Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

A.Y. 2022-2023

DATA MINING AND WAREHOUSE

AYUSH JAIN

COMPUTER ENGINEERING | TE – B2 | 60004200132

EXPERIMENT – 1

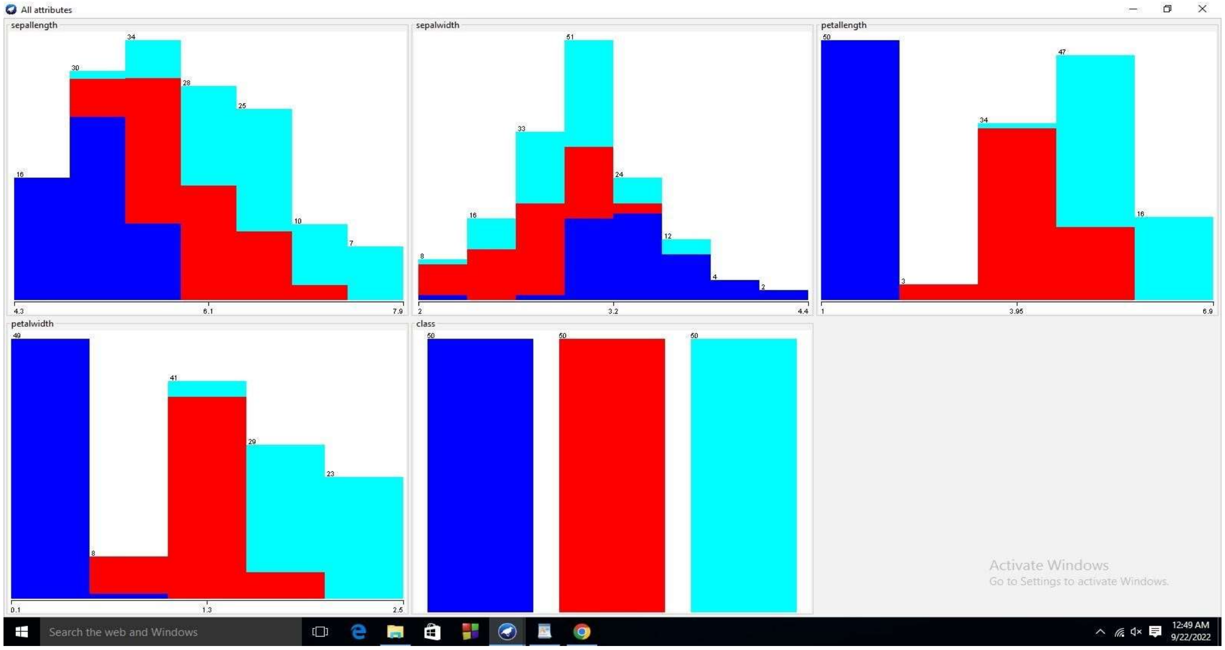
Aim: Perform data Pre-processing task using Weka data mining tool

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:

1. Pre-processing:

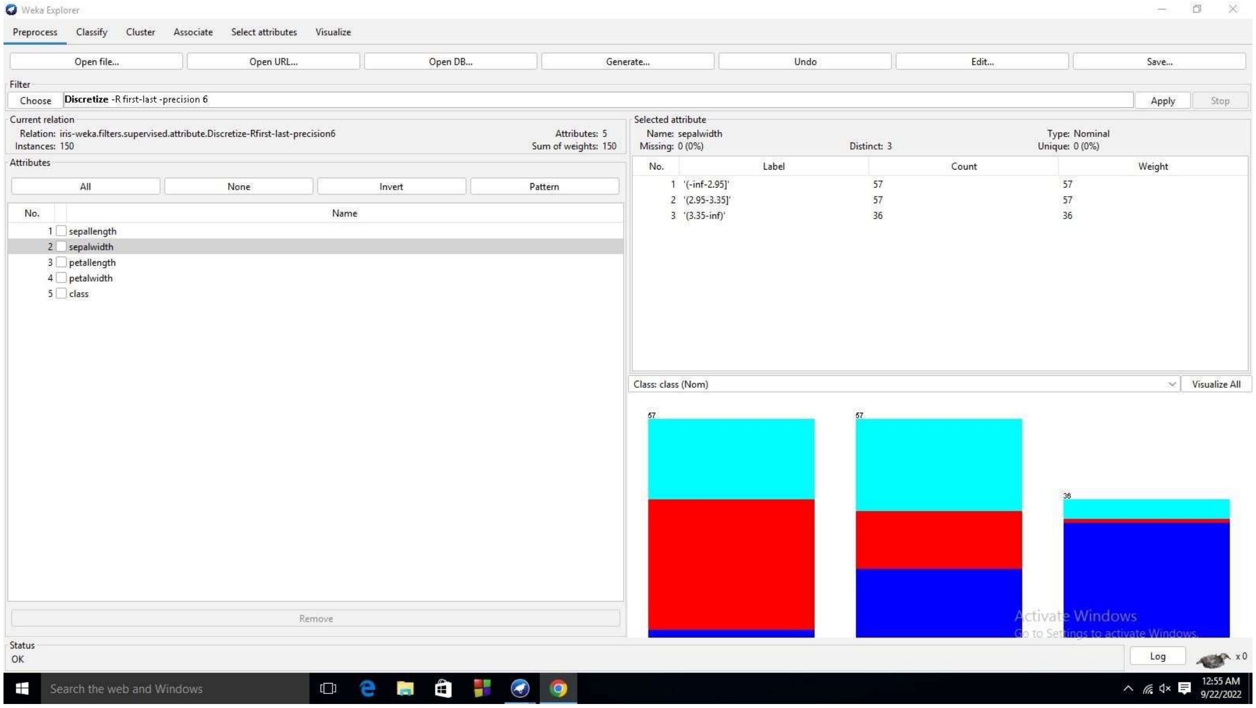
- **Visualize-all:**



This is histogram visualization of all attributes without any filter for iris dataset. We can infer that labels in sepal length and sepal width are closely related in terms of frequency distribution and sepal length is skewed-right and sepal width is skewed-left whereas label in petal length and petal width are not so closely related in terms of frequency distribution.

- **Filter:**

  i) Discretization under supervised learning:



Filter
Choose **Discretize** -R first-last -precision 6

Current relation
Relation: iris-weka.filters.supervised.attribute.Discretize-Rfirst-last-precision6
Instances: 150
Attributes: 5
Sum of weights: 150

Attributes

No.		Name
1	□	sepallength
2	□	sepalwidth
3	□	petallength
4	□	petalwidth
5	□	class

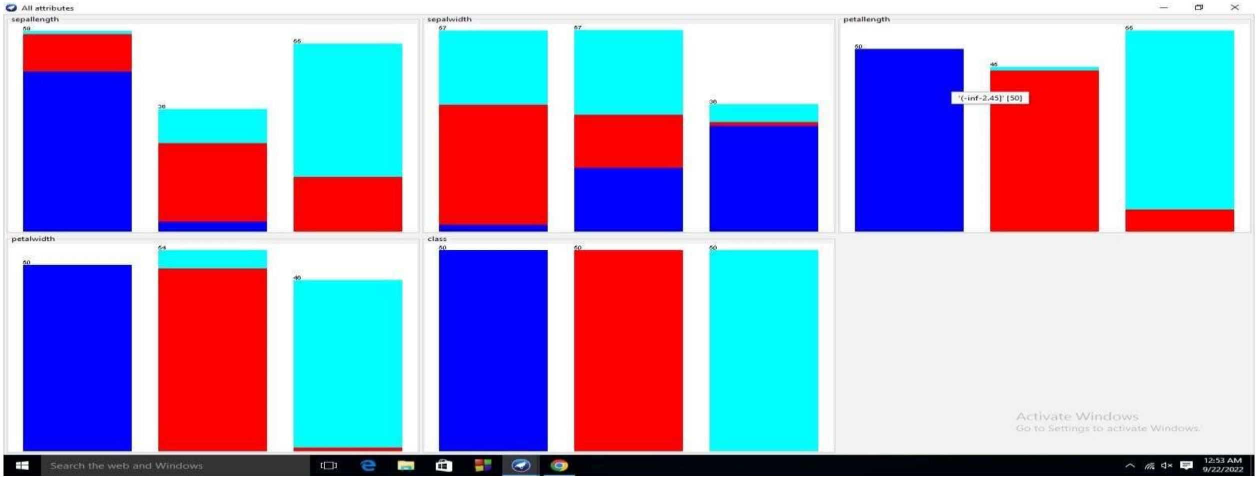
Selected attribute
Name: sepalwidth
Missing: 0 (0%)
Distinct: 3
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	'(-inf-2.95]'	57	57
2	'(2.95-3.35]'	57	57
3	'(3.35-inf)'	36	36

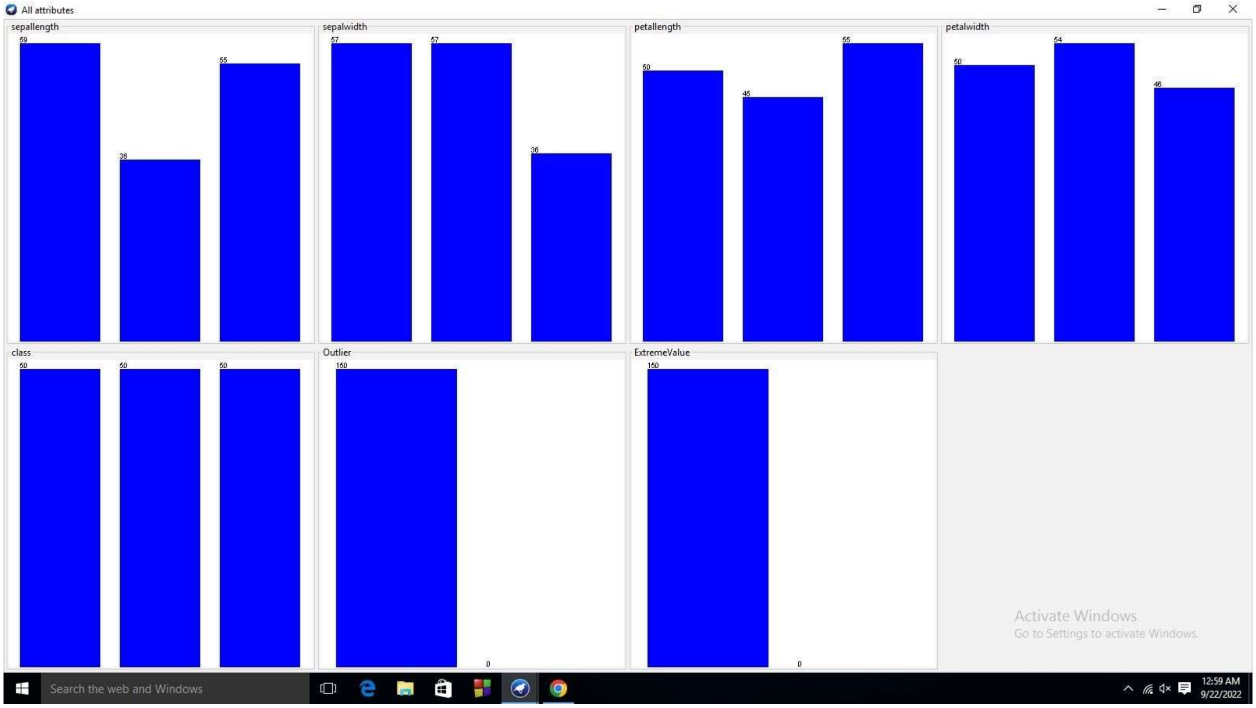
Class: class (Nom)

Status
OK

This is Pre-processing window of discrete supervised class (nominal datatype) where we can see it created three labels of different numerical intervals.

