and prediction are two main functionalities in DBMiner tool, more specifically, the classifier module is comprised of combinations between attribute-oriented and decision-tree induction methods for classifying the data accordingly, to historical trends. This module is very useful for customer profiling, it can predict which customer segment a new person might belong to according to their attributes.

Another very useful module is prediction, it uses patterns found in the data to predict the future events. for example; how much sales will be during the quarter or which of customers at risk will churn. clustering groups similar algorithm. DBMiner supports both categorical and numerical data and has the possibility of multi-level clustering, which allows the groupings to be regarded at various levels of detail.

Looking ahead, DBMiner is being further developed in terms of performance to facilitate easier management of increasingly large datasets, and expanded capabilities needs to be made with specialized databases handling spatial, temporal, or multimedia data. On further improving its utility with data-intensive applications, the architecture of DBMiner tool has been designed to support parallel processing with substantially faster computation involving distributed environments and reduced processing time as well as increased scalabilities. DBMiner is one of the versatile and robust tool for data mining pertaining to industrial requirements for data-driven insights.

It will integrate a relational database, concept hierarchies, and knowledge

discovery modules to later to the interactive exploration of data at different abstraction levels. DBminer provides actionable patterns and trends by using specialized mining techniques such as → characterization
→ association
→ classification
- and clustering and the combinations of DMQL (Data Mining Query language) provides more access and usability for users accustomed to SQL.

DBminer tool epitomizes a robust solution in which organizations can capitalize on their assets relating to data so they can produce strategies, and strategic decision, thereby staying abreast with the rapidly shifting technological landscape in data science and analytics.

Completed 8/11/24

E-commerce Customer Retention: A Data-Driven Analysis

Presented by
Harshita Sharma
21117711922
AIDS-B
G-2

Index

Introduction

- In today's competitive landscape, understanding customer behavior is crucial for creating targeted marketing strategies.
- This project leverages data mining techniques, specifically using Weka, to uncover hidden patterns in customer data.
- By analyzing factors like age, payment methods, and product categories, this study aims to segment customers and predict purchasing behaviors, providing actionable insights to drive strategic decisions.
- This project explores customer purchasing behavior using data mining techniques within Weka, a powerful machine-learning tool.
- The goal is to analyze customer attributes, segment users by payment preferences, and develop predictive models that help businesses understand purchasing patterns.

Problem Statement

Customer behavior and preferences can be complex and influenced by various factors, such as age, payment methods, and product type. This project seeks to address the challenge of identifying meaningful patterns in customer data to improve business strategies. Specifically, We aim to cluster customers based on their payment methods and create classifiers to predict customer purchasing trends. These insights will help businesses make data-driven decisions for customer engagement, personalized marketing, and inventory management.

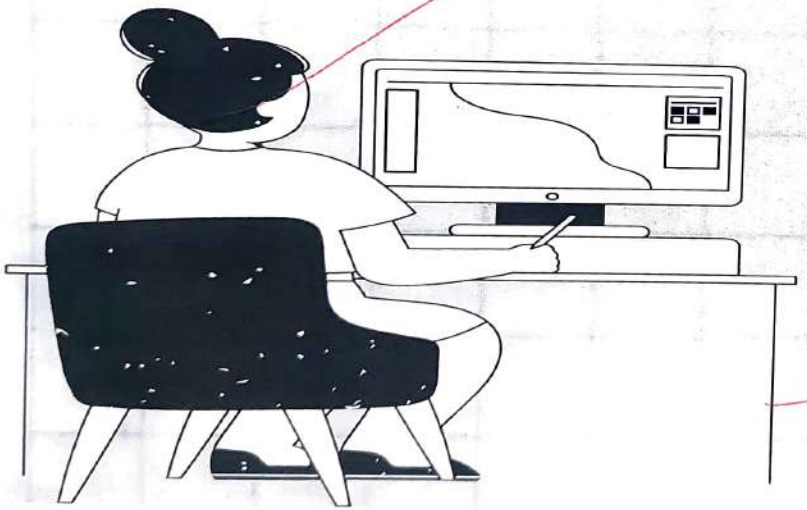
01. **Core Challenge:** Customer purchasing decisions are influenced by a variety of factors, including demographics, payment methods, and product categories. Understanding these patterns is essential for effective customer segmentation and targeted marketing.

02. **Scope of Study:** This project focuses on segmenting customers based on payment methods and creating predictive classifiers for purchasing trends. The data is processed, visualized, and analyzed using Weka, where various clustering and classification techniques provide insights into customer behavior.

03. **Relevance of Study:** Analyzing customer purchasing patterns allows businesses to develop strategies for customer retention, personalized marketing, and sales forecasting. This study bridges the gap between raw customer data and actionable business insights.

Methodology

This "E-commerce Customer Behavior and Purchase Dataset" is taken from the Kaggle, to simulate a real online shopping environment. All the detailed aspects of the customer's actions and purchase patterns within an e-commerce platform are captured. Such a dataset is suitable for data analysis and predictive modeling tasks in e-commerce, like predicting customer churn, market basket analysis, recommendation systems, and trend exploration.



