

DATA MINING AND WAREHOUSE

PRACTICAL 1

Name: Jigar Siddhpura

SAP: 60004210155

Division/Batch: B2

Branch: Computer Engineering

Aim

Perform data Pre-processing task using Weka data mining tool

Theory

WEKA - an open-source software provides tools for data pre-processing, implementation of several Machine Learning algorithms, and visualization tools so that you can develop machine learning techniques and apply them to real-world data mining problems.



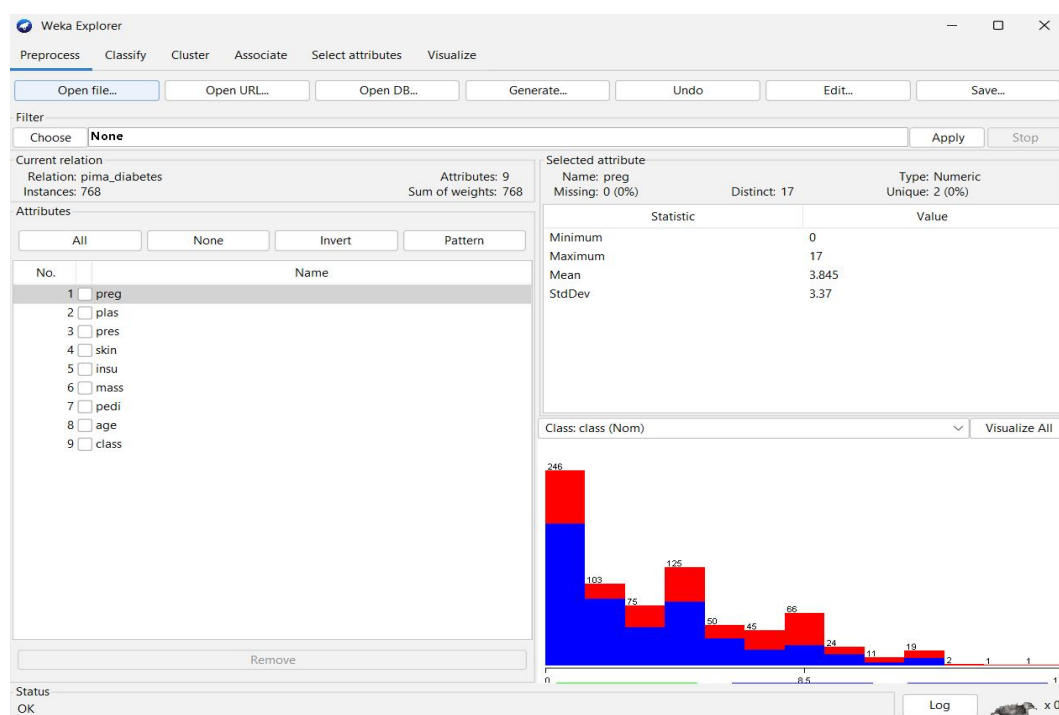
Tasks performed through Weka:

The following tasks are performed through Weka:

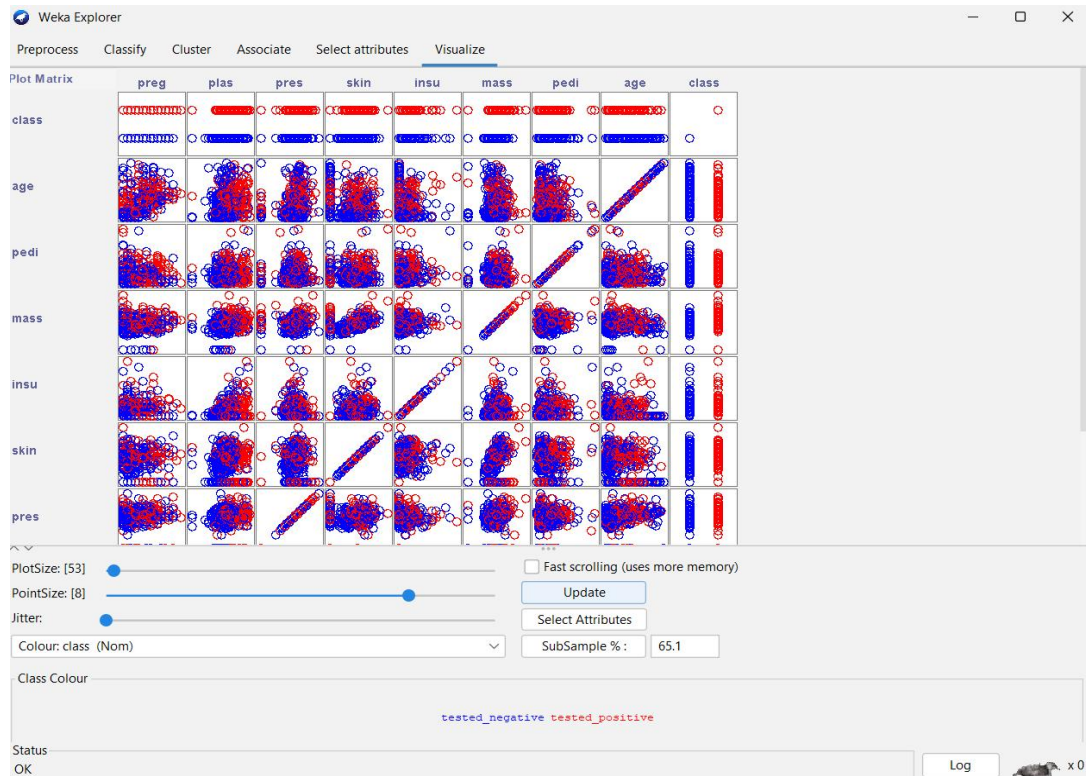
1. **Pre-processing:** Data pre-processing can refer to manipulation or dropping of data before it is used in order to ensure or enhance performance
2. **Classification:** Data classification is the process of analysing structured or unstructured data and organizing it into categories
3. **Clustering:** Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group are more similar to each other than to those in other groups.
4. **Association Rule:** Association rules are used to find correlations and co-occurrences between data sets
5. **Select Attributes:** Attribute subset Selection is a technique which is used for data reduction in data mining process. Data reduction reduces the size of data so that it can be used for analysis purposes more efficiently.
6. **Visualization:** Data visualization is an interdisciplinary field that deals with the graphic representation of data.