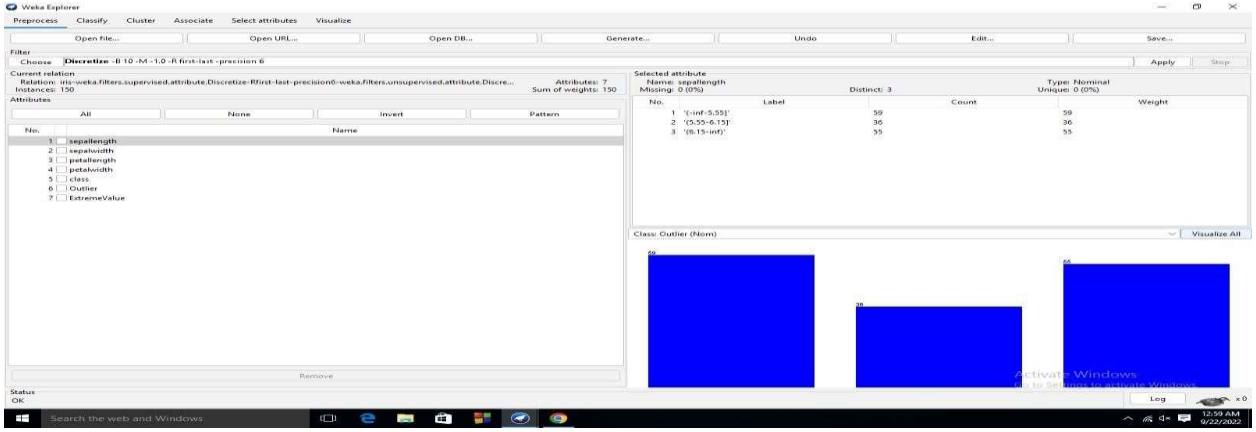


This is stacked column chart of discretized supervised learning



Above picture shows the extreme and outlier values of different attributes extreme values for discretize supervised learning. Insights from this is that extreme value for sepal length is 59 and two numerical intervals have same extreme value of 57 in sepal width. Outlier and Extreme values are 150.

ii) Discretization under unsupervised learning:



Filter
Discretize -B 10 -M -1.0 -R first-last -precision 6

Current relation
Relation: iris-weka.filters.supervised.attribute.Discretize-Rfirst-last-precision6-weka.filters.unsupervised.attribute.Discre... Attributes: 7
Instances: 150 Sum of weights: 150

Attributes

No.		Name
1	□	sepallength
2	□	sepalwidth
3	□	petallength
4	□	petalwidth
5	□	class
6	□	Outlier
7	□	ExtremeValue

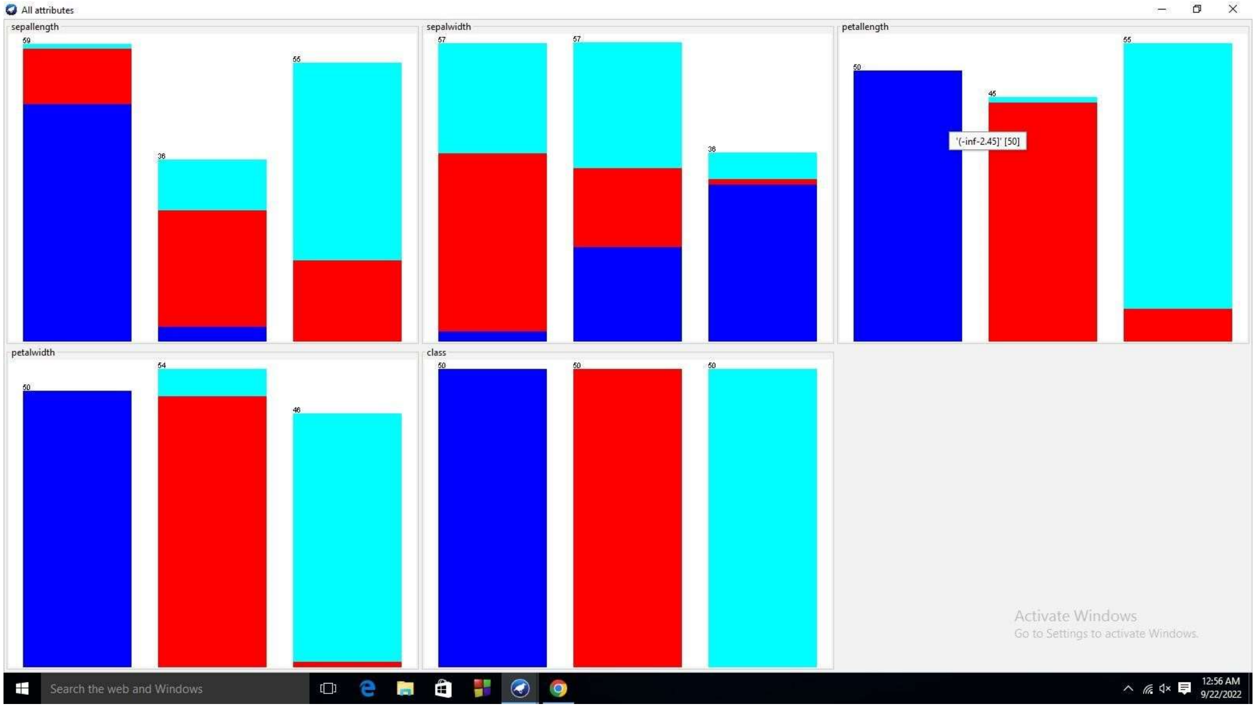
Selected attribute
Name: sepallength Type: Nominal
Missing: 0 (0%) Distinct: 3 Unique: 0 (0%)

No.	Label	Count	Weight
1	'(-inf-5.55]'	59	59
2	'(5.55-6.15]'	36	36
3	'(6.15-inf)'	55	55

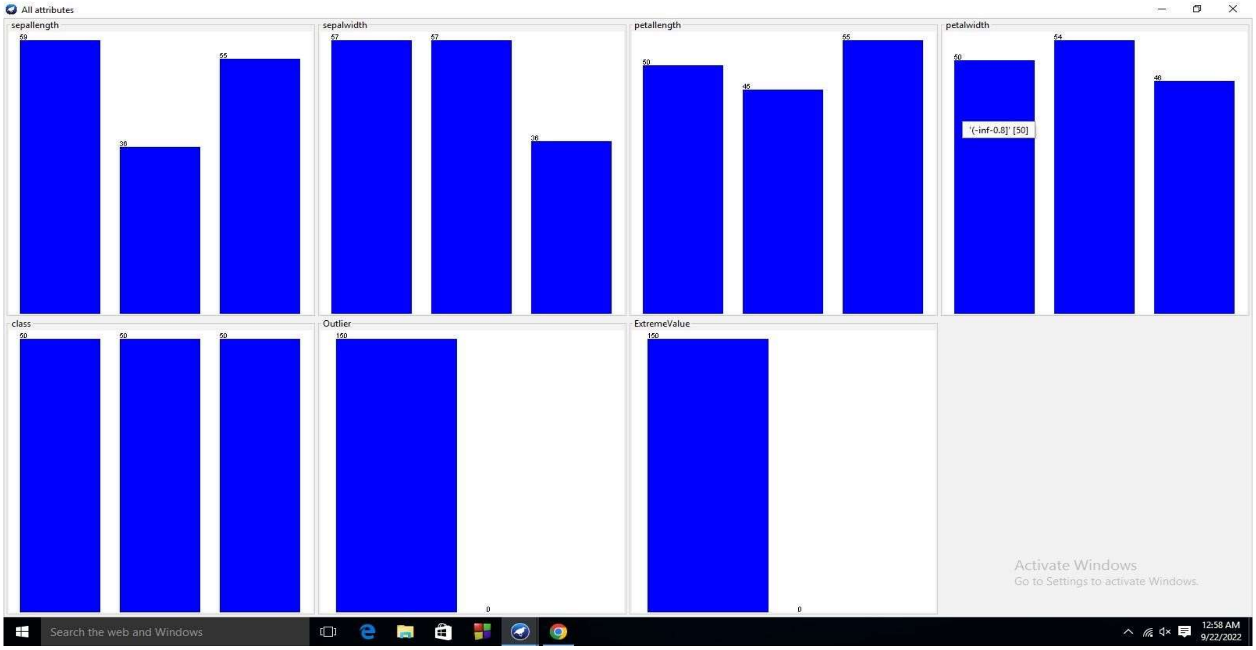
Class: Outlier (Nom)

Status
OK

Pre-processing window of discretize unsupervised window where outlier for first bin is 59 which is the highest.