❖ Pre-Processing stage

Original Dataset

- **Attributes: 13 attributes,** including Customer ID, Purchase Date, Product Category, Product Price, Quantity, Total Purchase Amount, Payment Method, Customer Age, Returns, Customer Name, Age, Gender, & Churn.
- **Instances: 250,000 rows.**
- **Selected Attribute**: The Customer Age attribute with values ranging from **18 to 70**.
- **mean age** (43.941)
- **standard deviation:** (15.35)

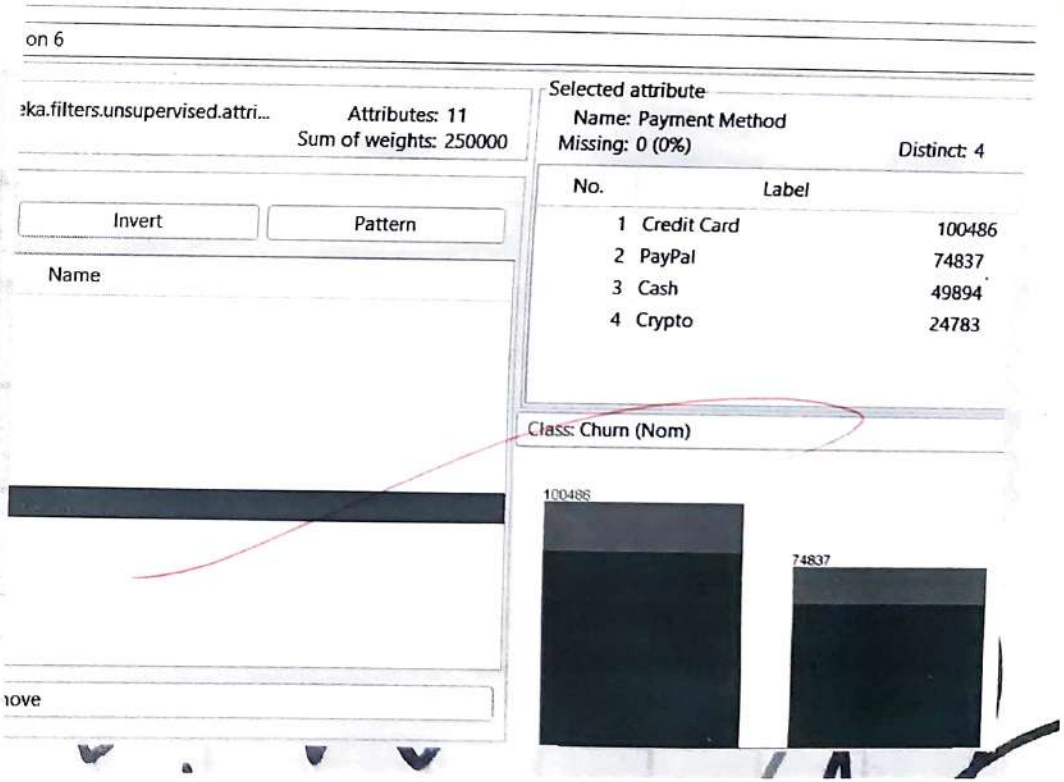
After Preprocessing and Noise Removal:

- **Normalization**: The Product Price attribute has been normalized, with a new range from **0 to 1**.
- The mean product price value after normalization is approximately **0.499**, with a standard deviation of **0.289**.
- **Attributes**: processed dataset now has **11 attributes** instead of the original **13**, **Instances**: The number of rows remains the same at **250,000**.

Discretization

Analysis

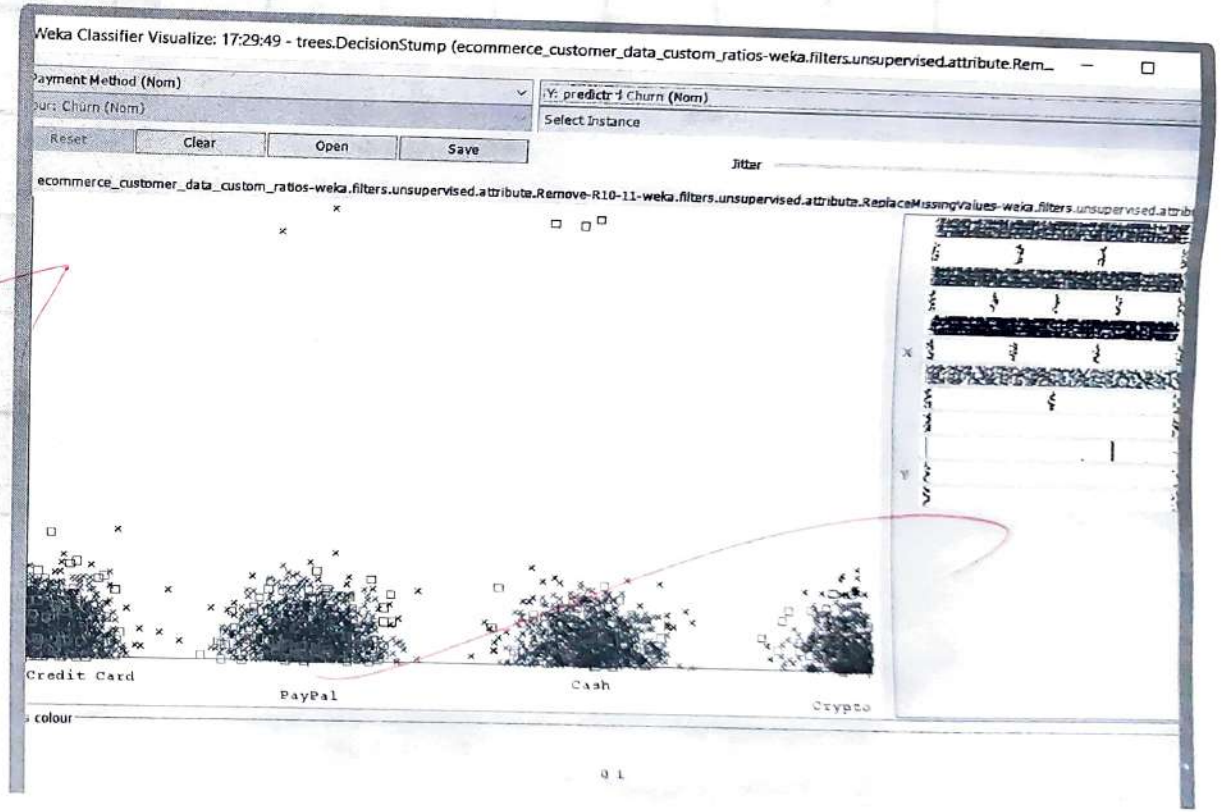
- **Filter Applied**: Discretize.
- **Selected Attribute:** Payment Method (nominal) with 4 categories:
  - **Credit Card**: 100,486
  - **PayPal**: 74,837
  - **Cash**: 49,894
  - **Crypto**: 24,783
- Churn Distribution:
  - Bar chart shows churn (red) vs. retention (blue) within each payment method.
  - **Credit Card** is the most common payment method.
- **Insight**: Differences in churn across payment methods suggest areas for targeted retention strategies.