Pre-processing activities to be observed in Weka

1. **Visualization:** Visualize scatter plot for all the attributes from the dataset selected from Weka. Determine correlation if any using these plots for different datasets

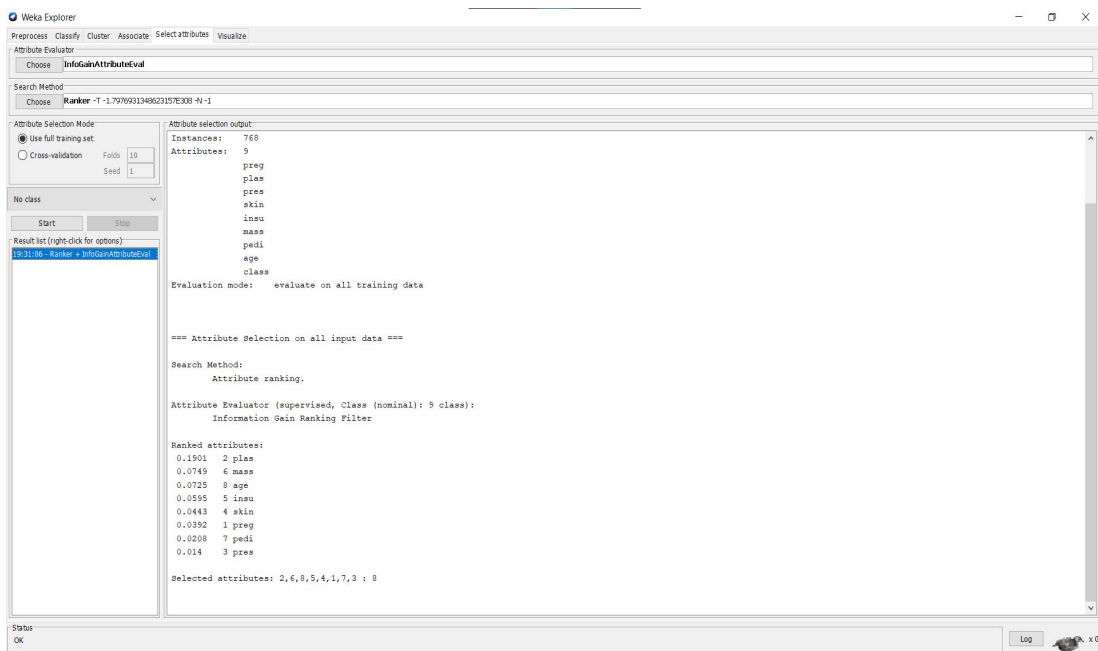


We have loaded the *Diabetes* dataset into Weka and have visualized the various attributes. The histograms for each attribute display the variation in values within that attribute and can be viewed for each attribute.