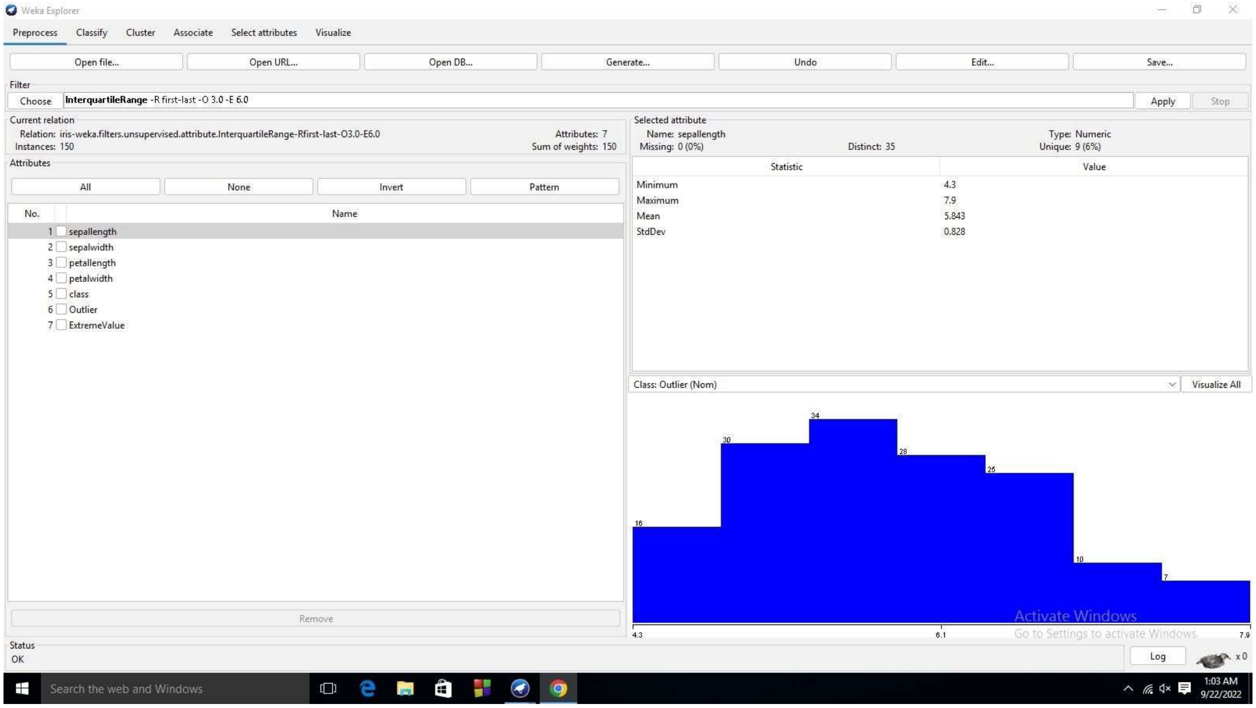


This is stacked column chart of discretized unsupervised learning



Above picture shows the extreme and outlier values of different attributes extreme values for discretize unsupervised learning . Insights from this is that extreme value for petal length is 55 and that of the petal width is 54.Outlier and Extreme values are 150

- **IQR:**



Weka Explorer

Filter: InterquartileRange -R first-last -O 3.0 -E 6.0

Current relation
Relation: iris-weka.filters.unsupervised.attribute.InterquartileRange-Rfirst-last-O3.0-E6.0
Instances: 150
Attributes: 7
Sum of weights: 150

Attributes

No.	Name
1	sepallength
2	sepalwidth
3	petallength
4	petalwidth
5	class
6	Outlier
7	ExtremeValue

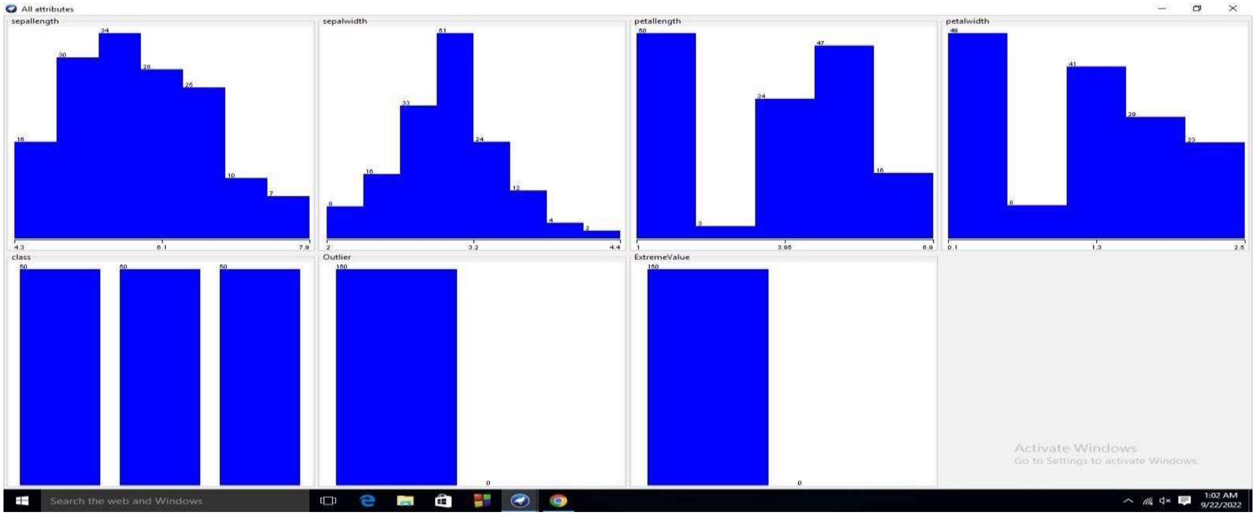
Selected attribute
Name: sepallength
Type: Numeric
Missing: 0 (0%)
Distinct: 35
Unique: 9 (6%)

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

Class: Outlier (Nom)

Status
OK

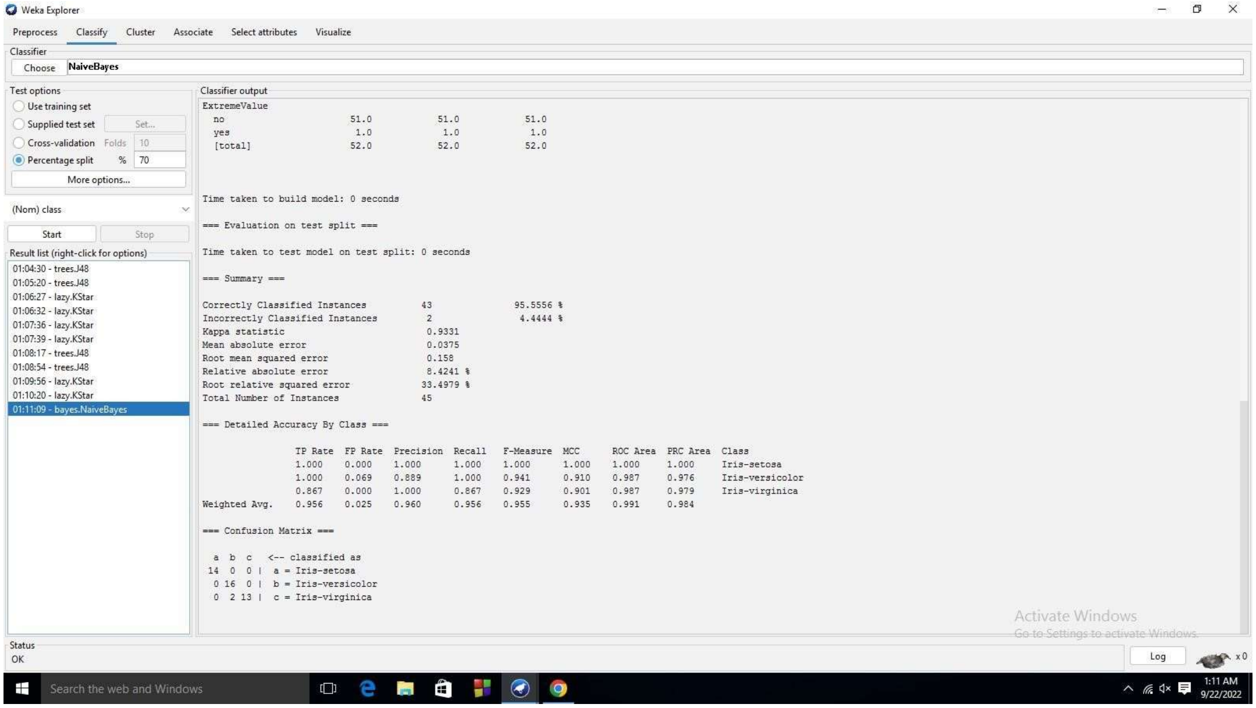
This is the pre-processing window of IQR with sepal length as chosen attribute whose outlier histogram is skewed-right with mode 54 and it has minimum value of 4.3 and maximum value as 7.9 .



This is IQR outlier and extreme histogram visualization of all attributes with sepal length and sepal width being unimodal and petal length and petal width being multi modal.

2. Classification:

- Naïve Bayes



Classifier: NaiveBayes

Test options: ○ Use training set ○ Supplied test set ○ Cross-validation Folds 10 ● Percentage split % 70

(Nom) class

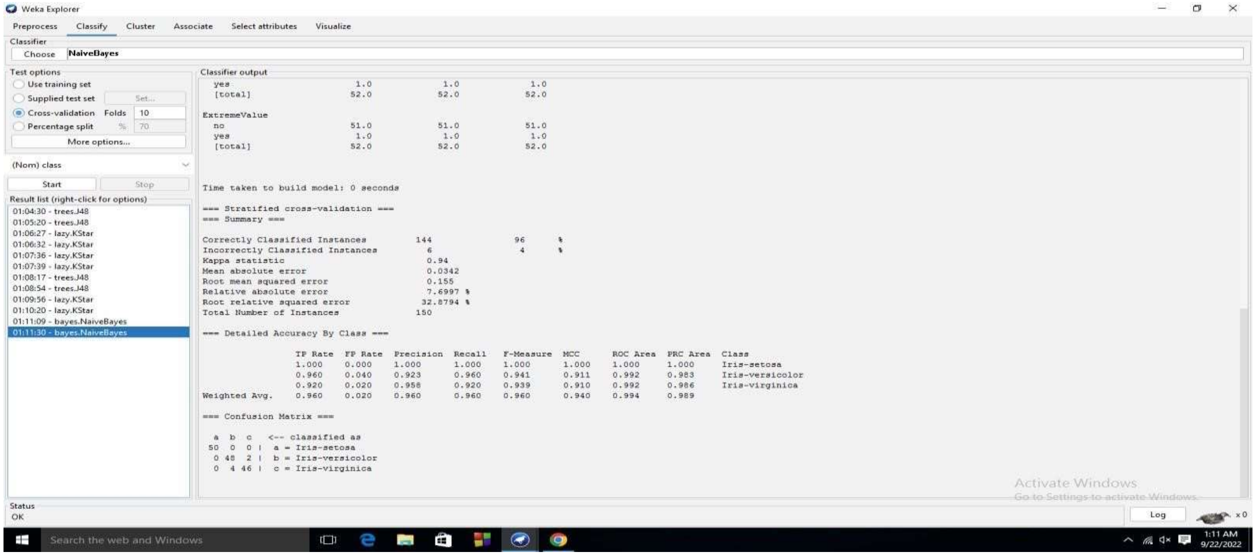
Result list (right-click for options):
01:04:30 - trees.J48
01:05:20 - trees.J48
01:06:27 - lazy.KStar
01:06:32 - lazy.KStar
01:07:36 - lazy.KStar
01:07:39 - lazy.KStar
01:08:17 - trees.J48
01:08:54 - trees.J48
01:09:56 - lazy.KStar
01:10:20 - lazy.KStar
01:11:09 - bayes.NaiveBayes