Decision Tree stamp classifier Algorithm

- The plot shows the distribution of customers across different payment methods and their predicted churn status. The color coding helps visualize the accuracy of the model's predictions.

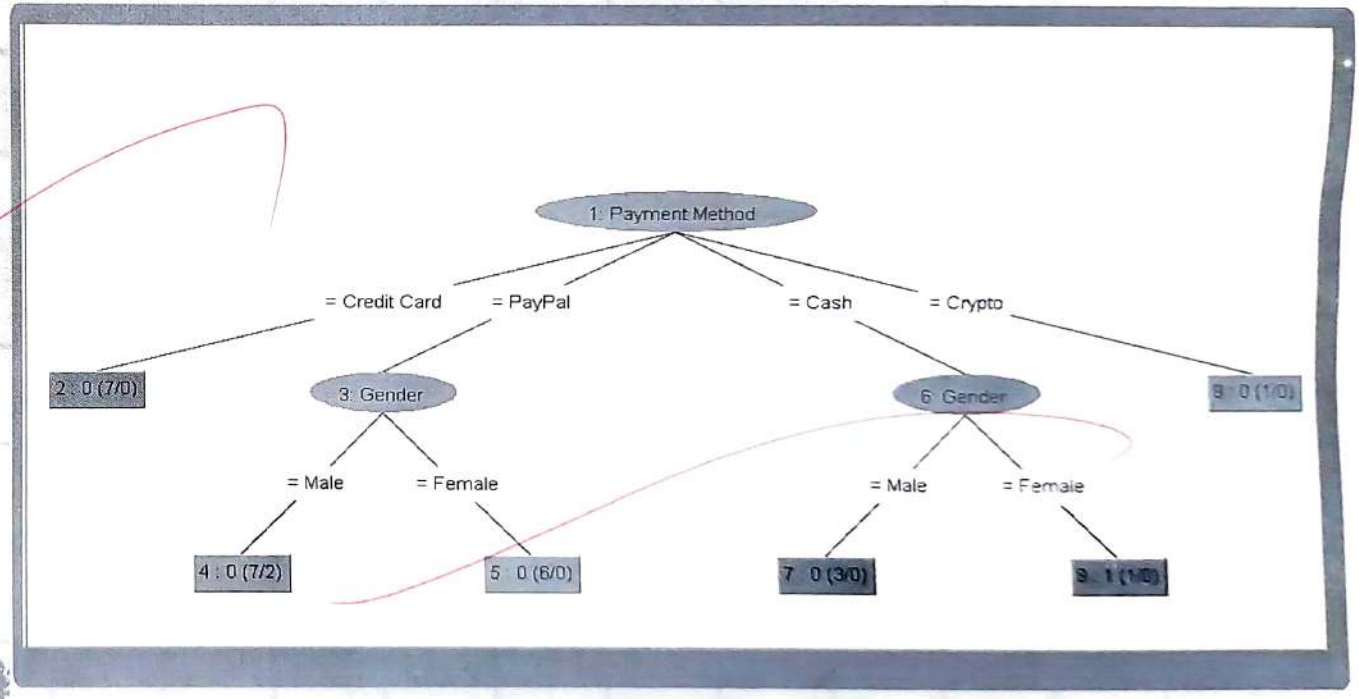
- The algorithm used to generate this decision tree is likely a **Decision Stump**, which is a simple decision tree with only one internal node and two leaf nodes. It is a basic form of a decision tree that can be used for classification tasks like this one.



Decision Tree

- Decision trees are a supervised learning algorithm used for both classification and regression tasks. They work by creating a tree-like model of decisions and their possible consequences.