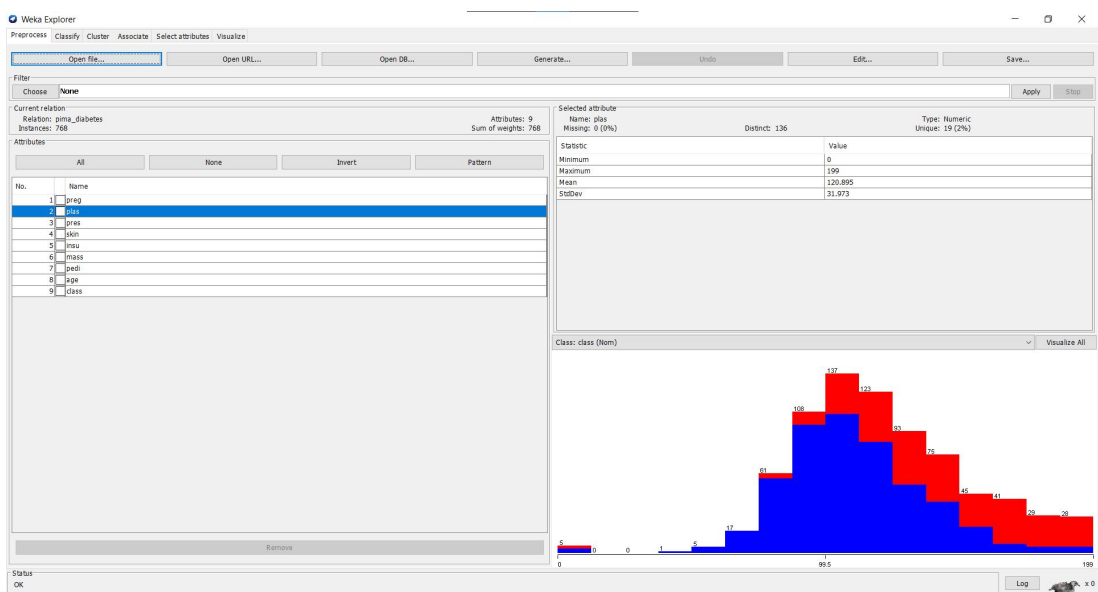


Here we can visualise the scatter plots for each of the attributes of the dataset taken in pairs. By observing the scatter plot, we can infer the following:

- a. Body mass index (weight in kg/ (height in m) ²) vs Triceps skin fold thickness (mm) – There is a positive correlation between these attributes
 - b. 2-Hour serum insulin (mu U/ml) vs Plasma glucose concentration 2 hours in an oral glucose tolerance test – There is a positive correlation between these attributes
 - c. Diabetes pedigree function vs Triceps skin fold thickness (mm) – There is no correlation between these attributes and the plot is sparsely distributed
2. **Select Attributes:** Apply suitable feature selection filters like Gain Ratio etc to choose relevant attributes from the list of attributes. Observe the ranks / priority provided by the filter.



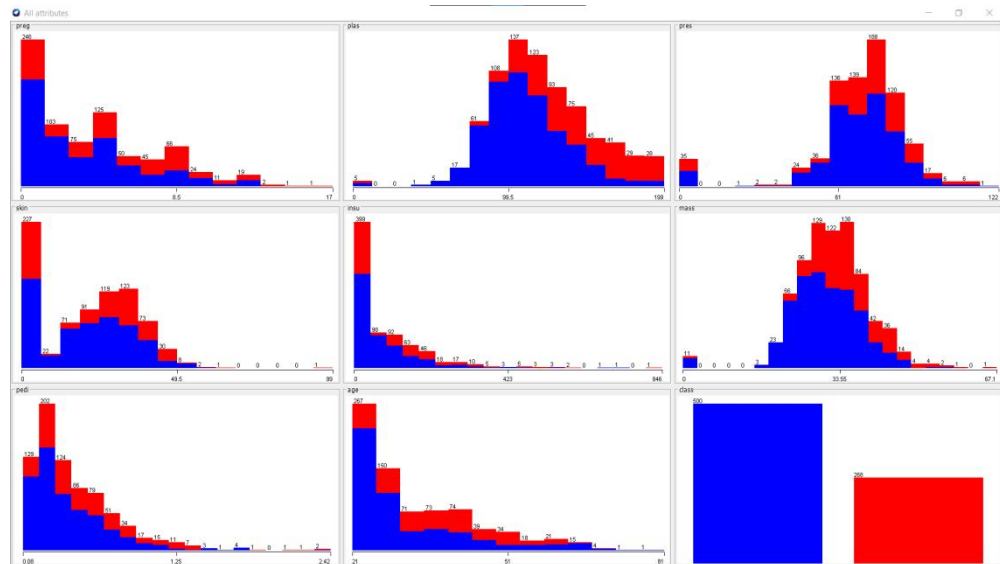
We now switch to the Select Attribute tab and have used the InfoGainAttributeEval method with the attribute Ranker. From the above results, the attribute ‘plas’ which stands for ‘Plasma glucose concentration 2 hours in an oral glucose tolerance test’ is the highest ranked with a score of 0.1901. This shows that the plasma attribute is the most important attribute for further processing.



The data can be seen influenced by the attribute 'plas' and a comparably clear separation in classes correlating with its values.

3. Pre-processing:

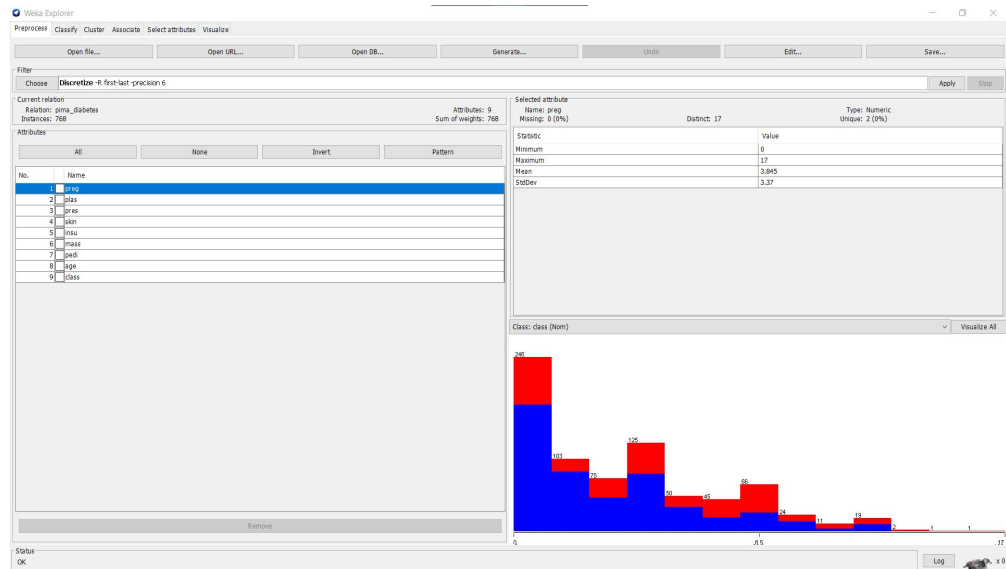
- a. **Visualize All:** Select this button to visualize histograms of all attributes.