Classifier output

```
ExtremeValue
  no                  51.0          51.0          51.0
  yes                  1.0           1.0           1.0
  [total]             52.0          52.0          52.0



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          43               95.5556 %
Incorrectly Classified Instances         2                4.4444 %
Kappa statistic                          0.9331
Mean absolute error                      0.0375
Root mean squared error                  0.158
Relative absolute error                  8.4241 %
Root relative squared error             33.4979 %
Total Number of Instances               45

=== 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    0.069    0.889      1.000    0.941      0.910    0.987     0.976     Iris-versicolor
                 0.867    0.000    1.000      0.867    0.929      0.901    0.987     0.979     Iris-virginica
Weighted Avg.    0.956    0.025    0.960      0.956    0.955      0.935    0.991     0.984

=== Confusion Matrix ===

  a  b  c   <-- classified as
 14  0  0 |  a = Iris-setosa
  0 16  0 |  b = Iris-versicolor
  0  2 13 |  c = Iris-virginica
```

Status: OK

This is Naïve Bayes percentage split of 70% where correctly classified instances was 95.556 With mean absolute error as 0.0375



Classifier: NaiveBayes

Test options: ○ Use training set ○ Supplied test set ● Cross-validation Folds 10 ○ Percentage split % 70

(Nom) class

Result list (right-click for options):
01:04:30 - trees.J48
01:05:20 - trees.J48
01:06:27 - lazy.KStar
01:06:32 - lazy.KStar
01:07:36 - lazy.KStar
01:07:39 - lazy.KStar
01:08:17 - trees.J48
01:08:54 - trees.J48
01:09:56 - lazy.KStar
01:10:20 - lazy.KStar
01:11:09 - bayes.NaiveBayes
01:11:30 - bayes.NaiveBayes

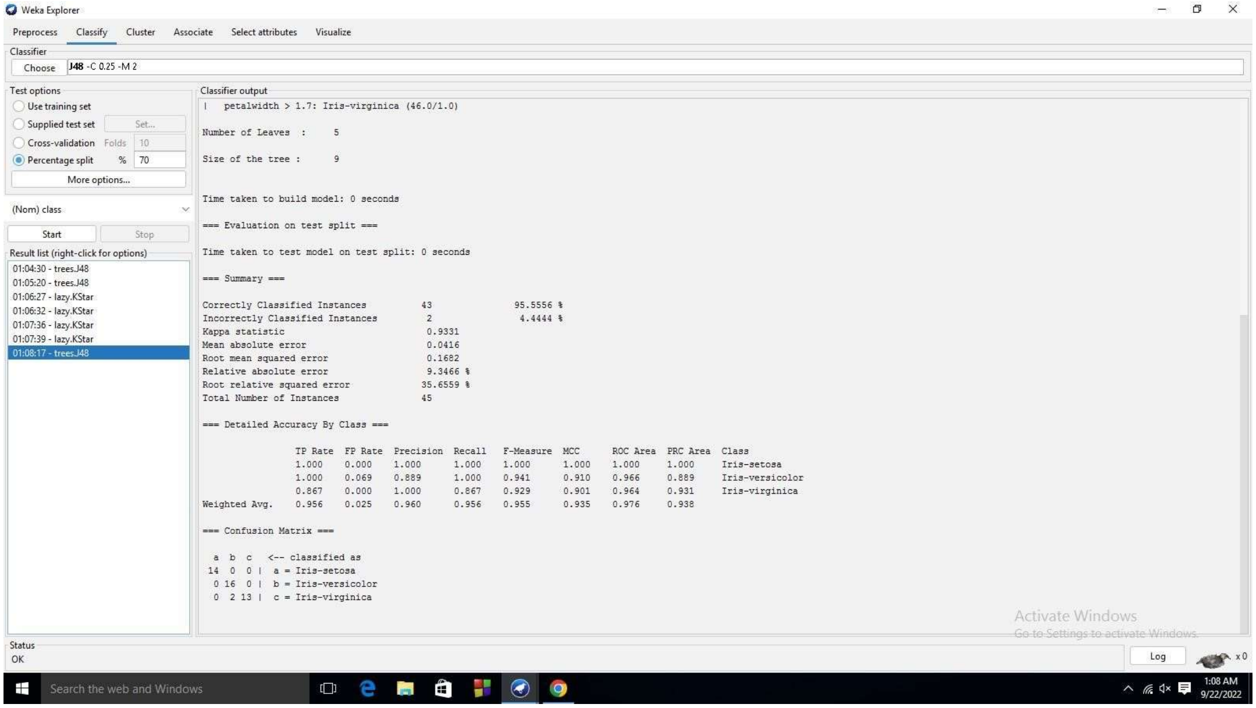
Classifier output

```
  yes                  1.0           1.0           1.0
  [total]             52.0          52.0          52.0

ExtremeValue
  no                  51.0          51.0          51.0
  yes                  1.0           1.0           1.0
  [total]             52.0          52.0          52.0



Time taken to build model: 0 seconds

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

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.0342
Root mean squared error                  0.155
Relative absolute error                  7.6997 %
Root relative squared error             32.8794 %
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    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Iris-setosa
                 0.960    0.040    0.923      0.960    0.941      0.911    0.992     0.983     Iris-versicolor
                 0.920    0.020    0.958      0.920    0.939      0.910    0.992     0.986     Iris-virginica
Weighted Avg.    0.960    0.020    0.960      0.960    0.960      0.940    0.994     0.989

=== Confusion Matrix ===

  a  b  c   <-- classified as
 50  0  0 |  a = Iris-setosa
  0 48  2 |  b = Iris-versicolor
  0  4 46 |  c = Iris-virginica
```

Status: OK

This is naïve Bayes cross validation with 10 folds in which correctly classified instances has increased to 96% with mean absolute error reduced to 0.342.

- J48 (Example of Decision Tree):



Classifier: J48 -C 0.25 -M 2

Test options:
○ Use training set
○ Supplied test set
○ Cross-validation Folds 10
● Percentage split % 70

Result list (right-click for options):
01:04:30 - trees.J48
01:05:20 - trees.J48
01:06:27 - lazy.KStar
01:06:32 - lazy.KStar
01:07:36 - lazy.KStar
01:07:39 - lazy.KStar
01:08:17 - trees.J48

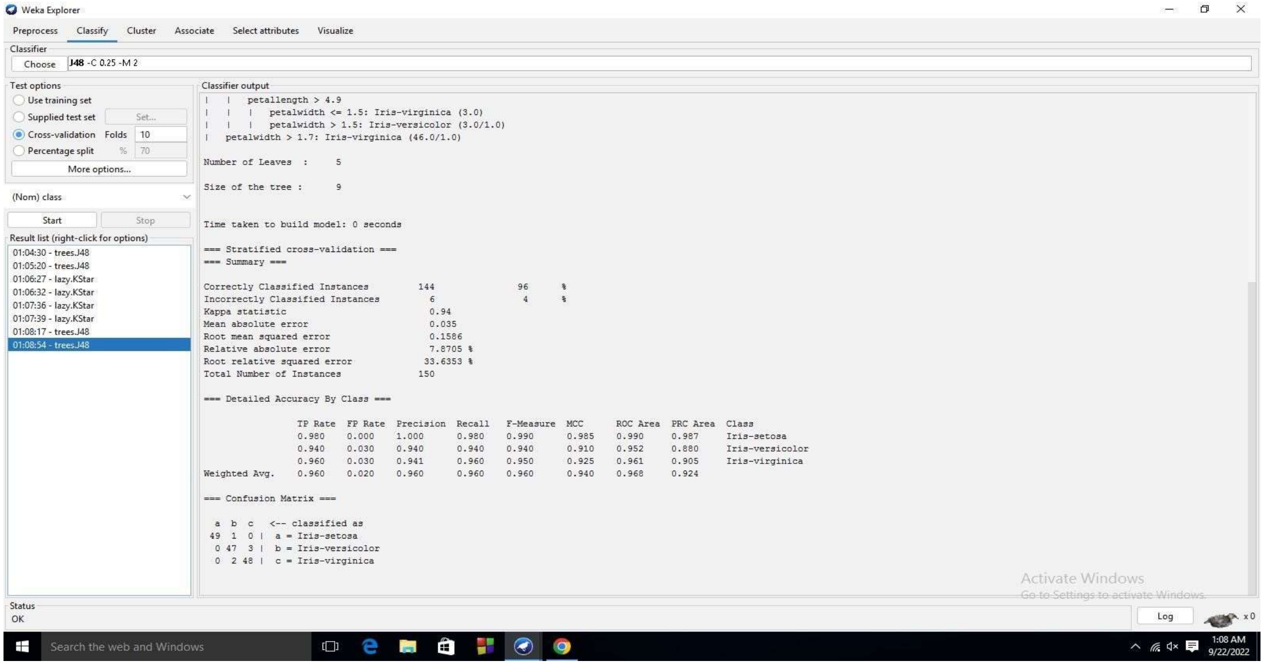
Classifier output:

```
|   petalwidth > 1.7: Iris-virginica (46.0/1.0)

Number of Leaves  :     5

Size of the tree :      9


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          43               95.5556 %
Incorrectly Classified Instances         2                4.4444 %
Kappa statistic                          0.9331
Mean absolute error                      0.0416
Root mean squared error                  0.1682
Relative absolute error                  9.3466 %
Root relative squared error             35.6559 %
Total Number of Instances               45

=== 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    0.069    0.889      1.000    0.941      0.910    0.966     0.889     Iris-versicolor
                 0.867    0.000    1.000      0.867    0.929      0.901    0.964     0.931     Iris-virginica
Weighted Avg.    0.956    0.025    0.960      0.956    0.955      0.935    0.976     0.938

=== Confusion Matrix ===

  a  b  c   <-- classified as
 14  0  0 |  a = Iris-setosa
  0 16  0 |  b = Iris-versicolor
  0  2 13 |  c = Iris-virginica
```

Status: OK

1:08 AM 9/22/2022

This is J48 performance split of 70/30 with correctly classified instances of 95.556 and mean absolute error of 0.0416