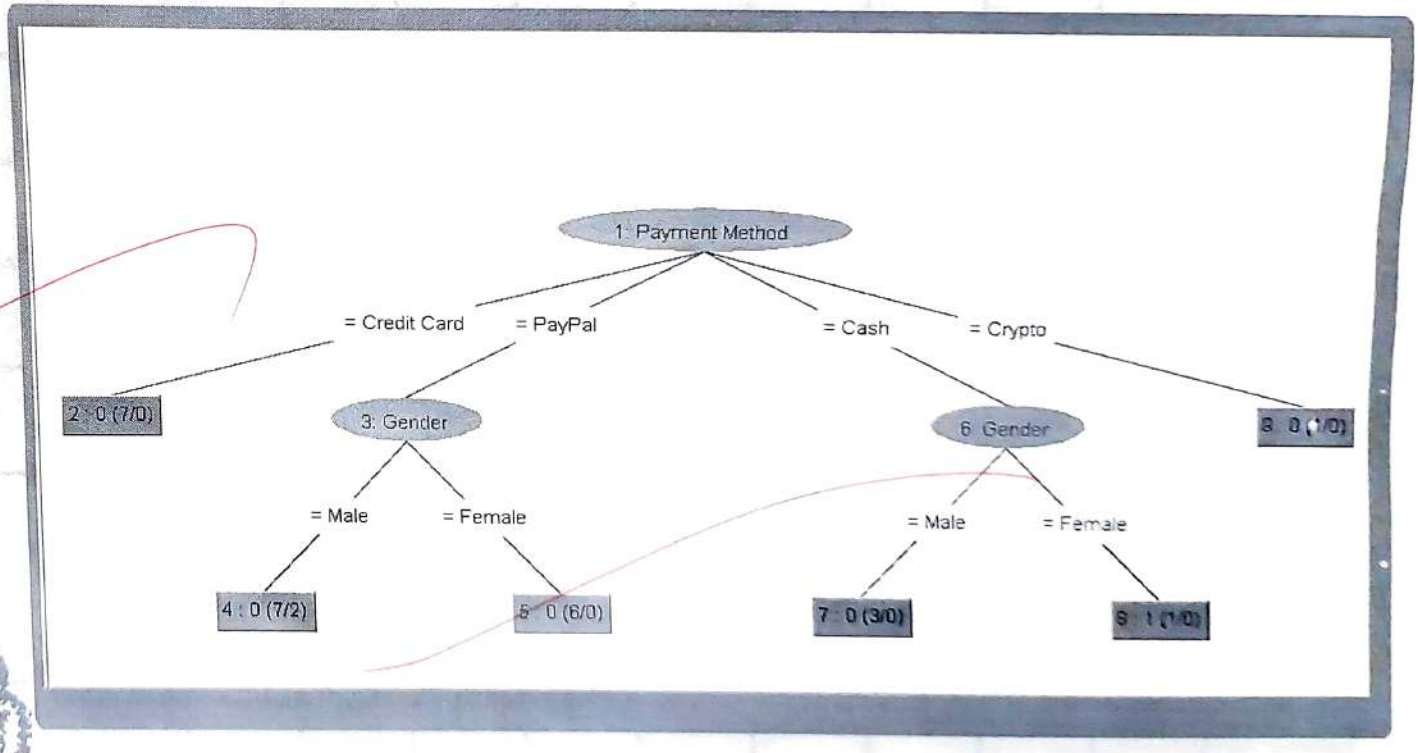
- **Root Node:** Splits data based on "Payment Method".
- **Child Nodes:** Further split based on "Gender".
- **Leaf Nodes:** Represent final prediction (churn or not churn) with associated instance counts



Decision Tree

- Decision trees are a supervised learning algorithm used for both classification and regression tasks. They work by creating a tree-like model of decisions and their possible consequences.

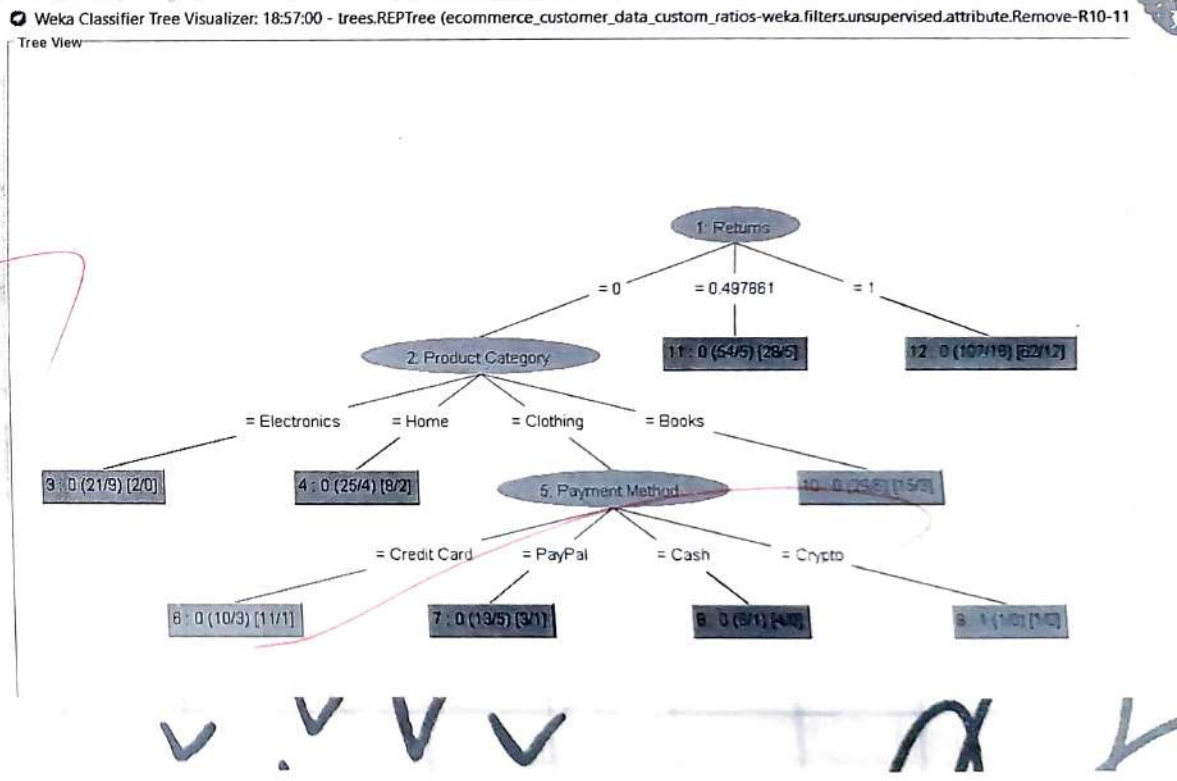
- **Root Node:** Splits data based on "Payment Method".
- **Child Nodes:** Further split based on "Gender".
- **Leaf Nodes:** Represent final prediction (churn or not churn) with associated instance counts



