Name:- Sufiyan Khan
Class & Branch :- TE-A(AI & DS) , Sem-5 , A1
Roll No:- A12
Subject:- DWM LAB

LAB Manual
PART A

(PART A: TO BE REFFERED BY STUDENTS)

Experiment No.06

A.1 Aim:

Data pre processing using WEKA ,classification ,clustering ,association Rule mining on data set using WEKA .

A.2 Prerequisite:

Familiarity with the WEKA tool.

A.3 Outcome:

After successful completion of this experiment students will be able to
Use classification and clustering algorithms of data mining.

A.4 Theory:

Preprocessing:

Data have quality if they satisfy the requirements of the intended use. There are many factors comprising data quality, including accuracy, completeness, consistency, timeliness, believability, and interpretability.

Major Tasks in Data Preprocessing:

In this section, we look at the major steps involved in data preprocessing, namely, data cleaning, data integration, data reduction, and data transformation.

Data cleaning routines work to "clean" the data by filling in missing values, smoothing noisy data, identifying or removing outliers, and resolving inconsistencies.

Data reduction obtains a reduced representation of the data set that is much smaller in volume, yet produces the same (or almost the same) analytical results. Data reduction strategies include *dimensionality reduction* and *numerosity reduction*.

In ***dimensionality reduction,*** data encoding schemes are applied so as to obtain a reduced or "compressed" representation of the original data.

In ***numerosity reduction,*** the data are replaced by alternative, smaller representations using parametric models (e.g., *regression* or *log-linear models*) or nonparametric models (e.g., *histograms, clusters*, *sampling*, or *data aggregation*).

Data Cleaning:

Real-world data tend to be incomplete, noisy, and inconsistent. *Data cleaning* (or *data cleansing*) routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data.

Dealing with Missing Values:-

1. **Ignore the tuple**: This is usually done when the class label is missing (assuming the mining task involves classification). This method is not very effective, unless the tuple contains several attributes with missing values. It is especially poor when the percentage of missing values per attribute varies considerably. By ignoring the tuple, we do not make use of the remaining attributes' values in the tuple. Such data could have been useful to the task at hand.

2. **Fill in the missing value manually:** In general, this approach is time consuming and may not be feasible given a large data set with many missing values.

3. **Use a global constant to fill in the missing value**: Replace all missing attribute values by the same constant such as a label like *"Unknown"* or -∞. If missing values are replaced by, say, *"Unknown*," then the mining program may mistakenly think that they form an interesting concept, since they all have a value in common—that of *"Unknown."* Hence, although this method is simple, it is not foolproof.

4. **Use a measure of central tendency for the attribute (e.g., the mean or median) to fill in the missing value**: central tendency, indicate the "middle" value of a data distribution. For normal (symmetric) data distributions, the mean can be used, while skewed data distribution should employ the median..
5. **Use the attribute mean or median for all samples belonging to the same class as the given tuple**: For example, if classifying customers according to *credit risk*, we may replace the missing value with the mean *income* value for customers in the same credit risk category as that of the given tuple. If the data distribution for a given class is skewed, the median value is a better choice.

6. **Use the most probable value to fill in the missing value**: This may be determined with regression, inference-based tools using a Bayesian formalism, or decision tree induction.

Dealing with Noise:

Noise is a random error or variance in a measured variable

Binning: Binning methods smooth a sorted data value by consulting its "neigh bourhood," that is, the values around it. The sorted values are distributed into a number of "buckets," or bins. Because binning methods consult the neigh bourhood of values, they perform local smoothing. In smoothing by bin means, each value in a bin is replaced by the mean value of the bin. Similarly, smoothing by bin medians can be employed, in which each bin value is replaced by the bin median. In smoothing by bin boundaries, the minimum and maximum values in a given bin are identified as the bin boundaries. Each bin value is then replaced by the closest boundary value.

Regression: Data smoothing can also be done by regression, a technique that conforms data values to a function. *Linear regression* involves finding the "best" line to fit two attributes (or variables) so that one attribute can be used to predict

the other. *Multiple linear regression* is an extension of linear regression, where more than two attributes are involved and the data are fit to a multidimensional surface.

Outlier analysis: Outliers may be detected by clustering, for example, where similar values are organized into groups, or "clusters." Intuitively, values that fall outside of the set of clusters may be considered outliers.

PART B

(PART B: TO BE COMPLETED BY STUDENTS)

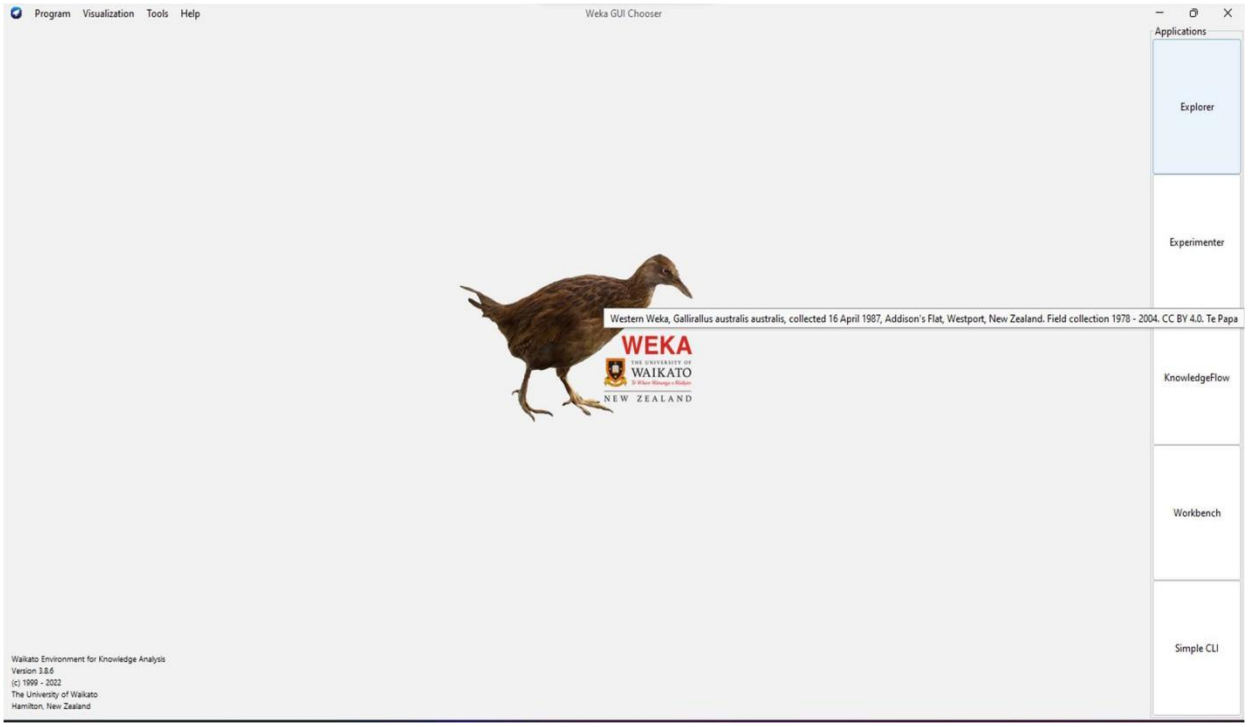
Roll. No:- A12	Name:- Sufiyan Khan
Class:- TE-A(AI & DS)	Batch:- A1
Date of Experiment:- 1 0 /10/23	Date of Submission:- 16/10/23
Grade:	

Weka - Preprocessing The Data:-

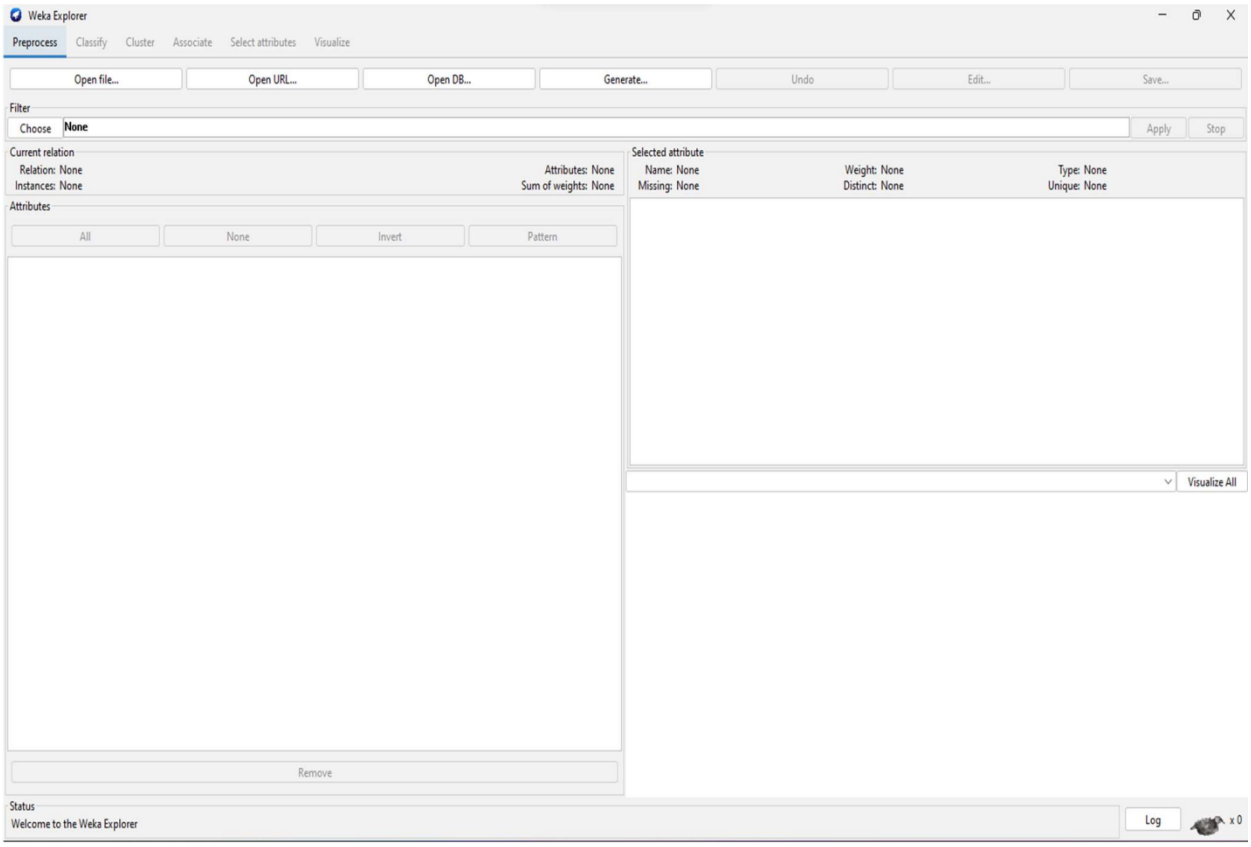
The data that is collected from the field contains many unwanted things that leads to wrong analysis. For example, the data may contain null fields, it may contain columns that are irrelevant to the current analysis, and so on. Thus, the data must be preprocessed to meet the requirements of the type of analysis you are seeking. This is the done in the preprocessing module.