Classifier: J48 -C 0.25 -M 2

Test options:
○ Use training set
○ Supplied test set
● Cross-validation Folds 10
○ Percentage split % 70

Result list (right-click for options):
01:04:30 - trees.J48
01:05:20 - trees.J48
01:06:27 - lazy.KStar
01:06:32 - lazy.KStar
01:07:36 - lazy.KStar
01:07:39 - lazy.KStar
01:08:17 - trees.J48
01:08:54 - trees.J48

Classifier output:

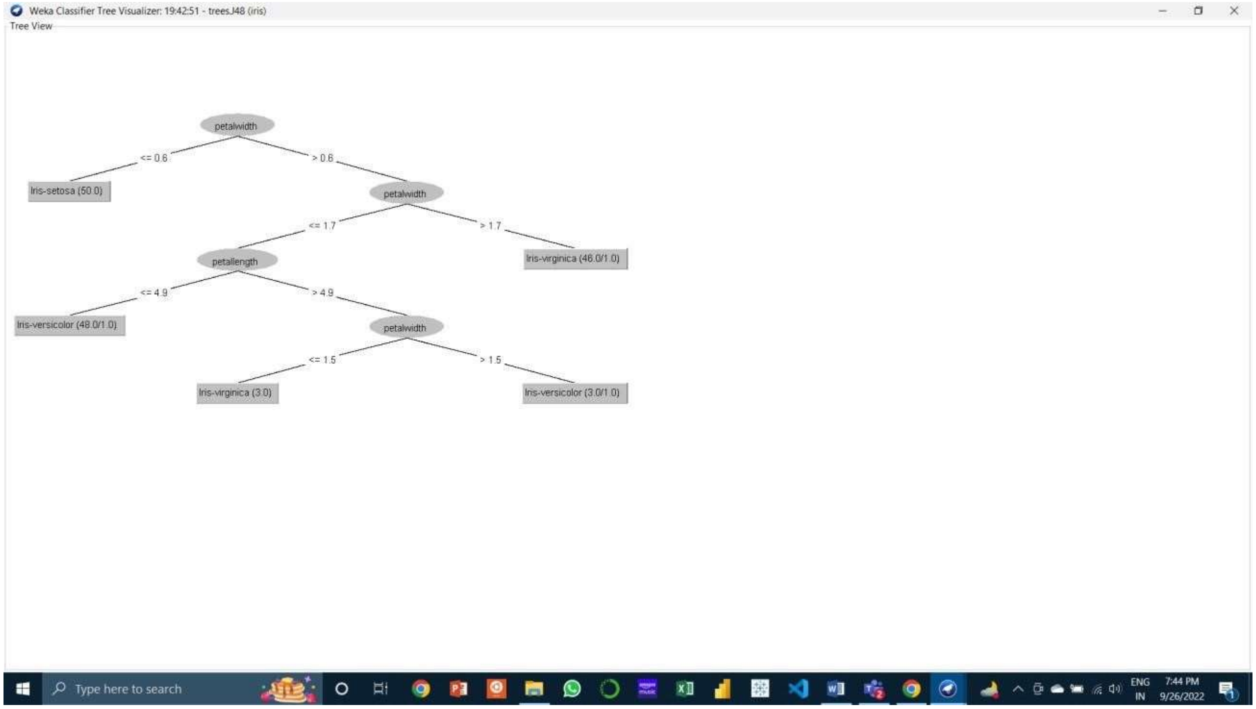
```
|   |   petallength > 4.9
|   |   |   petalwidth <= 1.5: Iris-virginica (3.0)
|   |   |   petalwidth > 1.5: Iris-versicolor (3.0/1.0)
|   petalwidth > 1.7: Iris-virginica (46.0/1.0)

Number of Leaves  :     5

Size of the tree :      9


Time taken to build model: 0 seconds

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

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
Total Number of Instances              150

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.980    0.000    1.000      0.980    0.990      0.985    0.990     0.987     Iris-setosa
                 0.940    0.030    0.940      0.940    0.940      0.910    0.952     0.880     Iris-versicolor
                 0.960    0.030    0.941      0.960    0.950      0.925    0.961     0.905     Iris-virginica
Weighted Avg.    0.960    0.020    0.960      0.960    0.960      0.940    0.968     0.924

=== Confusion Matrix ===

  a  b  c   <-- classified as
 49  1  0 |  a = Iris-setosa
  0 47  3 |  b = Iris-versicolor
  0  2 48 |  c = Iris-virginica
```

Status: OK

1:08 AM 9/22/2022

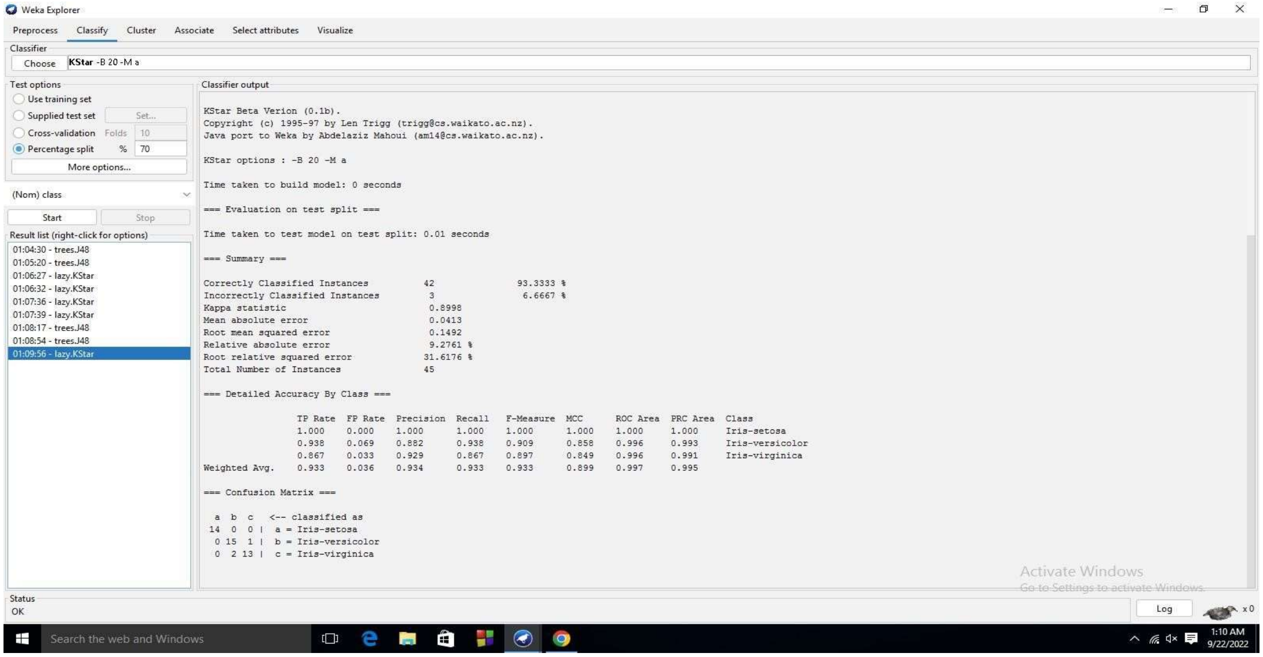
This is J48 cross validation with 10 folds and correctly classified instances at 96% and mean absolute error as 0.035

A.Y. 2022-2023



This is the visualization of J48 tree

- **K-star:**



Classifier
Choose KStar -B 20 -M a

Test options
○ Use training set
○ Supplied test set Set...
○ Cross-validation Folds 10
● Percentage split % 70

(Nom) class

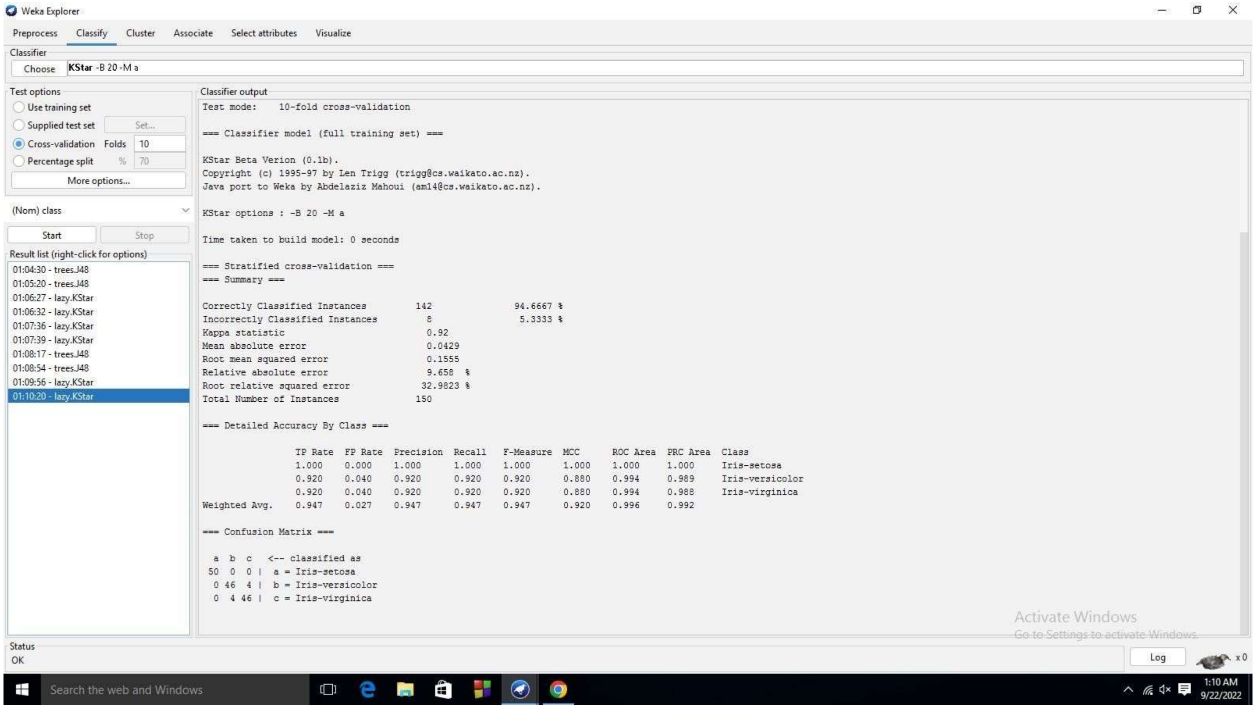
Result list (right-click for options)
01:04:30 - trees.J48
01:05:20 - trees.J48
01:06:27 - lazy.KStar
01:06:32 - lazy.KStar
01:07:36 - lazy.KStar
01:07:39 - lazy.KStar
01:08:17 - trees.J48
01:08:54 - trees.J48
01:09:56 - lazy.KStar