REPTREE

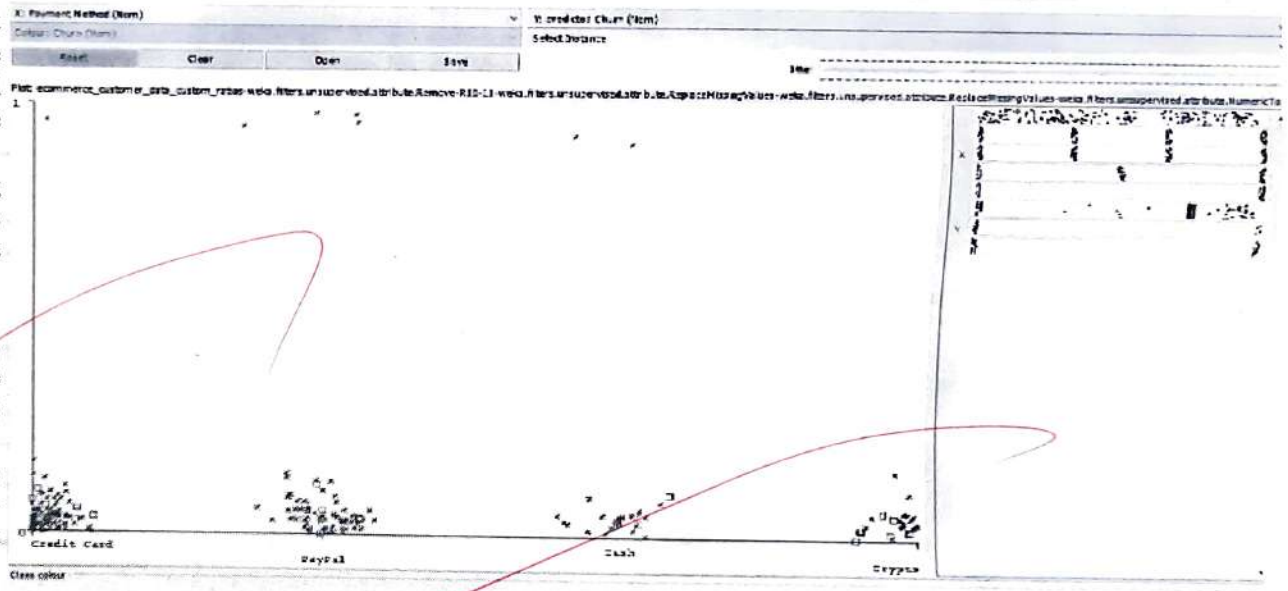
Analysis

- **REPTree (Reduced Error Pruning Tree)** is a decision tree algorithm used in data mining for classification and regression tasks. It's known for its efficiency and simplicity.
- This tree is likely used to predict customer churn based on various factors like product returns, product category, and payment method.