To demonstrate the available features in preprocessing, we will use the Weather database that is provided in the installation.

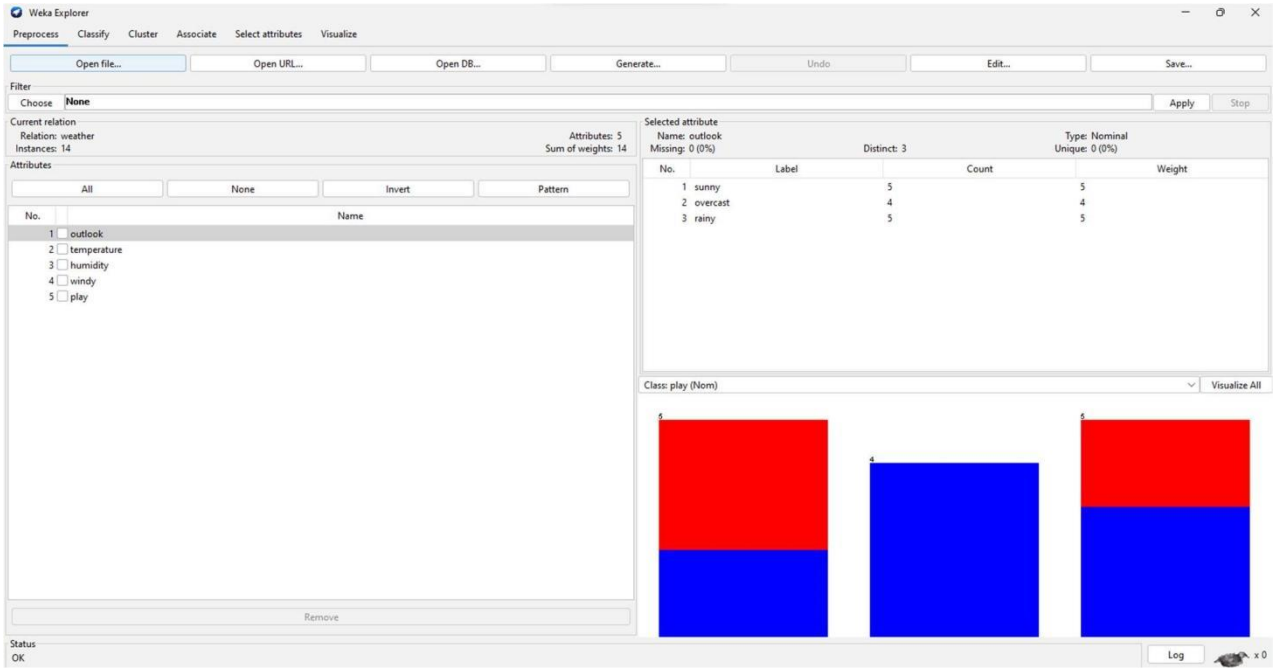
Weka Interface:-



Weka Explorer:-



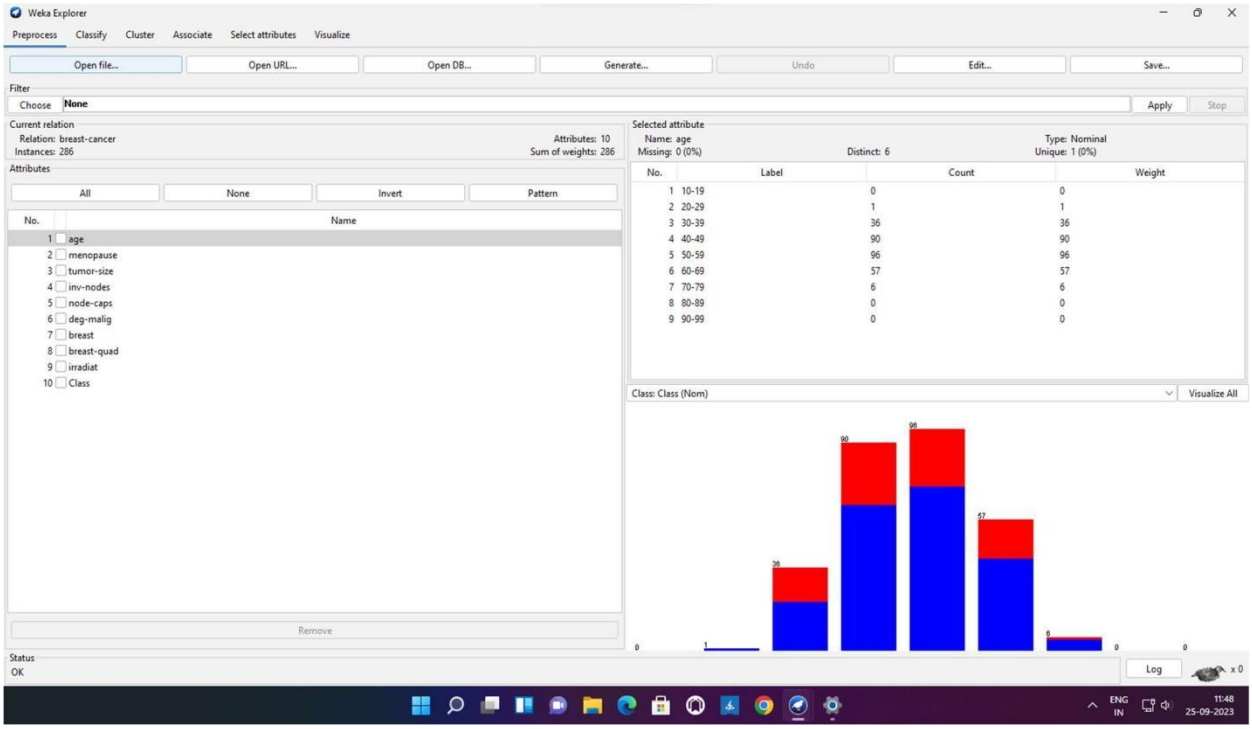
Weka Explorer – understanding weka explorer by opening dataset weather-numeric.arff:-