Status
OK

Classifier output

```
KStar Beta Verion (0.1b).
Copyright (c) 1995-97 by Len Trigg (trigg@cs.waikato.ac.nz).
Java port to Weka by Abdelaziz Mahoui (am14@cs.waikato.ac.nz).

KStar options : -B 20 -M a

Time taken to build model: 0 seconds

=== Evaluation on test split ===

Time taken to test model on test split: 0.01 seconds

=== Summary ===

Correctly Classified Instances          42               93.3333 %
Incorrectly Classified Instances         3                6.6667 %
Kappa statistic                          0.8998
Mean absolute error                      0.0413
Root mean squared error                  0.1492
Relative absolute error                  9.2761 %
Root relative squared error             31.6176 %
Total Number of Instances               45

=== 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
                 0.938    0.069    0.882      0.938    0.909      0.858    0.996     0.993     Iris-versicolor
                 0.867    0.033    0.929      0.867    0.897      0.849    0.996     0.991     Iris-virginica
Weighted Avg.    0.933    0.036    0.934      0.933    0.933      0.899    0.997     0.995

=== Confusion Matrix ===

  a  b  c   <-- classified as
 14  0  0 |  a = Iris-setosa
  0 15  1 |  b = Iris-versicolor
  0  2 13 |  c = Iris-virginica
```

This is K-Star percentage split 70/30 with correctly classified instances at 93.33% with mean absolute error being 0.0413



Classifier: KStar -B 20 -M a

Test options
○ Use training set
○ Supplied test set
● Cross-validation Folds 10
○ Percentage split % 70

(Nom) class

Result list (right-click for options)

```
01:04:30 - trees.J48
01:05:20 - trees.J48
01:06:27 - lazy.KStar
01:06:32 - lazy.KStar
01:07:36 - lazy.KStar
01:07:39 - lazy.KStar
01:08:17 - trees.J48
01:08:54 - trees.J48
01:09:56 - lazy.KStar
01:10:20 - lazy.KStar
```

Classifier output

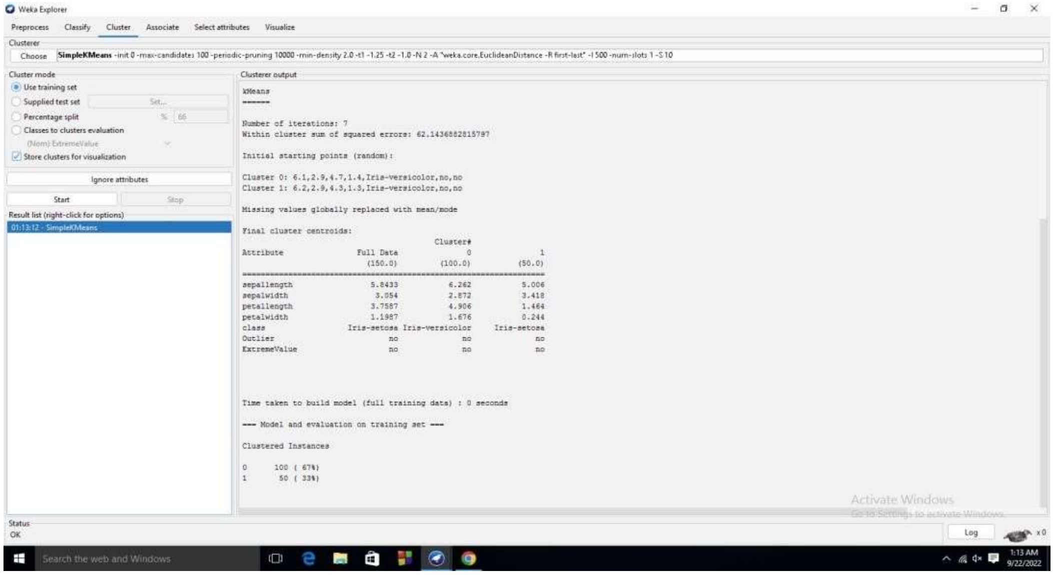
```
Test mode:    10-fold cross-validation

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

KStar Beta Verion (0.1b).
Copyright (c) 1995-97 by Len Trigg (trigg@cs.waikato.ac.nz).
Java port to Weka by Abdelaziz Mahoui (am14@cs.waikato.ac.nz).

KStar options : -B 20 -M a

Time taken to build model: 0 seconds

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

Correctly Classified Instances         142               94.6667 %
Incorrectly Classified Instances         8                5.3333 %
Kappa statistic                          0.92
Mean absolute error                      0.0429
Root mean squared error                  0.1555
Relative absolute error                  9.658  %
Root relative squared error             32.9823 %
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    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Iris-setosa
                 0.920    0.040    0.920      0.920    0.920      0.880    0.994     0.989     Iris-versicolor
                 0.920    0.040    0.920      0.920    0.920      0.880    0.994     0.988     Iris-virginica
Weighted Avg.    0.947    0.027    0.947      0.947    0.947      0.920    0.996     0.992

=== Confusion Matrix ===

  a  b  c   <-- classified as
 50  0  0 |  a = Iris-setosa
  0 46  4 |  b = Iris-versicolor
  0  4 46 |  c = Iris-virginica
```

Status
OK

This is the K-Star cross-validation with 10 folds having correctly classified instances at 94.67% and mean absolute error being 0.0429

3. Clustering:

- **K-Means**



Clusterer: **SimpleKMeans** -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10

Cluster mode
● Use training set
○ Supplied test set
○ Percentage split % 66
○ Classes to clusters evaluation (Nom) ExtremeValue
☑ Store clusters for visualization

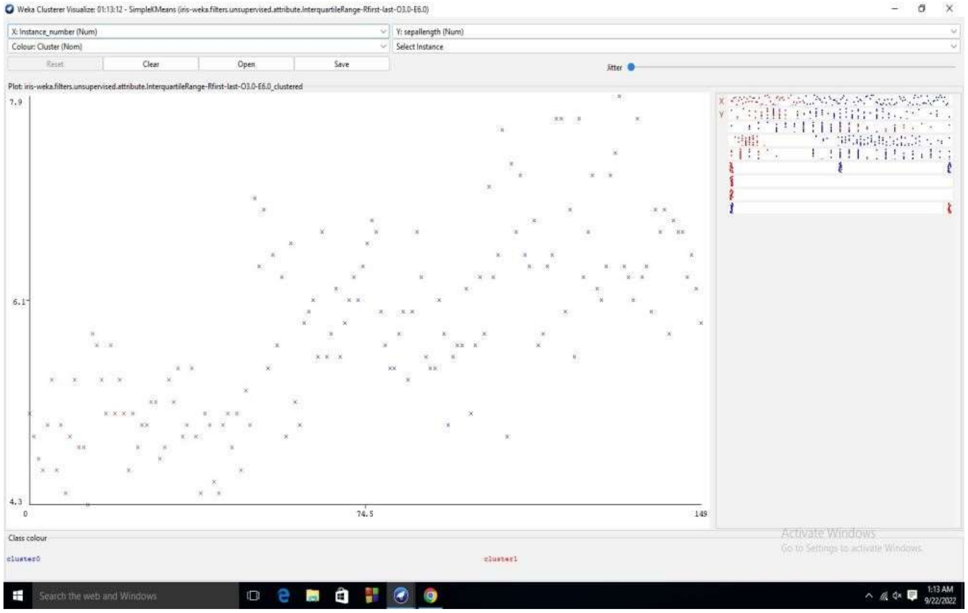
Result list (right-click for options)
01:13:12 - SimpleKMeans

Clusterer output

```
kMeans
======

Number of iterations: 7
Within cluster sum of squared errors: 62.1436882815797

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4,Iris-versicolor,no,no
Cluster 1: 6.2,2.9,4.3,1.3,Iris-versicolor,no,no

Missing values globally replaced with mean/mode

Final cluster centroids:
                                      Cluster#
Attribute          Full Data                 0            1
                     (150.0)           (100.0)       (50.0)
============================================================
sepallength           5.8433             6.262        5.006
sepalwidth             3.054             2.872        3.418
petallength           3.7587             4.906        1.464
petalwidth            1.1987             1.676        0.244
class            Iris-setosa Iris-versicolor   Iris-setosa
Outlier                   no                no           no
ExtremeValue              no                no           no



Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      100 ( 67%)
1       50 ( 33%)
```

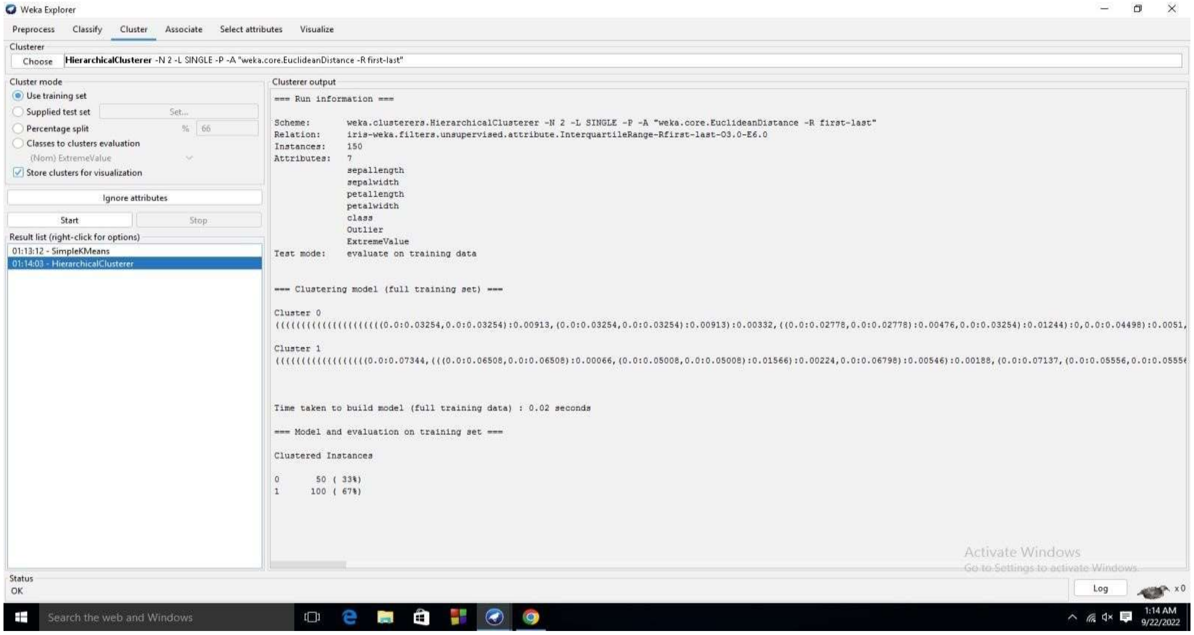
Status
OK

This is output of the model of K-Means with cluster 0 having 67% and cluster 1 having 33%


Weka Clusterer Visualize: 01:13:12 - SimpleKMeans (iris-weka.filters.unsupervised.attribute.InterquartileRange-Rfirst-last-O3.0-E6.0)
X: Instance_number (Num)
Y: sepallength (Num)
Colour: Cluster (Nom)
Select Instance
Reset
Clear
Open
Save
Jitter
Plot: iris-weka.filters.unsupervised.attribute.InterquartileRange-Rfirst-last-O3.0-E6.0_clustered
7.9
6.1
4.3
0
74.5
149
X
Y
Class colour
cluster0
cluster1

This is the scatter plot for K-Means cluster assignment where we can conclude that there are two clusters one at the beginning and second in the middle also there is a upward trend.