Here we have visualized all of the attributes and their histograms. We can infer that some attributes like skin, pedi, preg, insu, age are left skewed while pres, plas, mass are normally distributed.

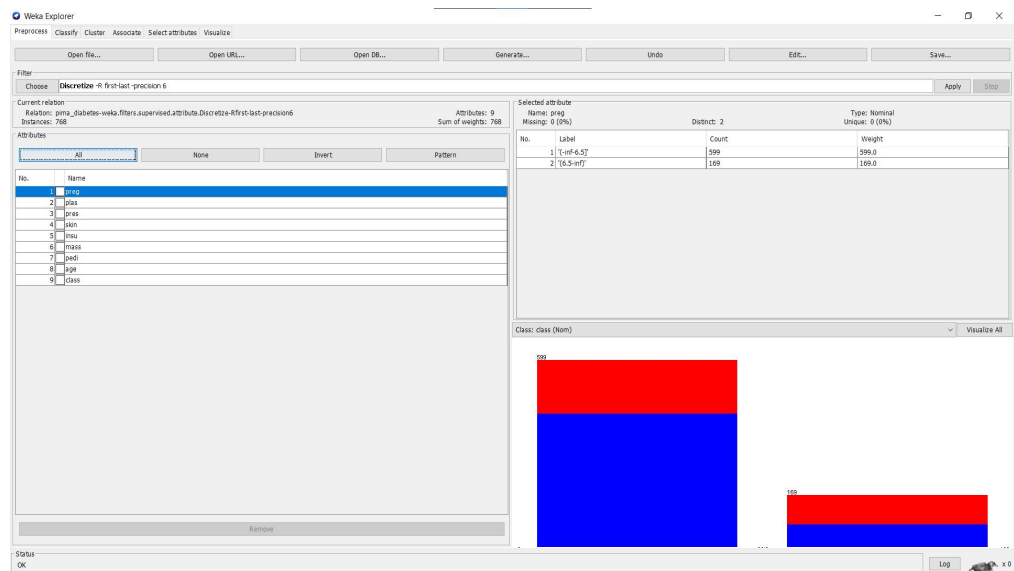
- b. **Filter:** Choose Discretization under Unsupervised and Supervised methods. Observe the discretization and the outliers.

Before:



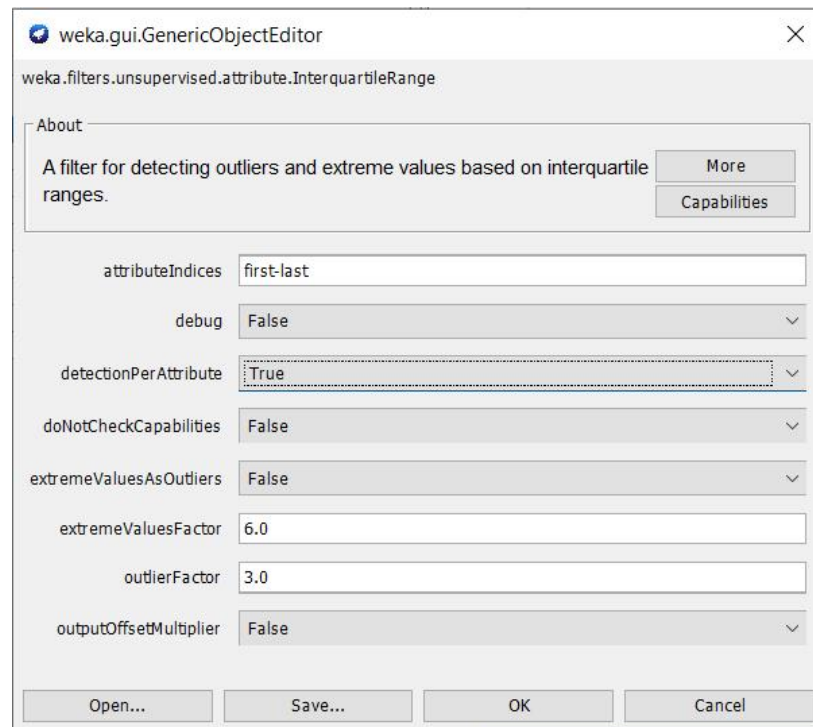
The attribute value before discretization shows a continuous histogram with varying values and no of instances.