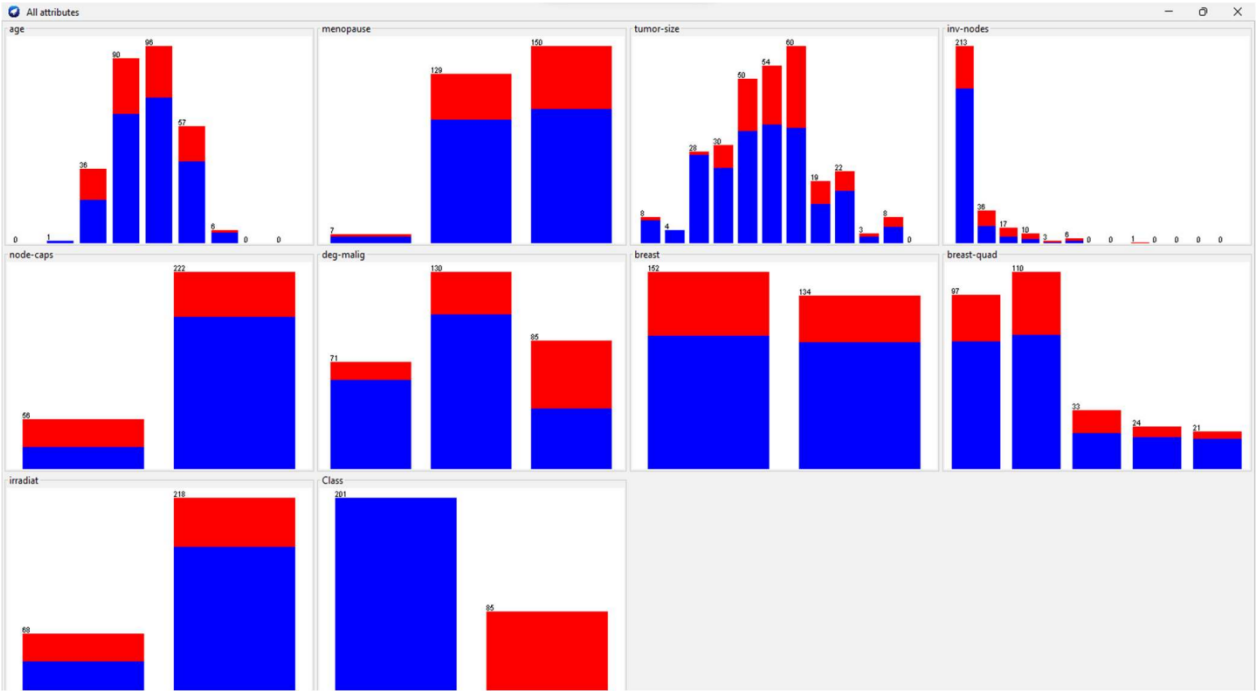
PREPROCESSING USING WEKA:-

Steps involved in preprocessing using Weka:-

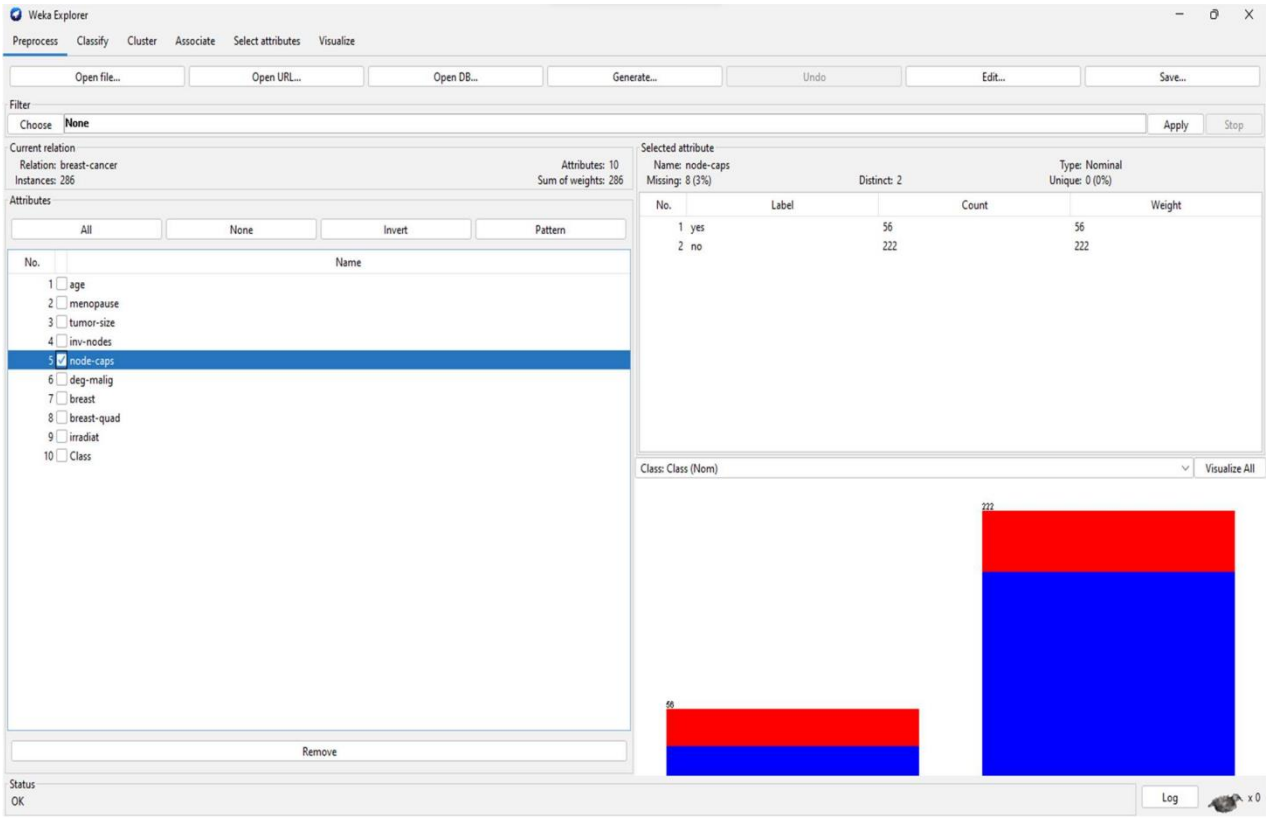
Step 1:- Select the dataset "breast-cancer" and open it:



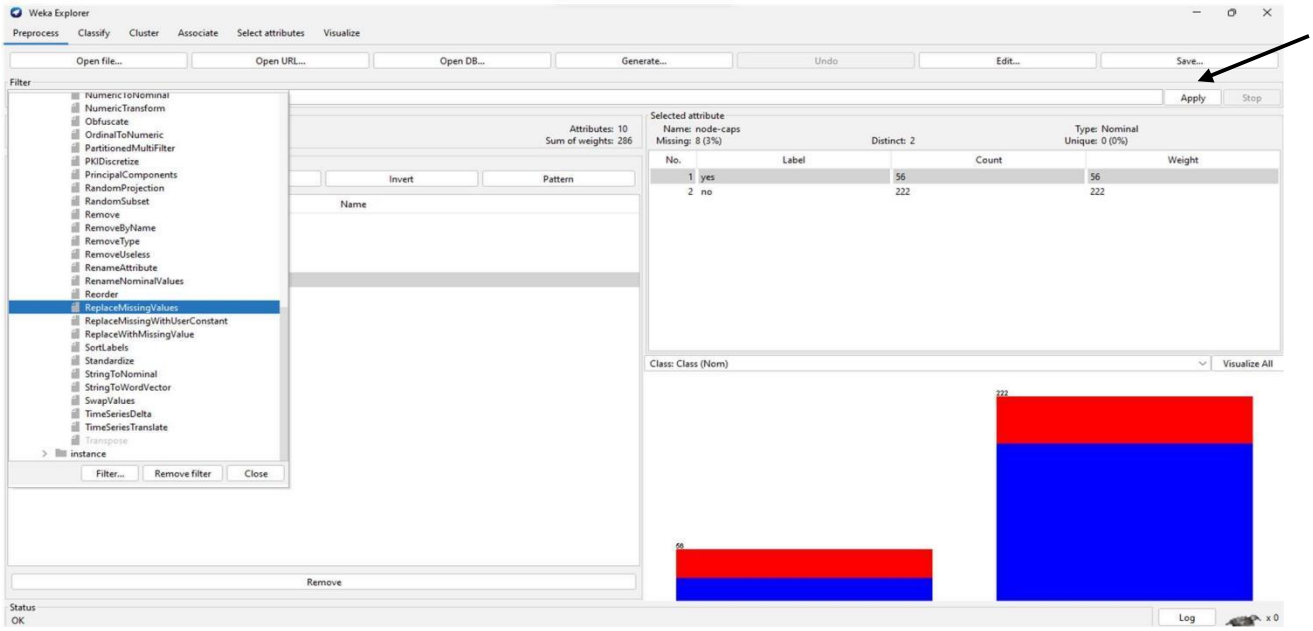
Step 2 :- Check every attribute for missing values and visualizing each attributes:-



Checking missing values for attribute node-caps:-

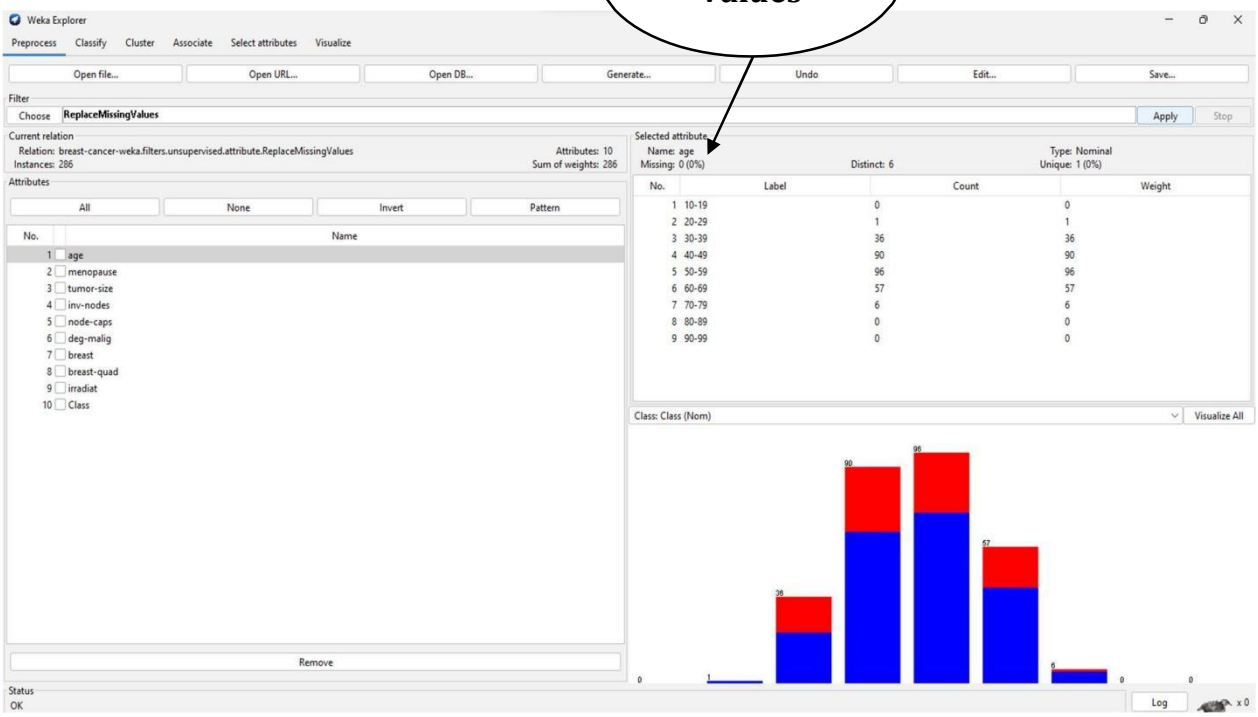


Step 3:- To fill the missing values with mean/median values , Select filter from Unsupervised -> attribute -> ReplaceMissingValues.



Select the attribute for which the filter is to be applied and click on apply.