- Hierarchal Cluster:



Clusterer: **HierarchicalClusterer** -N 2 -L SINGLE -P -A "weka.core.EuclideanDistance -R first-last"

Cluster mode
● Use training set
○ Supplied test set
○ Percentage split % 66
○ Classes to clusters evaluation
(Nom) ExtremeValue
☑ Store clusters for visualization

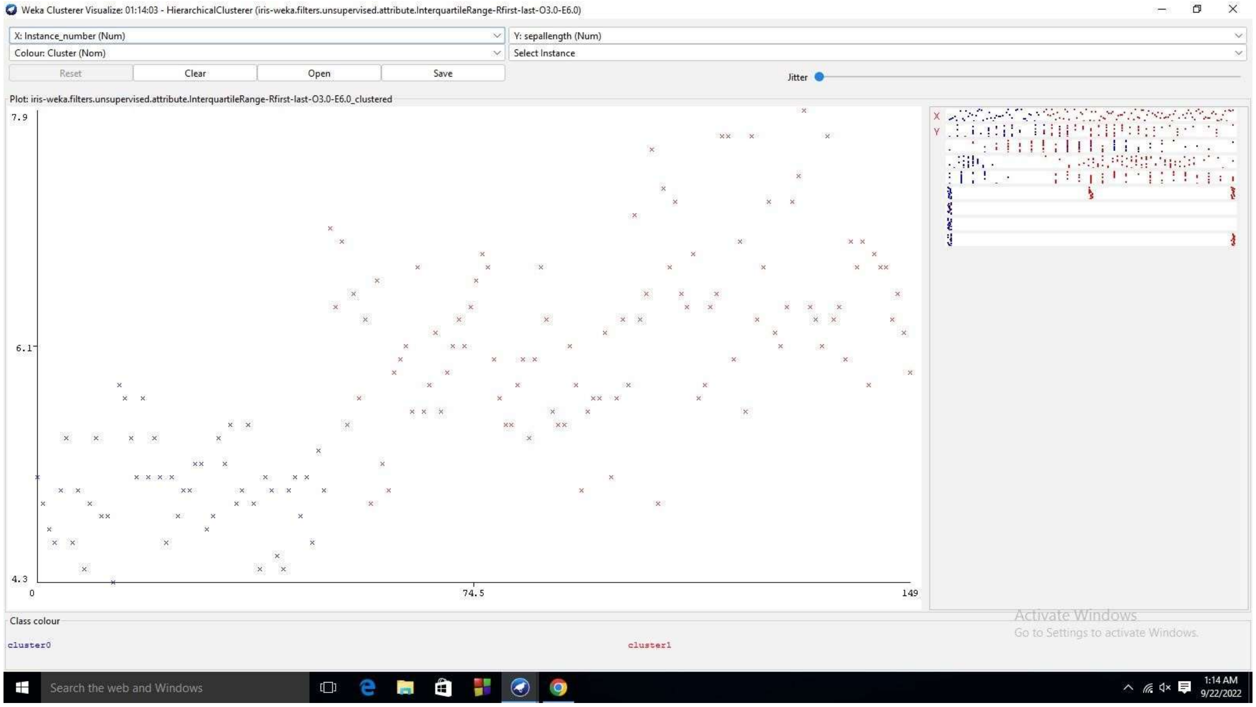
Result list (right-click for options)
01:13:12 - SimpleKMeans
01:14:03 - HierarchicalClusterer

Clusterer output

```
=== Run information ===

Scheme:       weka.clusterers.HierarchicalClusterer -N 2 -L SINGLE -P -A "weka.core.EuclideanDistance -R first-last"
Relation:     iris-weka.filters.unsupervised.attribute.InterquartileRange-Rfirst-last-O3.0-E6.0
Instances:    150
Attributes:   7
              sepallength
              sepalwidth
              petallength
              petalwidth
              class
              Outlier
              ExtremeValue
Test mode:    evaluate on training data


=== Clustering model (full training set) ===

Cluster 0
((((((((((((((((((((0.0:0.03254,0.0:0.03254):0.00913,(0.0:0.03254,0.0:0.03254):0.00913):0.00332,((0.0:0.02778,0.0:0.02778):0.00476,0.0:0.03254):0.01244):0,0.0:0.04498):0.0051,

Cluster 1
(((((((((((((((((0.0:0.07344,(((0.0:0.06508,0.0:0.06508):0.00066,(0.0:0.05008,0.0:0.05008):0.01566):0.00224,0.0:0.06798):0.00546):0.00188,(0.0:0.07137,(0.0:0.05556,0.0:0.0555



Time taken to build model (full training data) : 0.02 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       50 ( 33%)
1      100 ( 67%)
```

Status
OK

The percentage split of cluster is same as above.


Weka Clusterer Visualize: 01:14:03 - HierarchicalClusterer (iris-weka.filters.unsupervised.attribute.InterquartileRange-Rfirst-last-O3.0-E6.0)
Reset
Clear
Open
Save
Jitter
7.9
6.1
4.3
0
74.5
149
X
Y

X: Instance_number (Num)
Y: sepallength (Num)
Colour: Cluster (Nom)
Select Instance

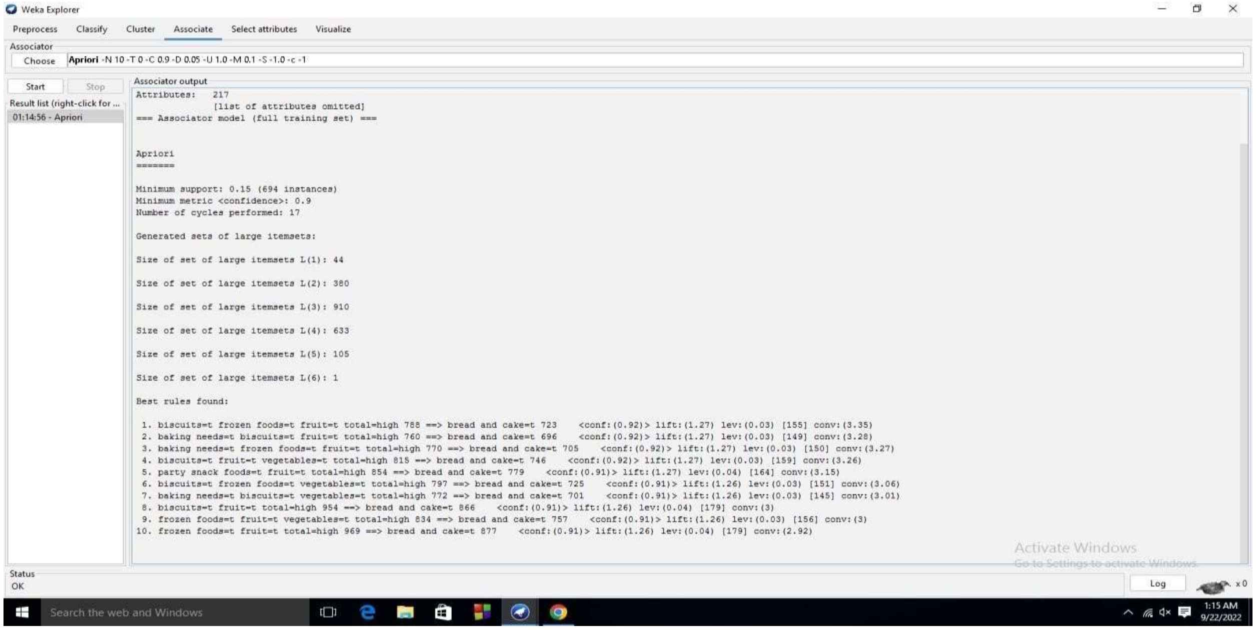
Plot: iris-weka.filters.unsupervised.attribute.InterquartileRange-Rfirst-last-O3.0-E6.0_clustered

Class colour
cluster0
cluster1

This is scatter plot of hierarchal cluster assignment. According to me, data is evenly distributed with upward trend and two cluster forming one at the start and other at middle.