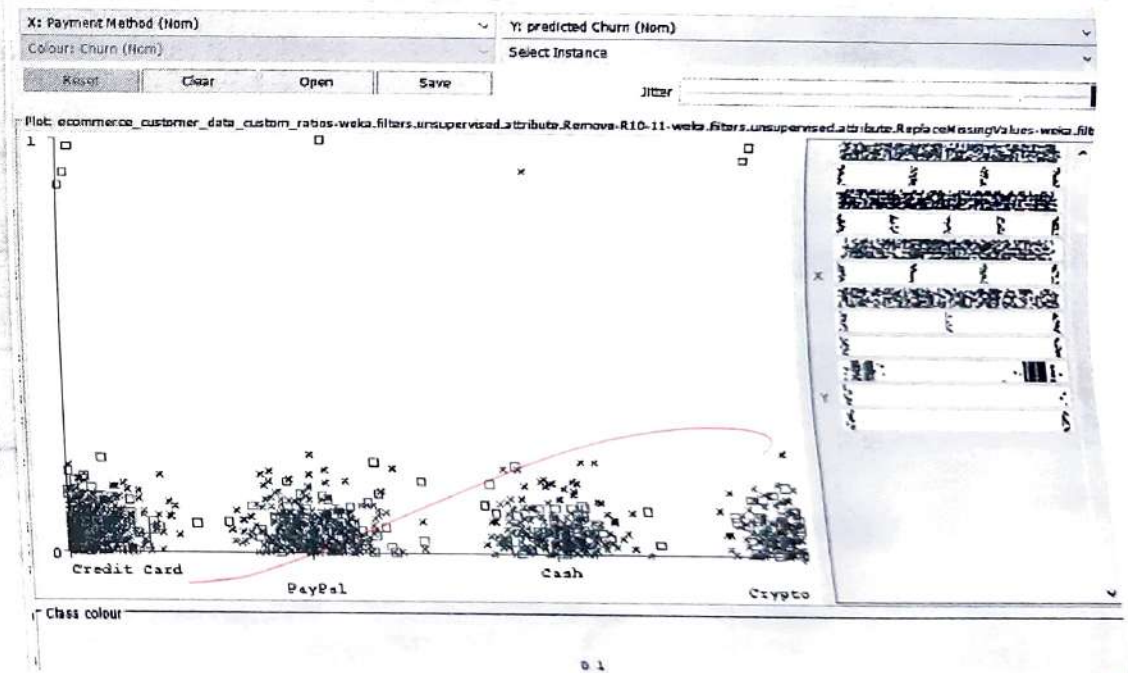
Bagging

- **Explanation:**
- The plot displays the performance improvement achieved by bagging, which trains multiple models on bootstrapped samples of the data and combines them to form a more stable, accurate classifier.
- Bagging reduces variance by averaging predictions from several models, which makes it particularly effective when individual models are prone to high variance.
- **Algorithm Insight:**
- By averaging results from diverse models, bagging creates a more reliable and accurate predictor. It is especially useful in datasets where individual classifiers may overfit.
- Bagging works well with algorithms like decision trees, which are inherently high-variance models. It increases stability without significantly increasing bias.



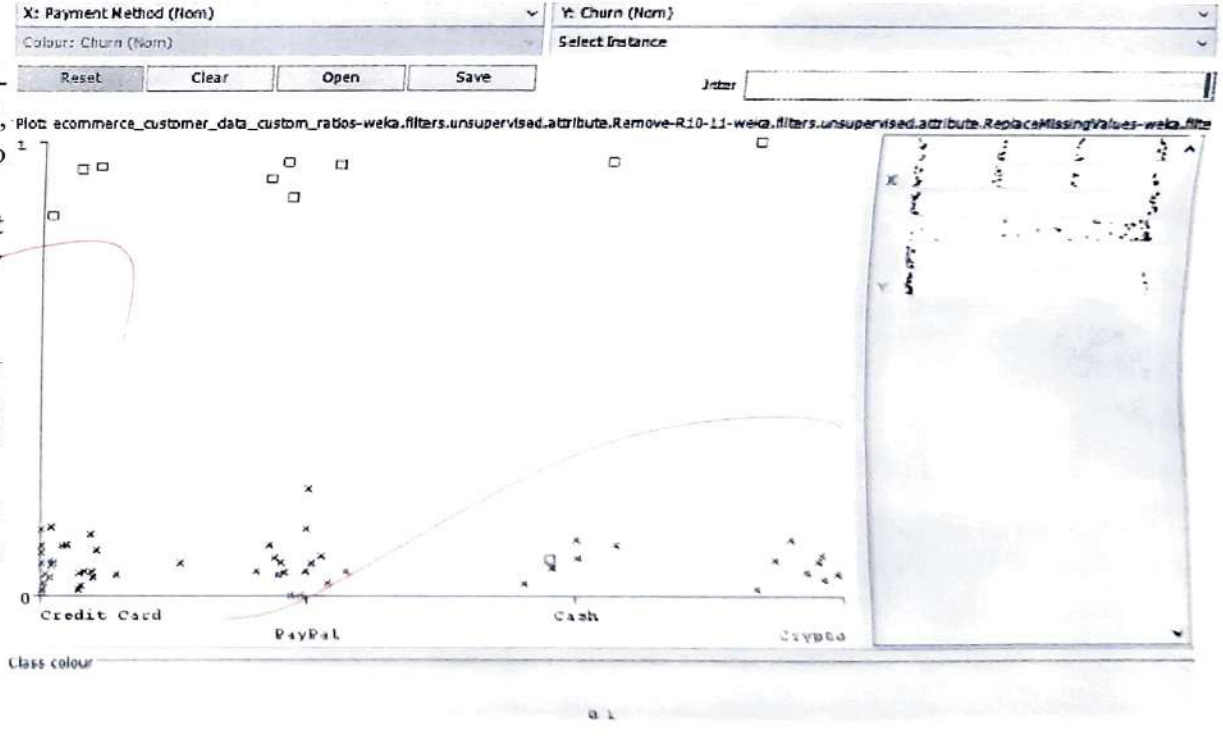
AdaBoostM1 Algorithm

- **Explanation:**
- The plot illustrates the error rate of the AdaBoostM1 algorithm over several boosting rounds, showing how misclassified samples are given more focus in each iteration.
- AdaBoostM1 is an ensemble method that improves accuracy by iteratively adjusting the weights of instances, especially those that are misclassified, to "boost" the predictive power of the overall model.
- **Algorithm Insight:**
- This model is effective for handling difficult-to-classify instances by combining weak learners into a strong predictive model. It is particularly suited for cases where improving accuracy on challenging samples is crucial.
- AdaBoostM1 is **sensitive to noisy data**, as misclassified noisy instances may receive higher weights, which can sometimes lead to overfitting.