After:

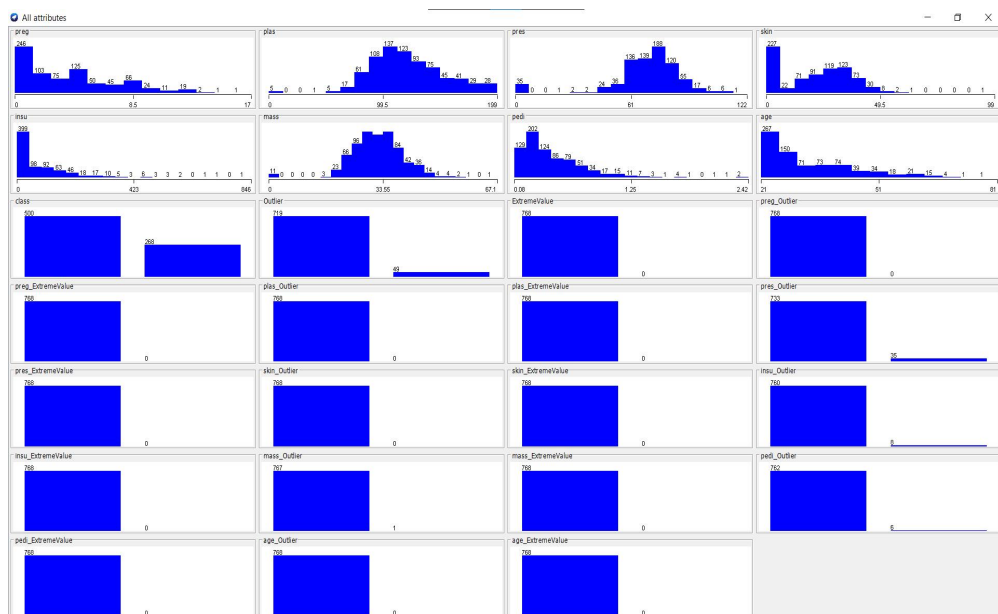


Here we have applied unsupervised discrete filtering to the dataset. For each attribute discrete buckets are created, and the instances are separated into their appropriate bucket. The attribute here is nominal and hence the bucket is split based on their values and extends from -infinity to 6.5 and from 6.5 to infinity respectively.

- c. **IQR:** Observe the IQR values for a selected attribute. Observe the outlier and extreme values



These are the settings that have been applied for IQR.



When we apply the Inter Quartile Range (IQR) per attribute, we are able to see the outliers present in each of the attributes separately along with the extreme values that are present. Outliers are reported at less than $Q1 - 1.5 * IQR$ and greater than $Q3 + 1.5 * IQR$.

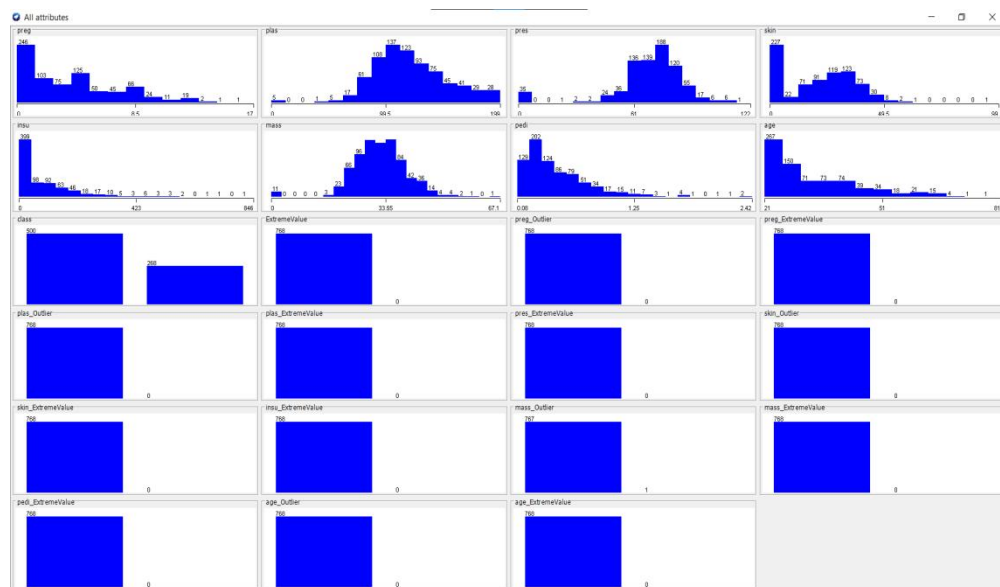
- d. **Remove the value:** Remove instances with outlier values and show the screenshots of dataset before and after the removal.