Step 4:- After Applying we can see that all missing values are filled by mean/median:-

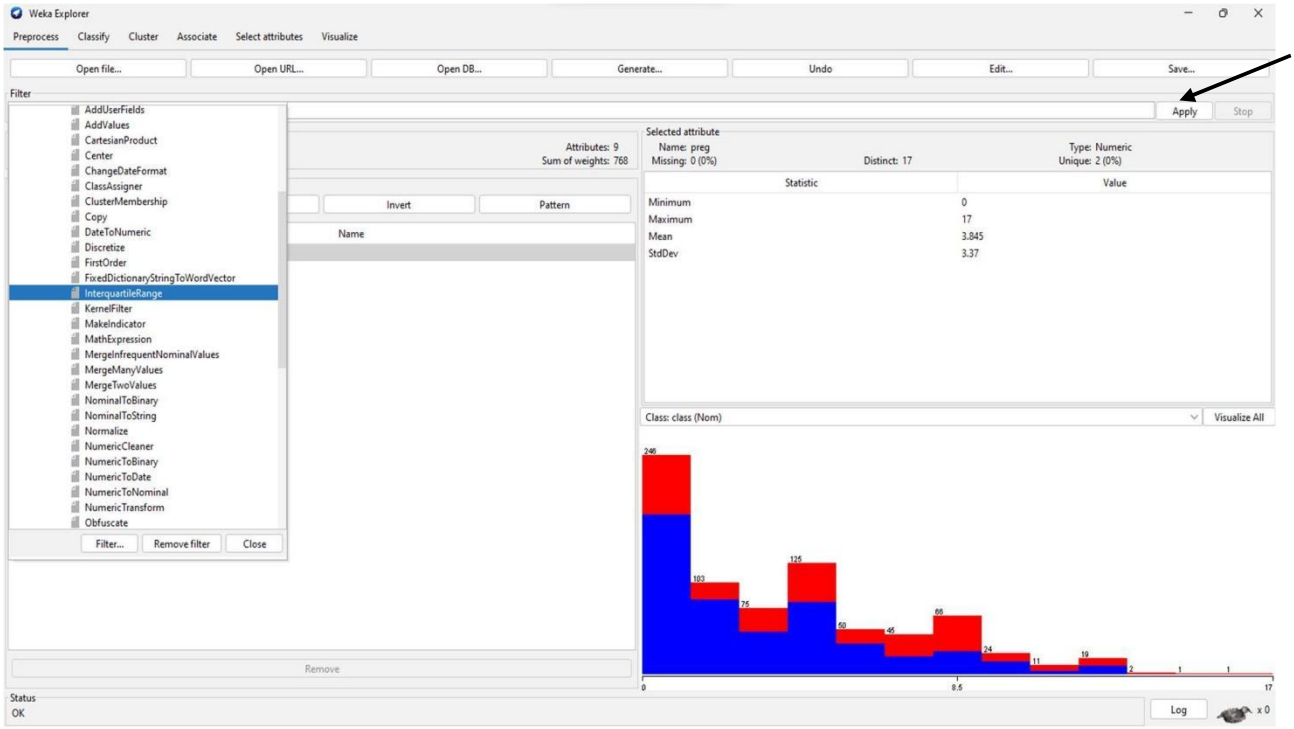


REMOVING OUTLIERS FROM THE DATASET:-

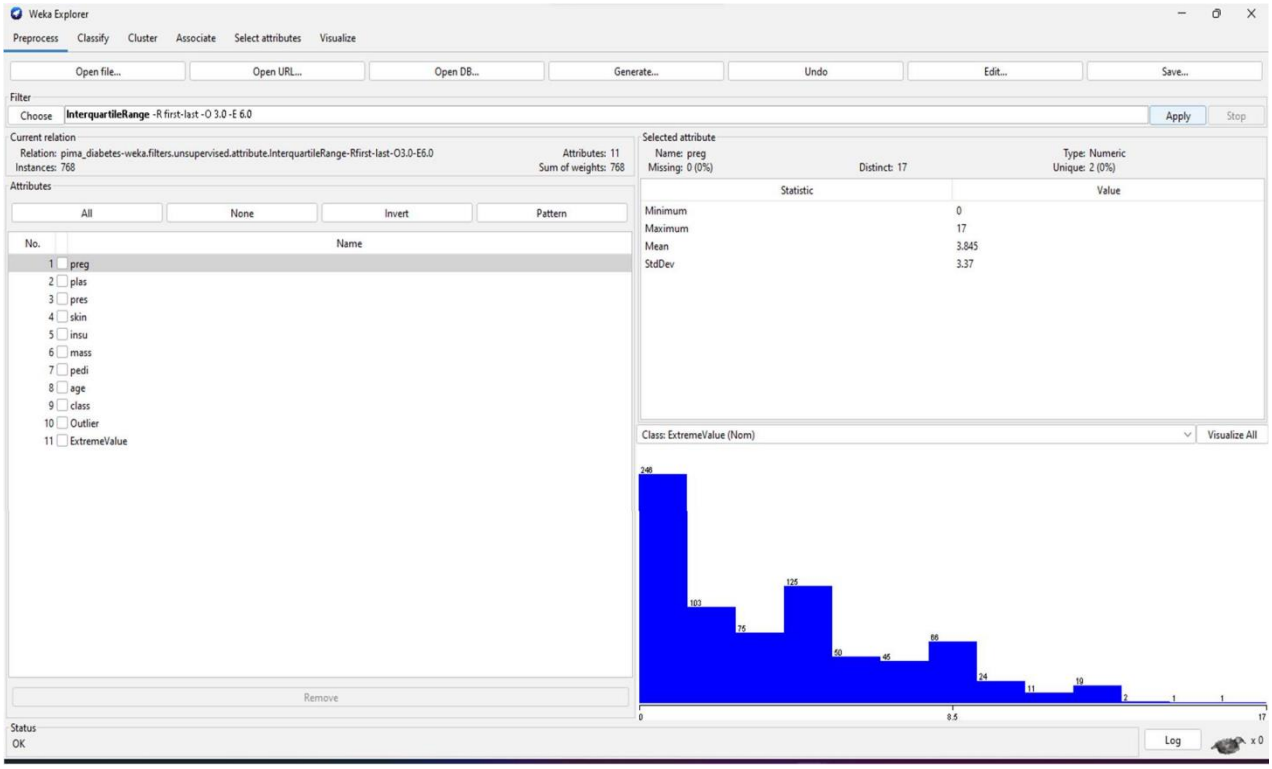
Steps involved are as follows:-

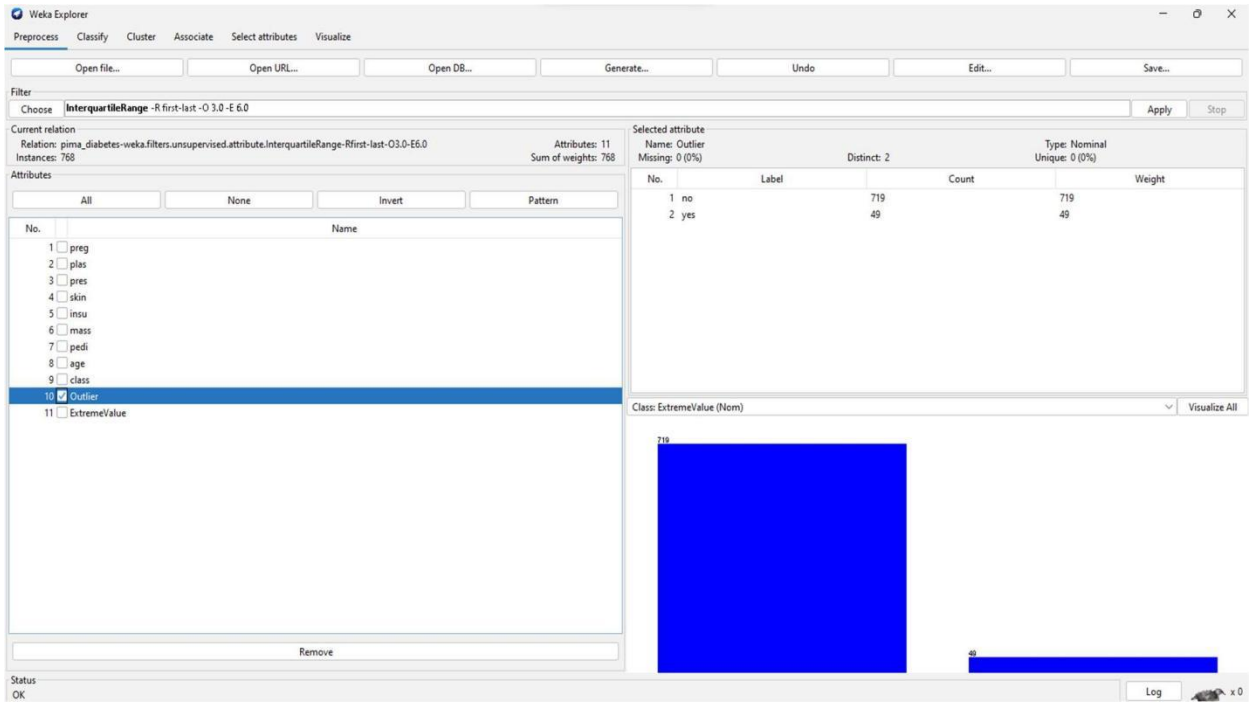
Step 1 :- Select the dataset "diabetes.arff":-

Step 2:- To detect the outliers , from "Filter" section go to unsupervised -> attribute -> InterquartileRange and click on Apply.



Step 3:- Check details of selected attribute “Outlier”. There are two labels present: no and yes. ‘Yes’ label tells us number of outliers present and ‘No’ label tells us normal values . So, here 49 outliers values exist out of 768 observations . we need to remove these outliers.