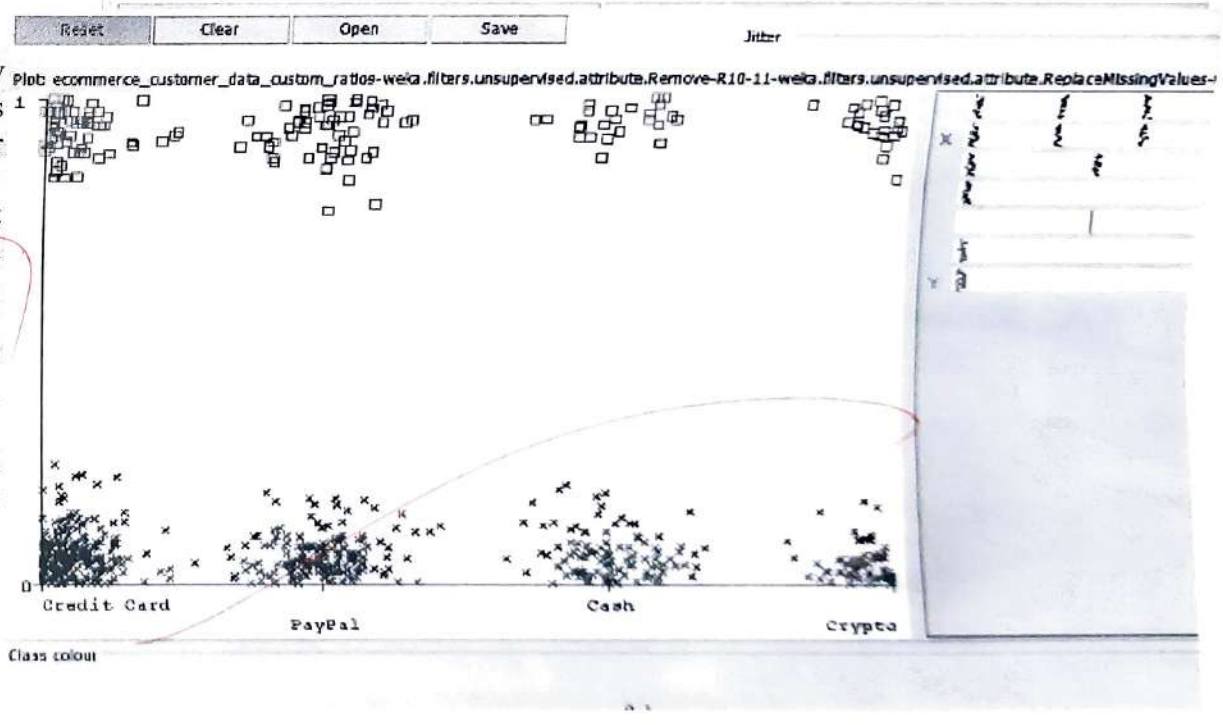
Meta-Vote Ensemble Algorithm

- **Explanation:**
- The plot shows the classification accuracy of the Meta-Vote ensemble, which combines J48, Random Forest, and Naïve Bayes classifiers by majority voting to enhance predictive performance.
- Meta-Vote ensembles leverage the strengths of different algorithms to achieve a higher overall accuracy by aggregating their individual predictions.
- **Algorithm Insight:**
- This approach benefits from the complementary strengths of various models. While J48 provides interpretability, Random Forest offers robustness, and Naïve Bayes adds probabilistic reasoning.
- Meta-Vote ensembles are particularly useful when individual models have diverse strengths that, when combined, provide a balanced prediction.



Simple Logistic Algorithm

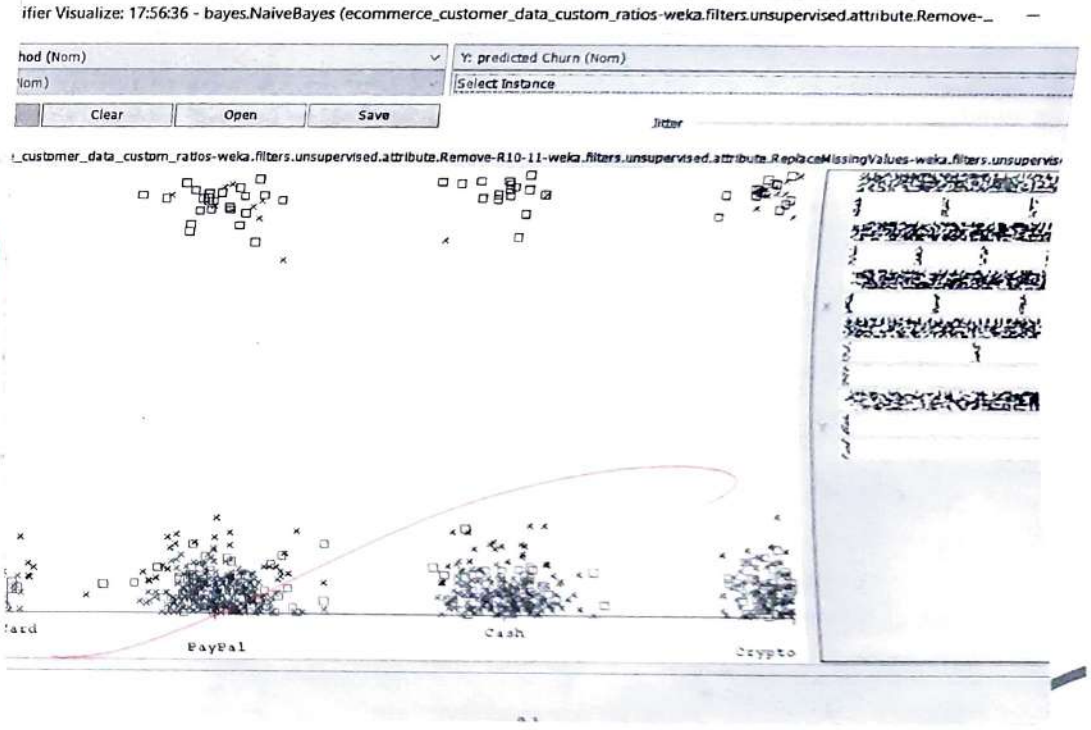
- **Explanation:**
- The plot illustrates the decision boundaries created by the Simple Logistic algorithm, which predicts probabilities for each class and is ideal for linear classification.
- Simple Logistic is a regression-based classifier that provides probabilities, offering a straightforward interpretation for binary or multi-class classification.
- **Algorithm Insight:**
- It is best suited for problems with linearly separable classes and offers a probabilistic output, which is useful for decision-making.
- While limited to linear relationships, it is a simple and interpretable model, effective for datasets with binary or few categories.