Before:



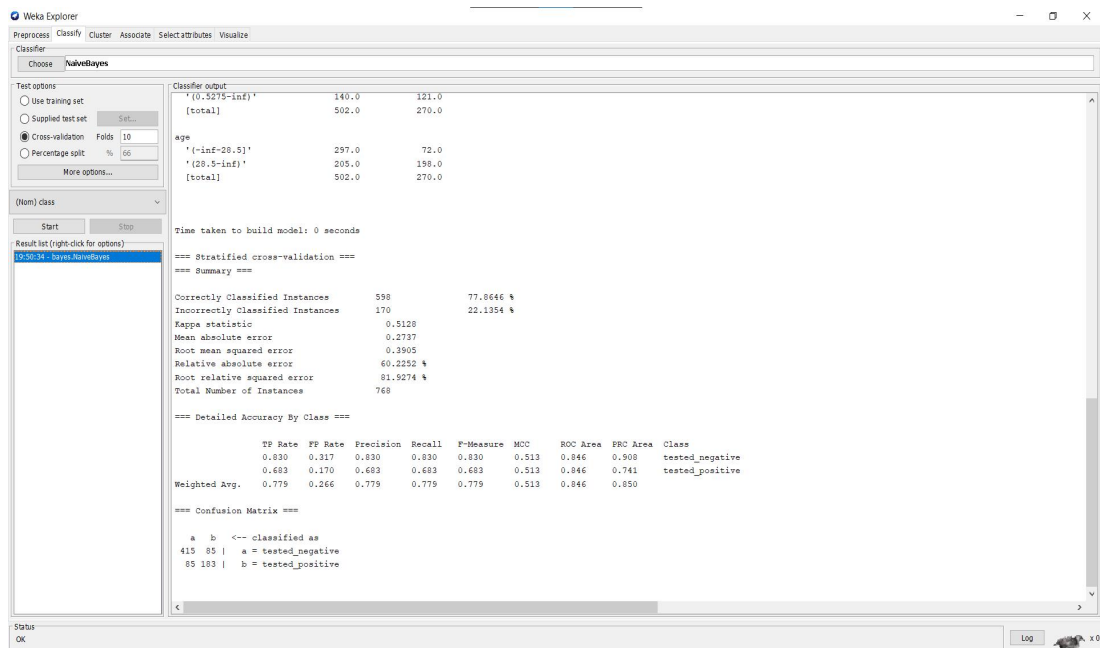
From the above IQR plot, we can determine the attributes that have the outliers. From this dataset, the attributes ‘predi’, ‘insu’ and ‘pres’ have outliers with a combined total of 49 instances.

After:



We can then use the attributes menu to remove those instances from the respective attributes and again visualising the attributes graphs shows that the outliers in the dataset have been removed.

4. Classification: Perform NB, kNN and DT/rule-based classification



The screenshot shows the Weka Explorer interface with the Naive Bayes classifier selected. The 'Test options' section shows 'Cross-validation' with 'Folds' set to 10. The 'Classifier output' pane displays the following results:

```
Classifier output
(0.5275-inf)'      140.0      121.0
[total]            502.0      270.0

age
'(-inf-28.5]'      297.0      72.0
'[28.5-inf)'       205.0      198.0
[total]            502.0      270.0

Time taken to build model: 0 seconds

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

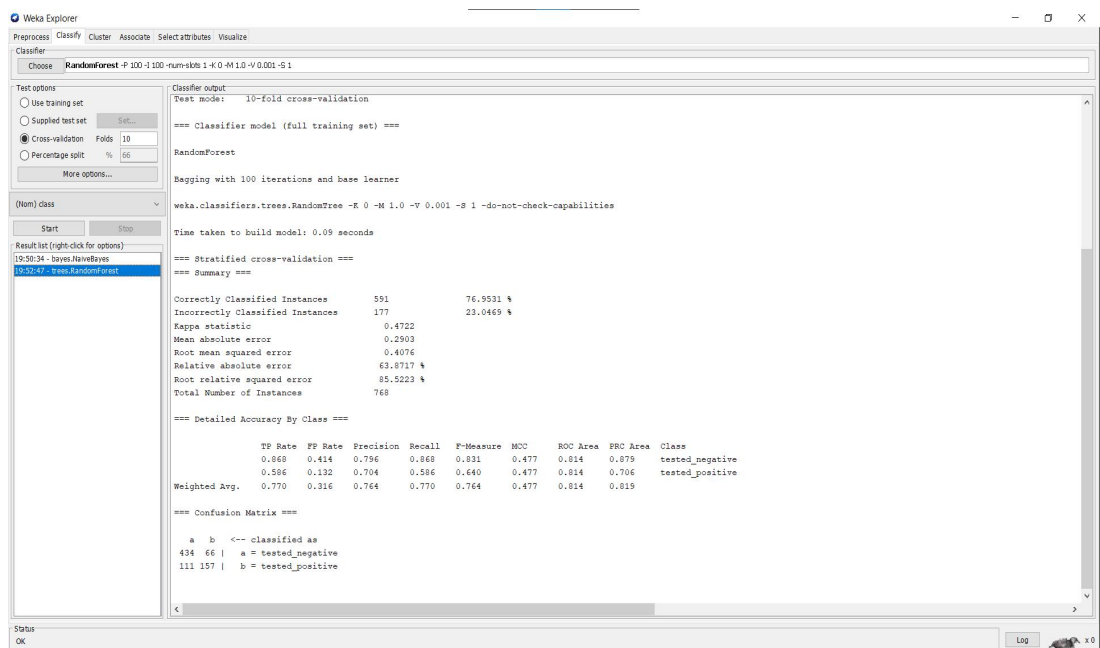
Correctly Classified Instances      598      77.8646 %
Incorrectly Classified Instances    170      22.1354 %
Kappa statistic                    0.5128
Mean absolute error                0.2737
Root mean squared error            0.3905
Relative absolute error            60.2252 %
Root relative squared error        81.9274 %
Total Number of Instances         768

=== Detailed Accuracy By Class ===

          TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
0.693   0.170   0.683    0.683   0.683    0.513   0.846   0.741   tested_negative
0.779   0.266   0.779    0.779   0.779    0.513   0.846   0.850   tested_positive
Weighted Avg.   0.779   0.266   0.779    0.779   0.779    0.513   0.846   0.850

=== Confusion Matrix ===
      a  b  <-- classified as
415  85 |  a = tested_negative
 85 183 |  b = tested_positive
```

We have performed Naïve Bayes classification on the dataset. The Naïve bayes classifier correctly classified 77.86% of the instances while 22.13% were incorrectly classified.