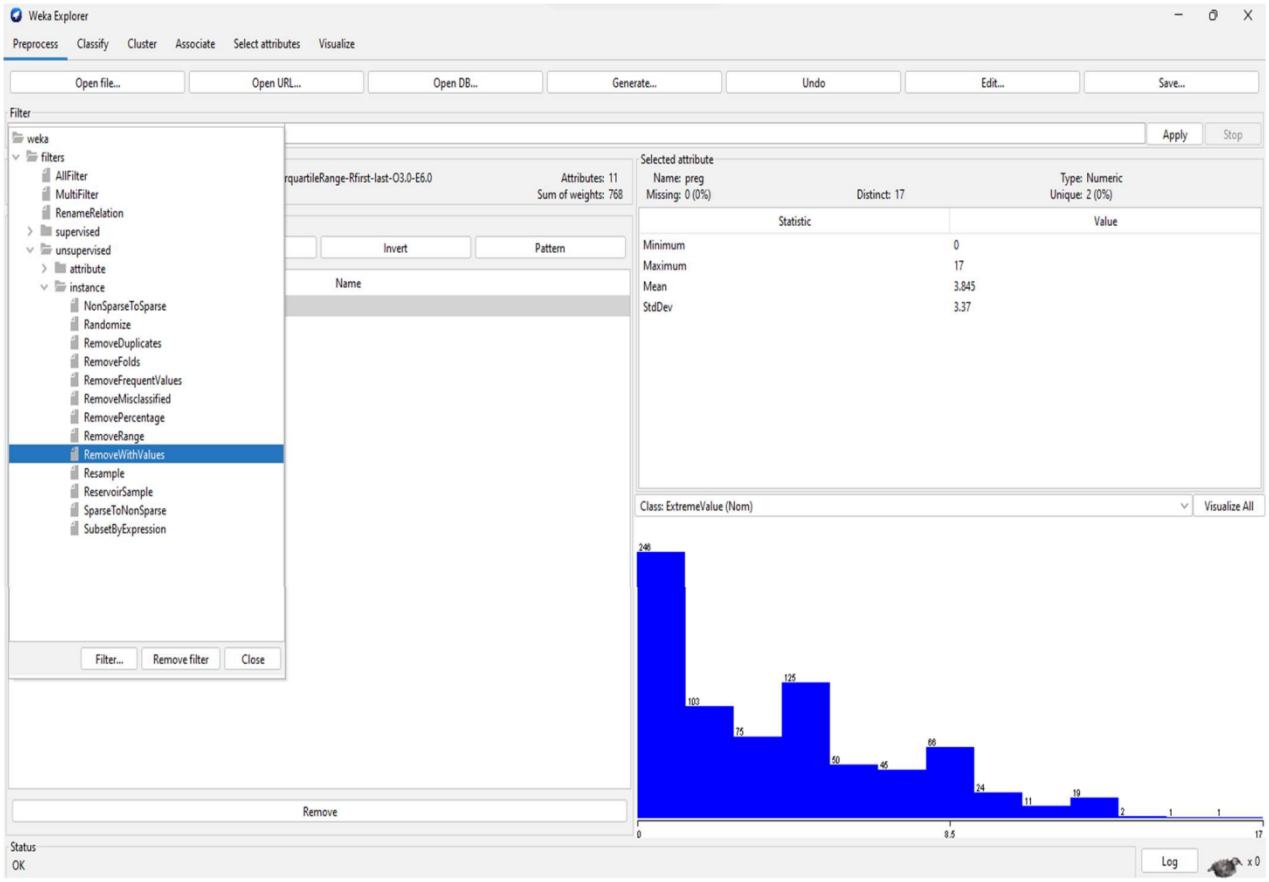




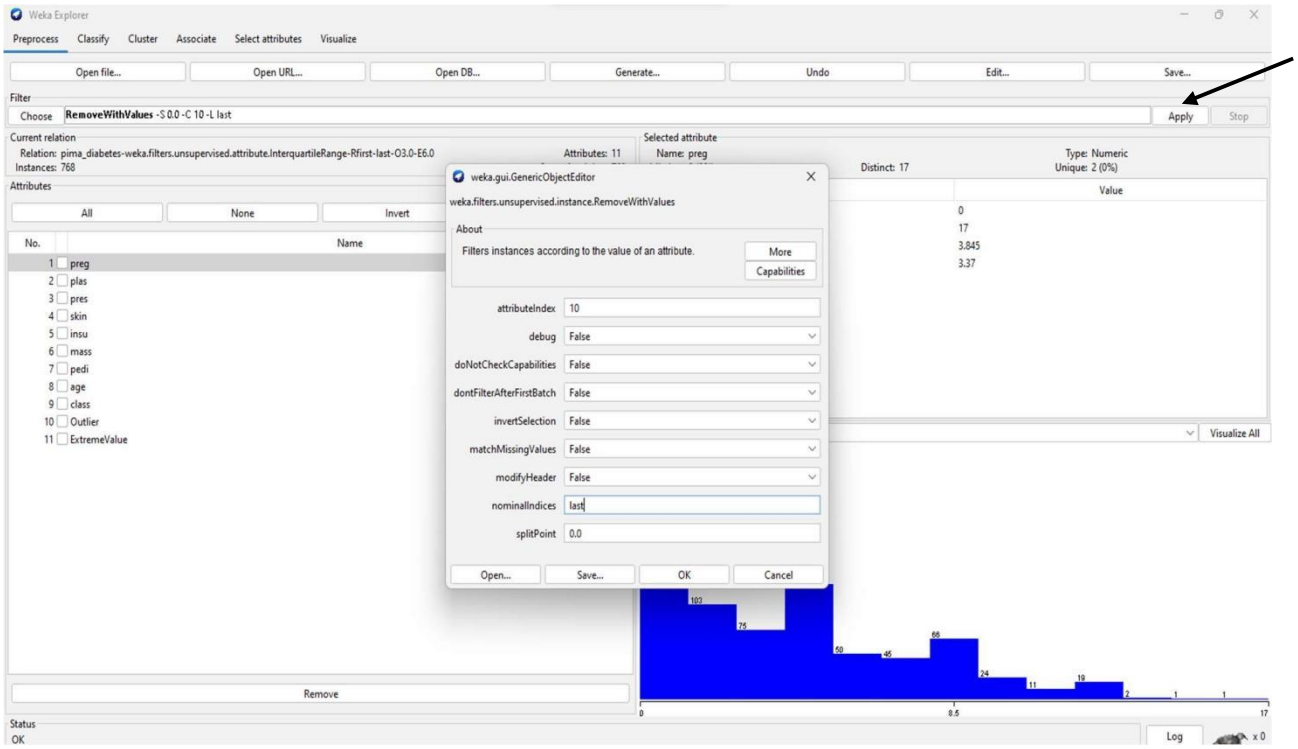
After applying filter save the file with different name and compare the original and updated file after removal of outliers.

Step 4:- To remove the outliers , Select the newly saved(updated) file where we added outliers and ExtremeValue attribute.

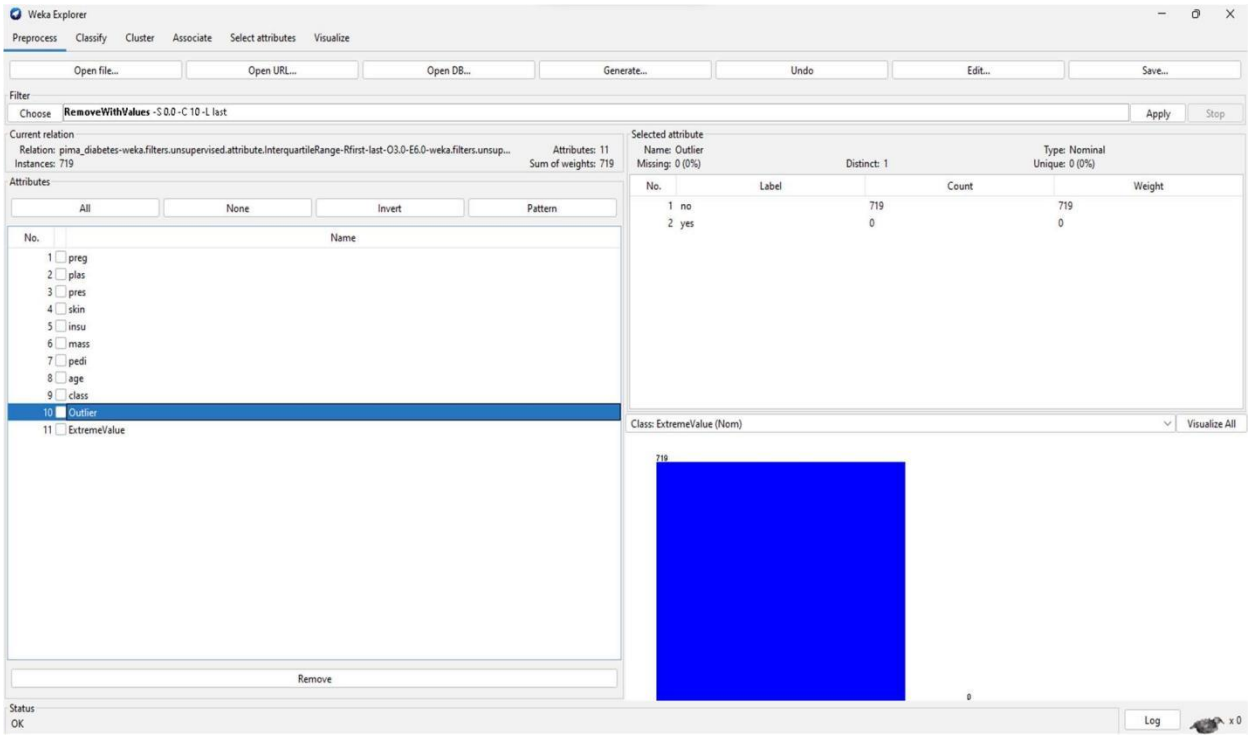
Step 5:- In the Filter section , go to unsupervised -> instance -> RemovewithValues.



Rightclick on RemovewithValues after selecting.You will see the settings: Select the attribute index as 10 as outlier is 10th attribute and set nominalIndices to last. Click on 'OK' and Apply the filter.