Naïve Bayes

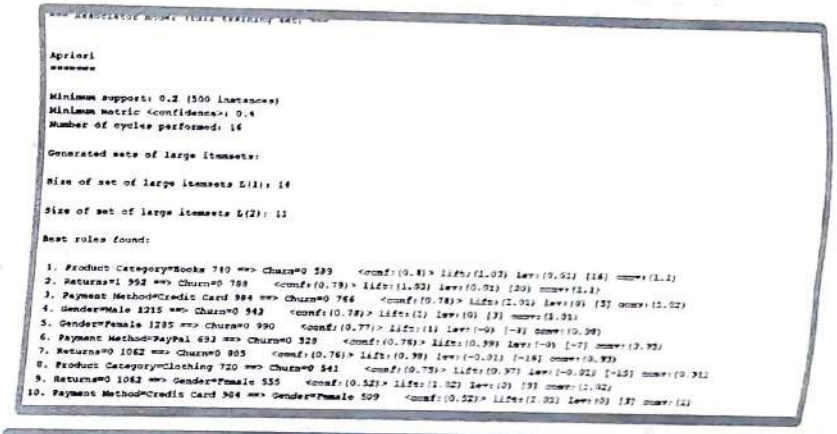
Analysis

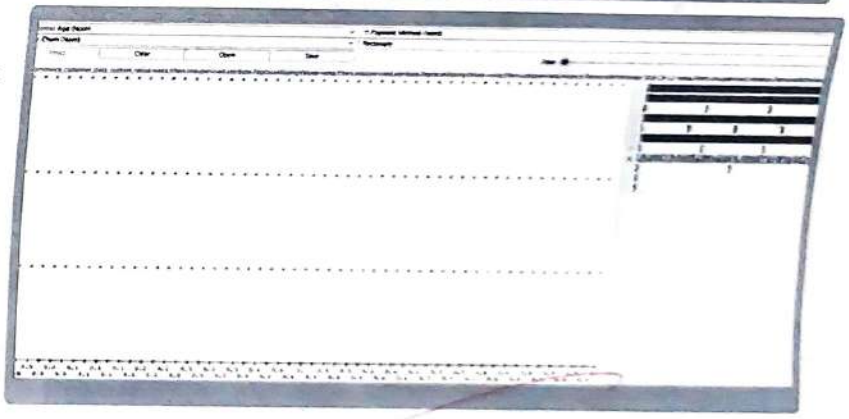
- - Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem.
- It assumes that the features are conditionally independent,
- Meaning that the occurrence of one feature does not influence the probability of another feature given the class.



Apriori Algorithm

- The first image shows the results of an Apriori association rule mining algorithm applied to the e-commerce dataset. This analysis reveals interesting patterns in customer behavior and product relationships.
- The second image presents a scatter plot visualization of customer data, focusing on customer age and payment method.





Insights

- **Customer Churn:** Certain customer demographics, product returns, and payment methods are linked to higher churn rates.
- **Product Relationships:** Specific products are frequently purchased together, offering opportunities for cross-selling and upselling.
- **Age and Payment Method:** Different age groups prefer different payment methods.
- **Targeted Marketing:** Tailor marketing campaigns to specific age groups and customer segments.
- **Churn Prevention:** Implement targeted retention strategies for at-risk customers.
- **Product Recommendations:** Leverage association rules for personalized product recommendations.
- **Payment Optimization:** Offer a variety of secure payment options to cater to different preferences.

Result

Our analysis yielded meaningful insights into customer segments:

- Clustering Outcome: Customer groups based on payment methods provided clear visual insights. For example, Orange represents active use of payment methods by age group, while Blue indicates inactivity.
- Classification Performance: The ensemble models, particularly AdaBoostM1 and meta-vote ensemble, showed the best accuracy, with error plots highlighting performance across classifiers.
- Error Analysis: Decision stump, Naïve Bayes, and other models displayed varying error rates, helping identify areas for potential improvement.

Conclusion

- In conclusion, this project highlights the significant value of data mining for customer analysis, using Weka to facilitate in-depth pattern recognition and predictive modeling.
- Through the integration of clustering and classification techniques, we identified key behavioral trends that can be strategically leveraged to improve customer engagement and drive business growth.
- The K-means clustering revealed distinct customer segments based on payment methods, providing actionable insights into age-based preferences and purchasing behaviors.
- Additionally, the use of ensemble methods, particularly the meta-vote and AdaBoostM1, offered high accuracy in classification, underscoring the robustness of combining models for enhanced prediction.
- These insights equip businesses to make data-driven decisions, personalize marketing efforts, and optimize resources to better align with customer needs.

Future Work

- This project underscores how data mining is not only a tool for analysis but also a pathway to strategic advancements in customer satisfaction and retention.
- Future work can expand upon this foundation by incorporating larger datasets and refining models to achieve even greater predictive precision, thereby continually enhancing the customer experience.

THANK YOU