4. Associate Rule:

Super market associate market basket analysis



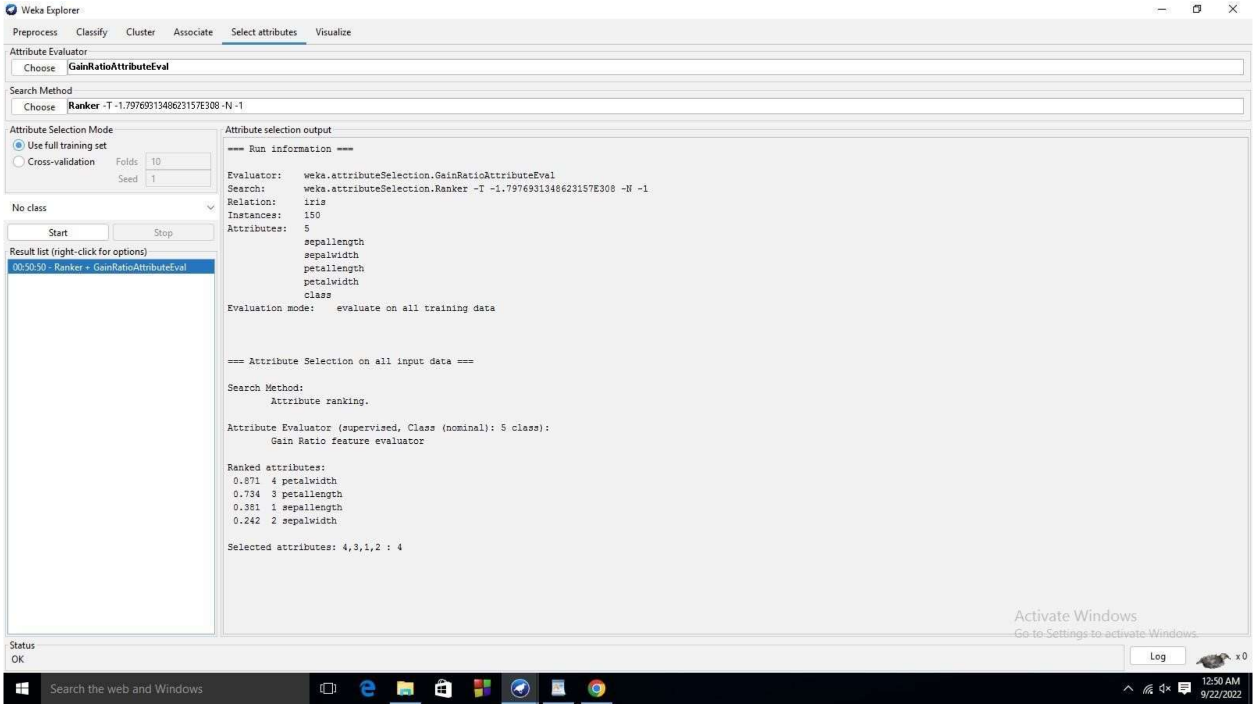
Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
Attributes:   217
              [list of attributes omitted]
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.15 (694 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 44

Size of set of large itemsets L(2): 380

Size of set of large itemsets L(3): 910

Size of set of large itemsets L(4): 633

Size of set of large itemsets L(5): 105

Size of set of large itemsets L(6): 1

Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    <conf:(0.92)> lift:(1.27) lev:(0.03) [155] conv:(3.35)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    <conf:(0.92)> lift:(1.27) lev:(0.03) [149] conv:(3.28)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    <conf:(0.92)> lift:(1.27) lev:(0.03) [150] conv:(3.27)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    <conf:(0.92)> lift:(1.27) lev:(0.03) [159] conv:(3.26)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    <conf:(0.91)> lift:(1.27) lev:(0.04) [164] conv:(3.15)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    <conf:(0.91)> lift:(1.26) lev:(0.03) [151] conv:(3.06)
 7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    <conf:(0.91)> lift:(1.26) lev:(0.03) [145] conv:(3.01)
 8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(3)
 9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    <conf:(0.91)> lift:(1.26) lev:(0.03) [156] conv:(3)
10. frozen foods=t fruit=t total=high 969 ==> bread and cake=t 877    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(2.92)
```

Confidence level of top 10 rules is either 0.92 or 0.91. We found that there is high relation between biscuits, frozen foods and fruits and many more.

5. Select Attributes:

- Gain Ratio:



Attribute Evaluator: GainRatioAttributeEval

Search Method: Ranker -T -1.7976931348623157E308 -N -1

Attribute Selection Mode: ● Use full training set ○ Cross-validation Folds 10 Seed 1

No class

Result list (right-click for options): 00:50:50 - Ranker + GainRatioAttributeEval

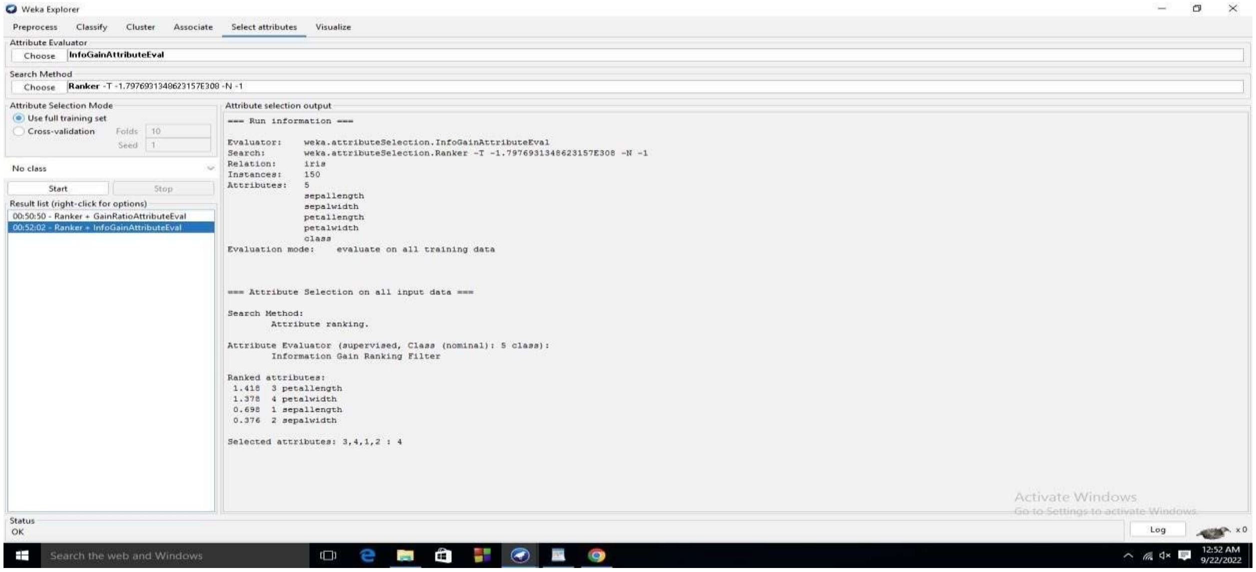
Attribute selection output

```
=== Run information ===

Evaluator:    weka.attributeSelection.GainRatioAttributeEval
Search:       weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation:     iris
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
              class
Evaluation mode:    evaluate on all training data



=== Attribute Selection on all input data ===

Search Method:
        Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 5 class):
        Gain Ratio feature evaluator

Ranked attributes:
 0.871  4 petalwidth
 0.734  3 petallength
 0.381  1 sepallength
 0.242  2 sepalwidth

Selected attributes: 4,3,1,2 : 4
```

Status: OK

Petal width has the highest rank with gain ratio 0.871 and sepal width has the lowest rank with gain ratio 0.242

ii. Info Gain Ratio



Attribute Evaluator: InfoGainAttributeEval

Search Method: Ranker -T -1.7976931348623157E308 -N -1

Attribute Selection Mode: ● Use full training set ○ Cross-validation Folds 10 Seed 1

No class

Result list (right-click for options):
00:50:50 - Ranker + GainRatioAttributeEval
00:52:02 - Ranker + InfoGainAttributeEval

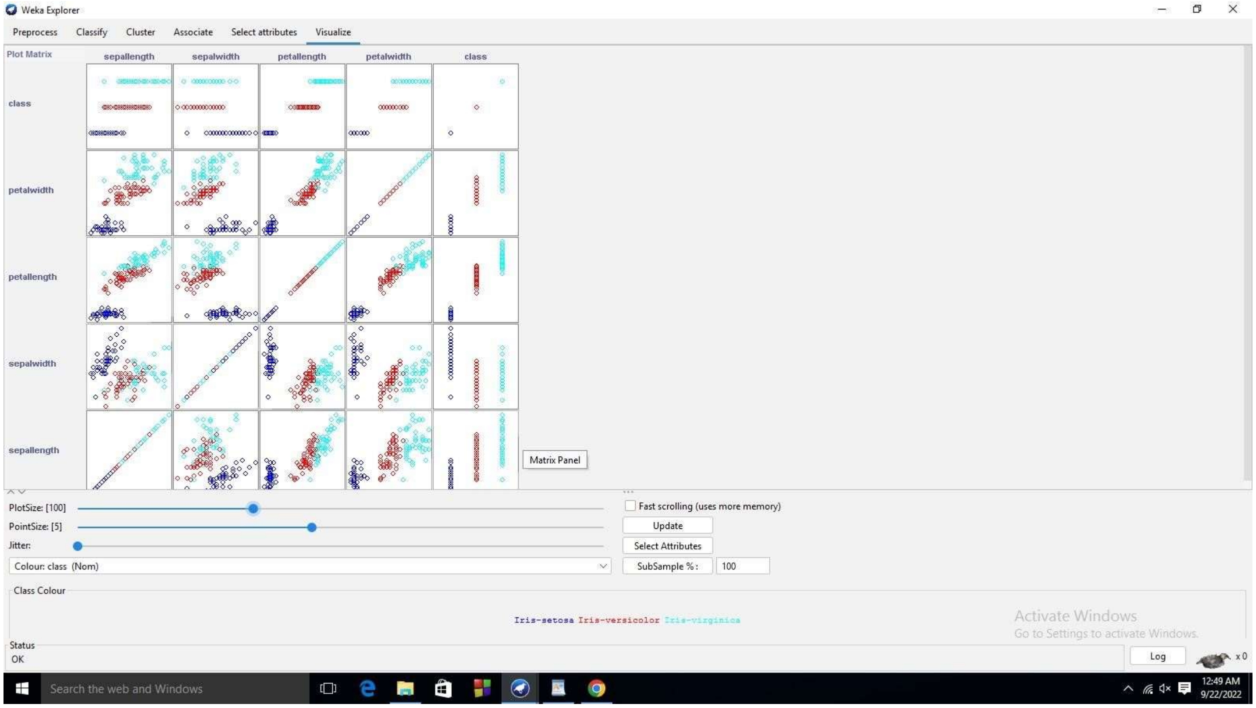
Attribute selection output

```
=== Run information ===

Evaluator:    weka.attributeSelection.InfoGainAttributeEval
Search:       weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation:     iris
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
              class
Evaluation mode:    evaluate on all training data



=== Attribute Selection on all input data ===

Search Method:
        Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 5 class):
        Information Gain Ranking Filter

Ranked attributes:
 1.418  3 petallength
 1.378  4 petalwidth
 0.698  1 sepallength
 0.376  2 sepalwidth

Selected attributes: 3,4,1,2 : 4
```

Status: OK

Petal length has the highest info gain with 1.418 and sepal width with least of 0.376

6. Visualization:



PlotSize: [100]
PointSize: [5]
Jitter:
Colour: class (Nom)
□ Fast scrolling (uses more memory)
SubSample % : 100
Status
OK

Petal width and Petal length are related by upward trendline
For lower value of petal-length we find values of sepal width to be more and we find it to be downward line.
Sepal length and Sepal Width are closely related with all the clusters in close proximity.

Conclusion:

J48, Naïve Bayes had highest correct classified instances with 96%.
Confidence level of top rules was around 0.92