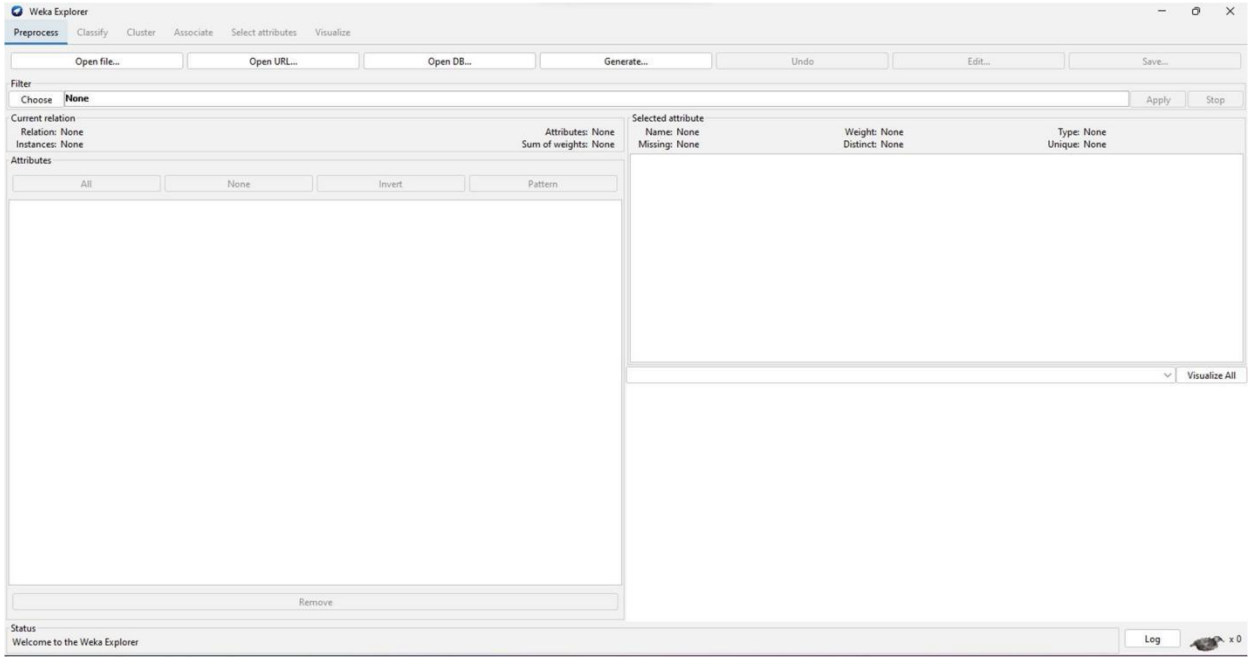
Step 6 :- Check the attribute "Outlier" again. We can see that there are no attributes present in YES label now . This indicates , we have successfully removed the outliers.



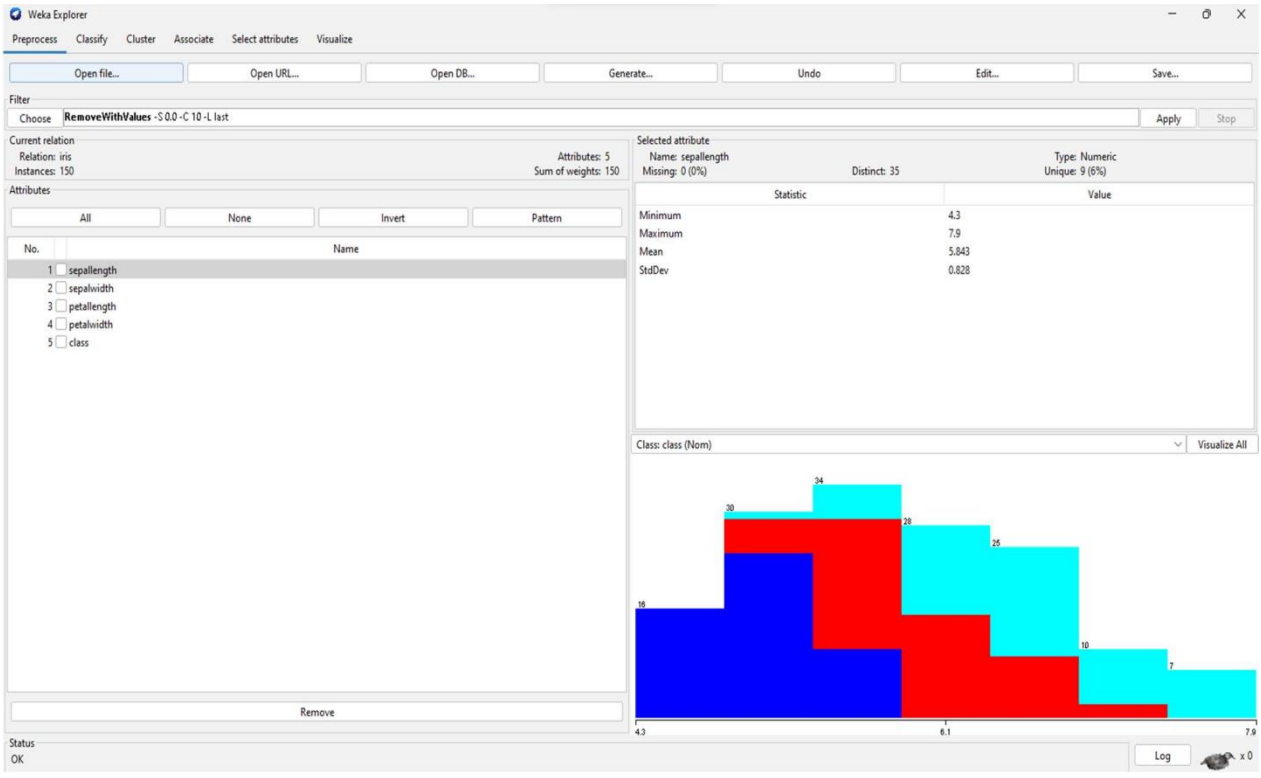
CLASSIFICATION USING WEKA :-

1) Decision Tree Classifier :-

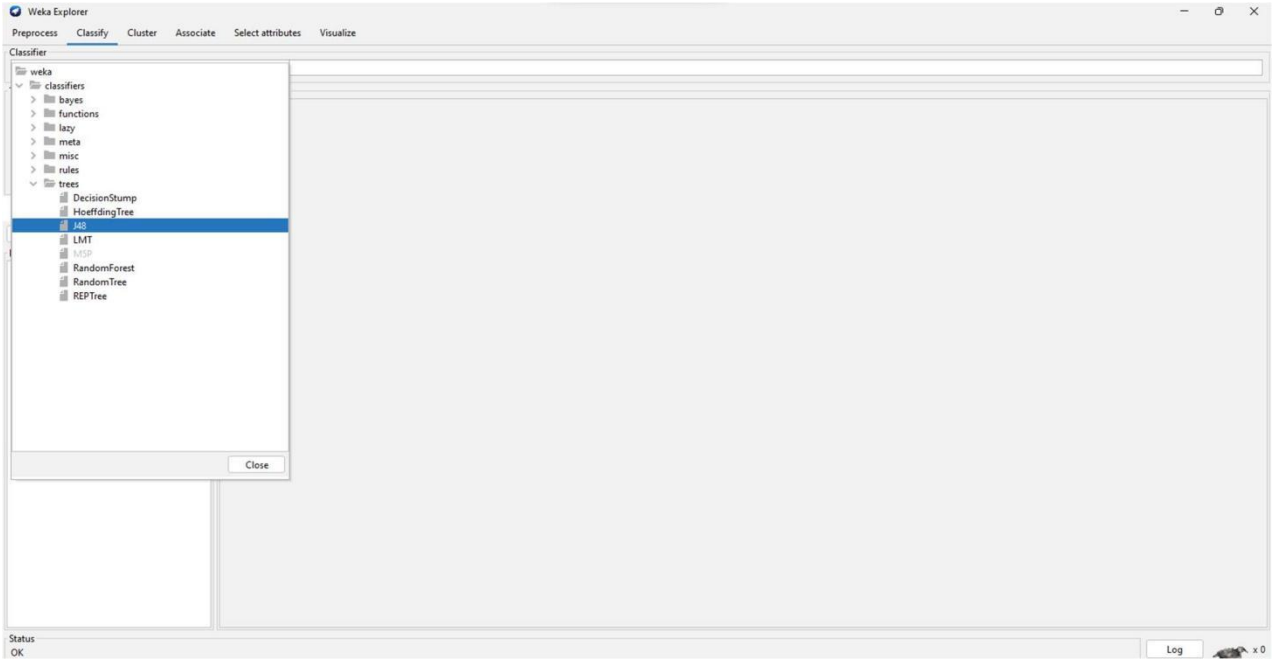
Step 1 :- Open Weka and click on Explorer



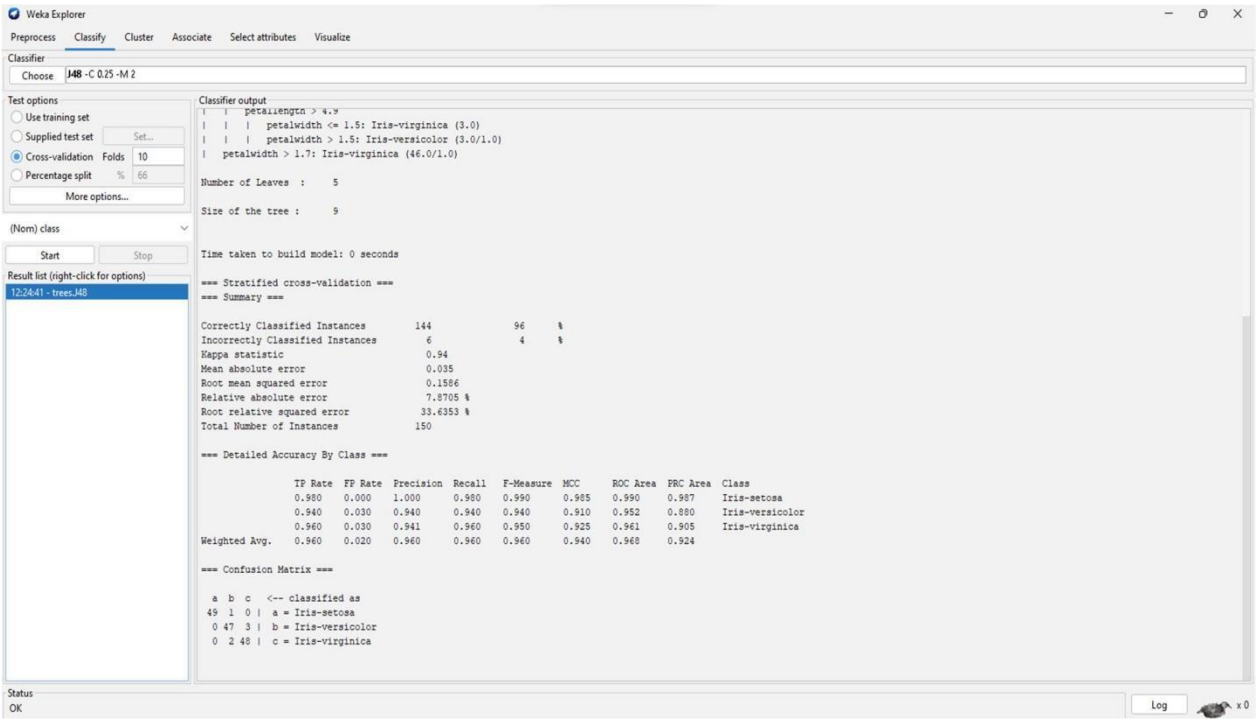
Step 2 :- Select iris.arff dataset from Open file section .