The screenshot shows the Weka Explorer interface with the Random Forest classifier selected. The 'Test options' section shows 'Cross-validation' with 'Folds' set to 10. The 'Classifier output' pane displays the following results:

```
Classifier output
Test mode: 10-fold cross-validation

=== Classifier model (full training set) ===

RandomForest

Bagging with 100 iterations and base learner
weka.classifiers.trees.RandomTree -K 0 -M 1.0 -T 0.001 -B 1 -do-not-check-capabilities

Time taken to build model: 0.09 seconds

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

Correctly Classified Instances      591      76.9531 %
Incorrectly Classified Instances    177      23.0469 %
Kappa statistic                    0.4722
Mean absolute error                0.2903
Root mean squared error            0.4076
Relative absolute error            65.8717 %
Root relative squared error        85.5223 %
Total Number of Instances         768

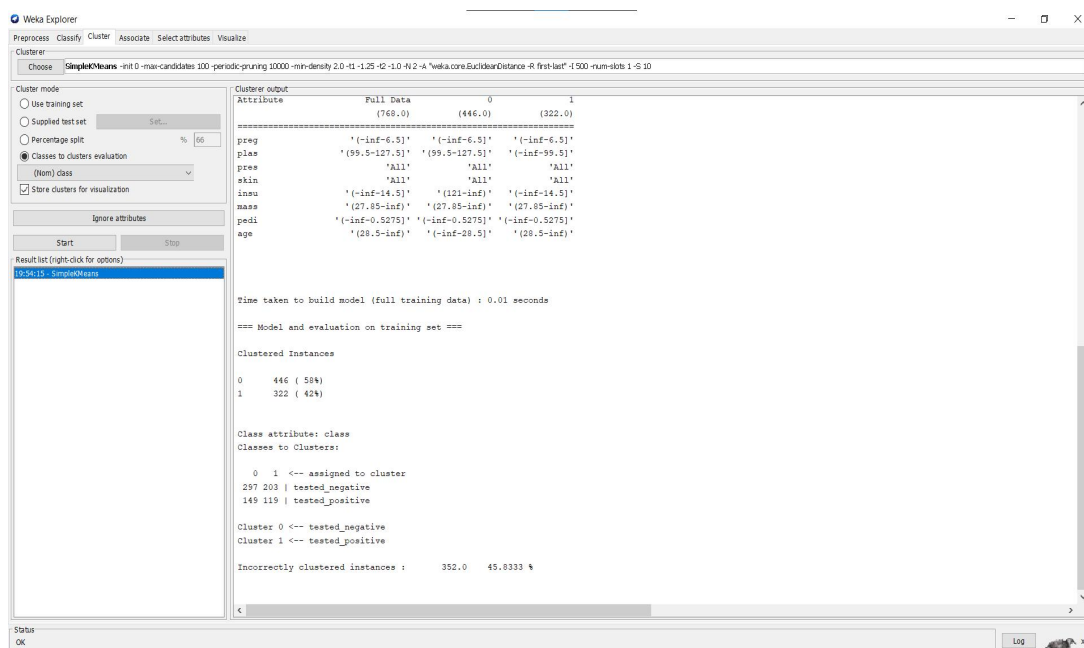
=== Detailed Accuracy By Class ===

          TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
0.669   0.414   0.796    0.868   0.831   0.477   0.814   0.879   tested_negative
0.586   0.132   0.704    0.586   0.640   0.477   0.814   0.706   tested_positive
Weighted Avg.   0.770   0.316   0.764    0.770   0.764   0.477   0.814   0.819

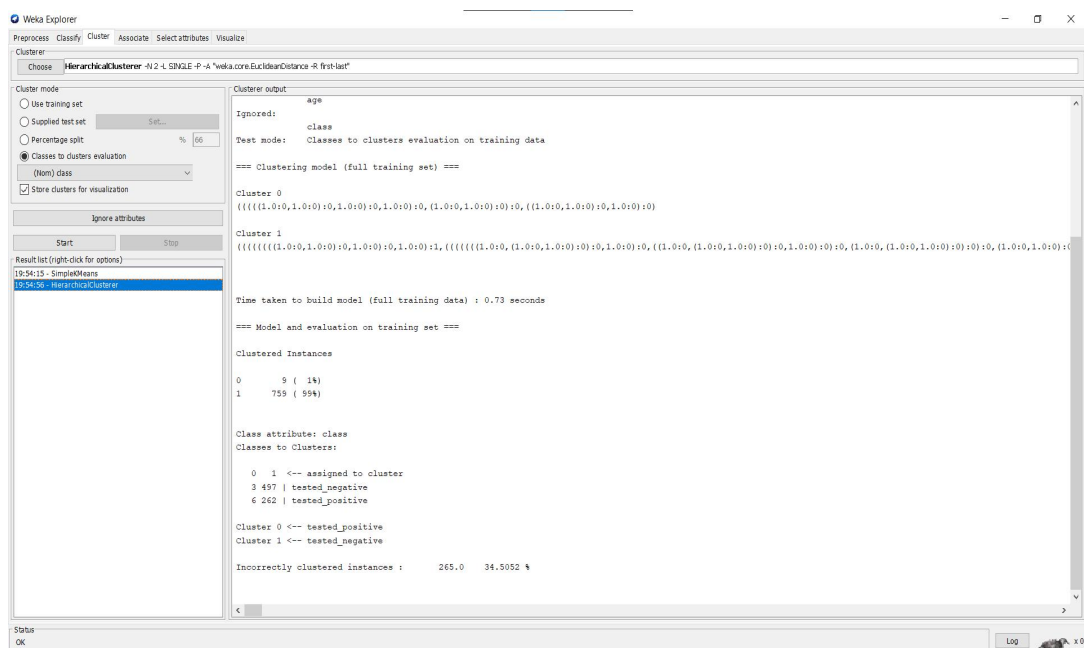
=== Confusion Matrix ===
      a  b  <-- classified as
434  66 |  a = tested_negative
111 157 |  b = tested_positive
```