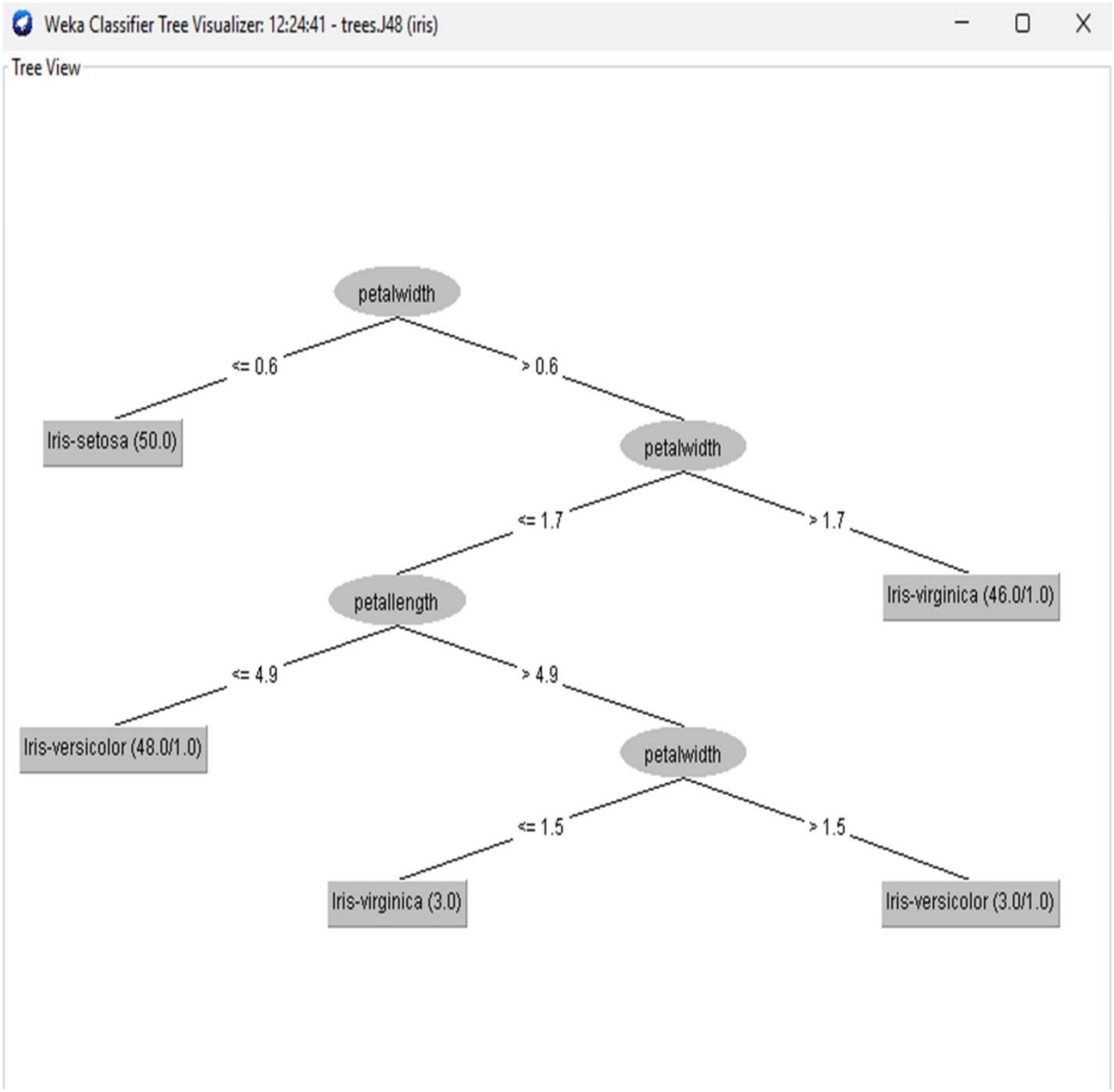
Step 3 :- Click on “Classify” . Select the classification algorithm to be executed on the dataset from “Choose” option . Here we have selected J48 algorithm under trees which create decision tree .



Step 4 :- Click on start to execute the algorithm . The output can be seen in output window.

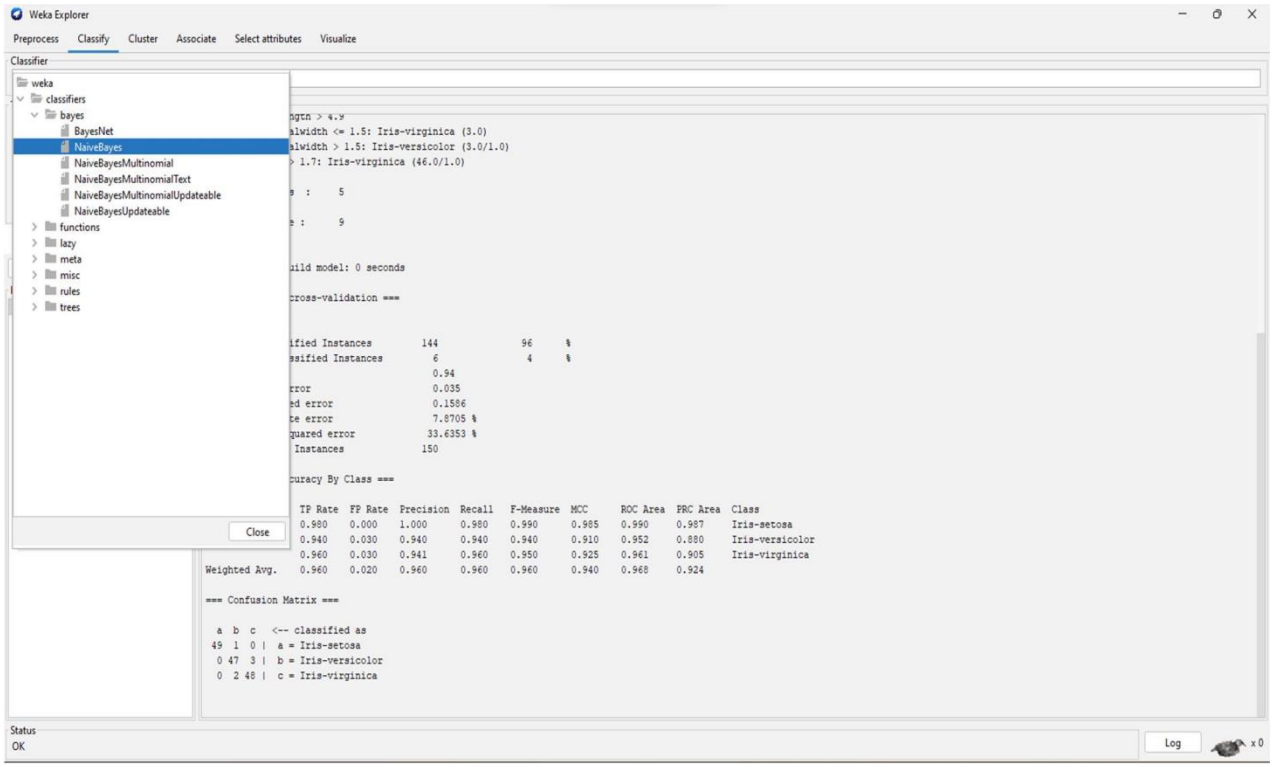


Step 5 :- to visualize the tree , Right click on result list and click on visualize tree.