We have performed Random Forest (Decision Tree/Rule Based) classification on the dataset. The Random Forest classifier correctly classified 76.95% of the instances while 23.04% were incorrectly classified.

5. **Clustering:** Perform kmeans, hierarchical clustering and explain the output

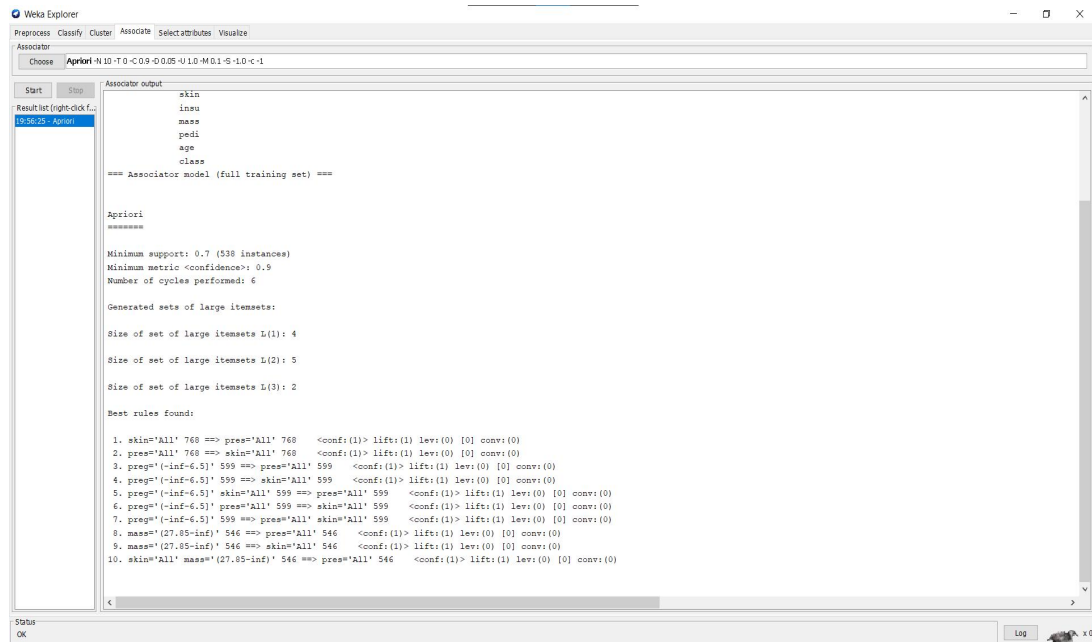


We have performed K-Means Clustering on the dataset to form groups of similar instances from the data. It incorrectly clustered 45.83% of the instances.



We have performed Hierarchical Clustering on the dataset to form groups of similar instances from the data. It incorrectly clustered 34.51% of the instances, thus being comparatively better than the K-Means clustering algorithm for this dataset.

6. Association rule mining: Perform apriori algo and show the rules created



Associative rule mining is used to find associations in the data. Weka has an associate tab that allows us to perform associative rule mining using algorithms like apriori.

Here we have applied apriori to the dataset. From the results, we can infer that:

- The attributes skin, mass and pres are associative
- The attributes mass and pres are associative
- The attributes preg and skin are associative

Conclusion

Weka is a very intuitive tool for exploring datasets and performing data mining operations. It provides extensive support for data visualisations through histograms to visualise variation in the instances within the data and through scatter plots for correlating various attributes. It also provides IQR which allows us to detect and remove outliers. We can also filter the data through discretisation and other processes and visualise the state of the data while doing so. It also allows us to select attributes from the dataset and then perform classification and clustering on the data to identify similar groups. Weka also allows us to perform associative mining on the data to find associated attributes for basket analysis.