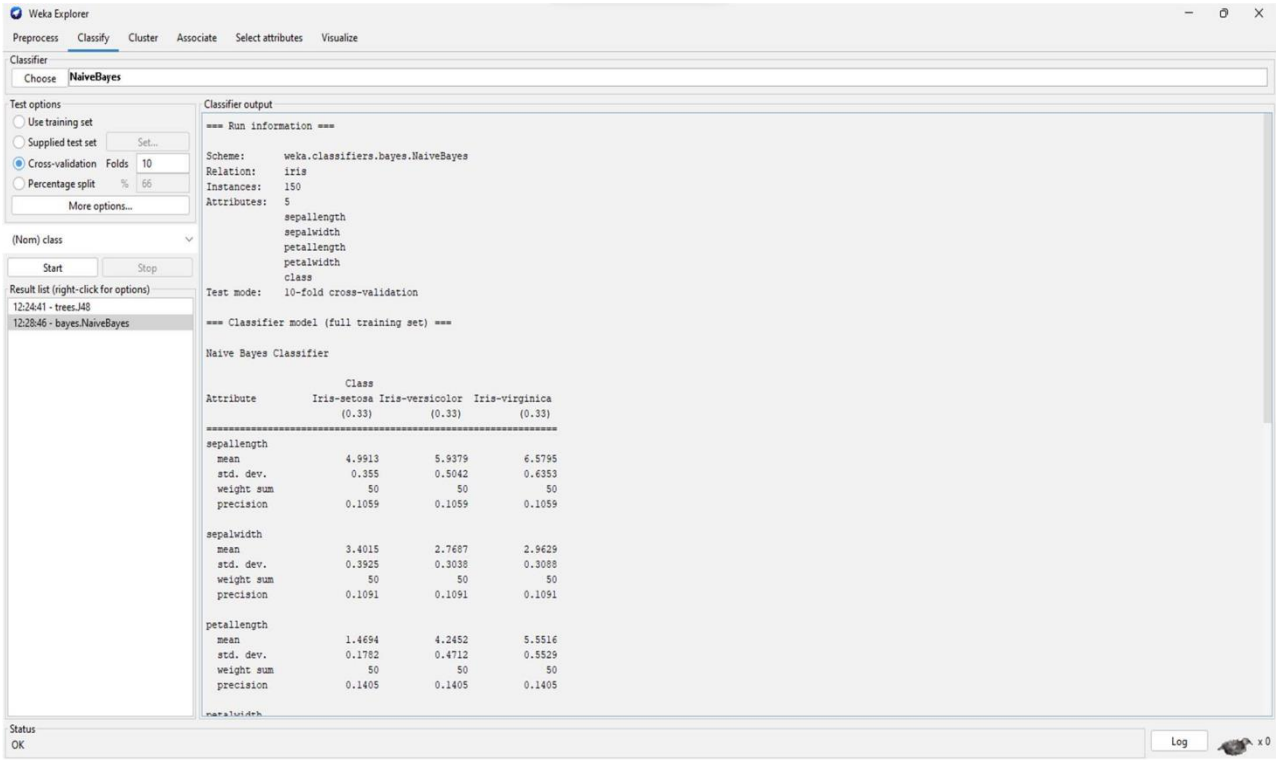


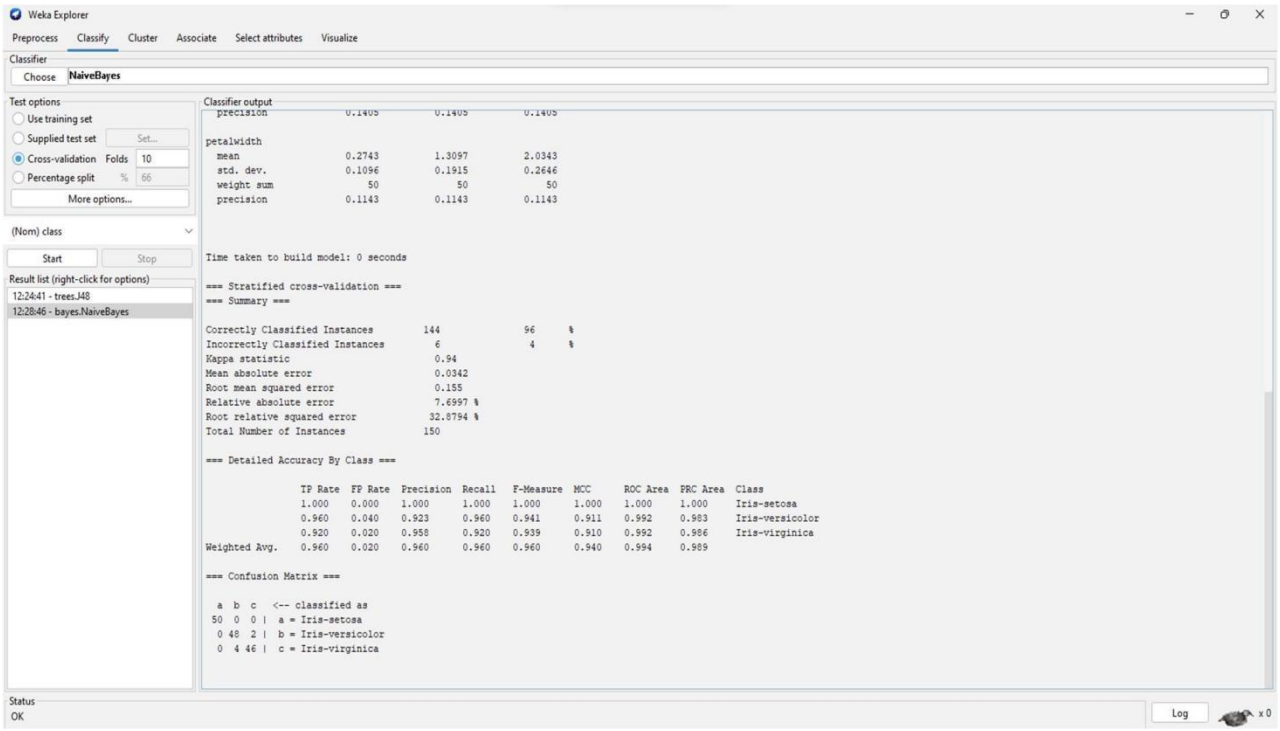
2) Naïve Bayes Classifier :-

Step 1 :- Select the Naves Bayes Classifier under Classifiers -> Bayes -> Naïve Bayes and click on start



Output after applying Naves Bayes:-





B.2 Observations and learning:

In this experiment we are using open source software – Weka for following operation:

1) Data preprocessing using Weka
2) Classification , Clustering and Association Rule mining on data set using Weka

Firstly we explored and understand Weka tool and its features in order to perform above operation . After understanding the Weka tool we perform Data preprocessing on “breast-cancer.arff” dataset and filled the missing values by mean/median by selecting preprocess section .Then we removed outliers from dataset “diabetes.arff” by following the above mentioned steps . At last we perform classification using Decision Tree Classifier and Naves Bayes Classifier which is provided on classify section and we visualized the decision tree using Weka features.

B.3 Conclusion:

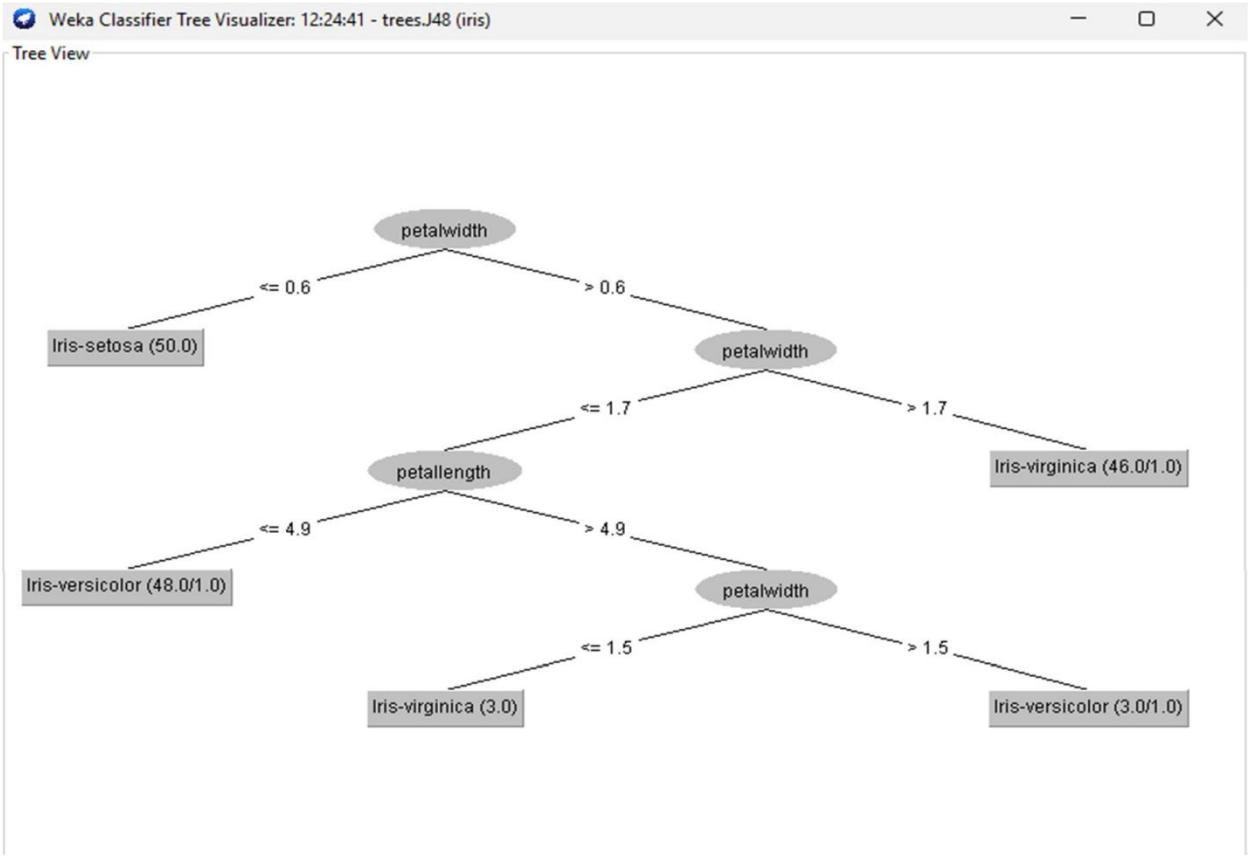
Hence in this experiment, we have successfully performed data preprocessing using Weka and applied Classification , Clustering , Association Rule mining alogirthm on various datasets using Weka through which we got deep insights about the Weka tool and able to perform various machine algorithm for data mining task using Weka.

B.4 Question of Curiosity:-

Q1: Draw the tree according to the classifier output and answers the following questions:

a) What is the depth of the tree?
b) How many leaf nodes are there in the tree?
c) How many tree nodes?

Ans:-



a) Depth of tree is **4**
b) There are **5** leaves node in the decision tree
c) There are **9** nodes in the decision tree