

Nama : Ferza Reyaldi
NIM : 09021281924060
Kelas : TI REGULER
Mata Kuliah : Data Mining

Data Mining with Weka

1.1 Introduction

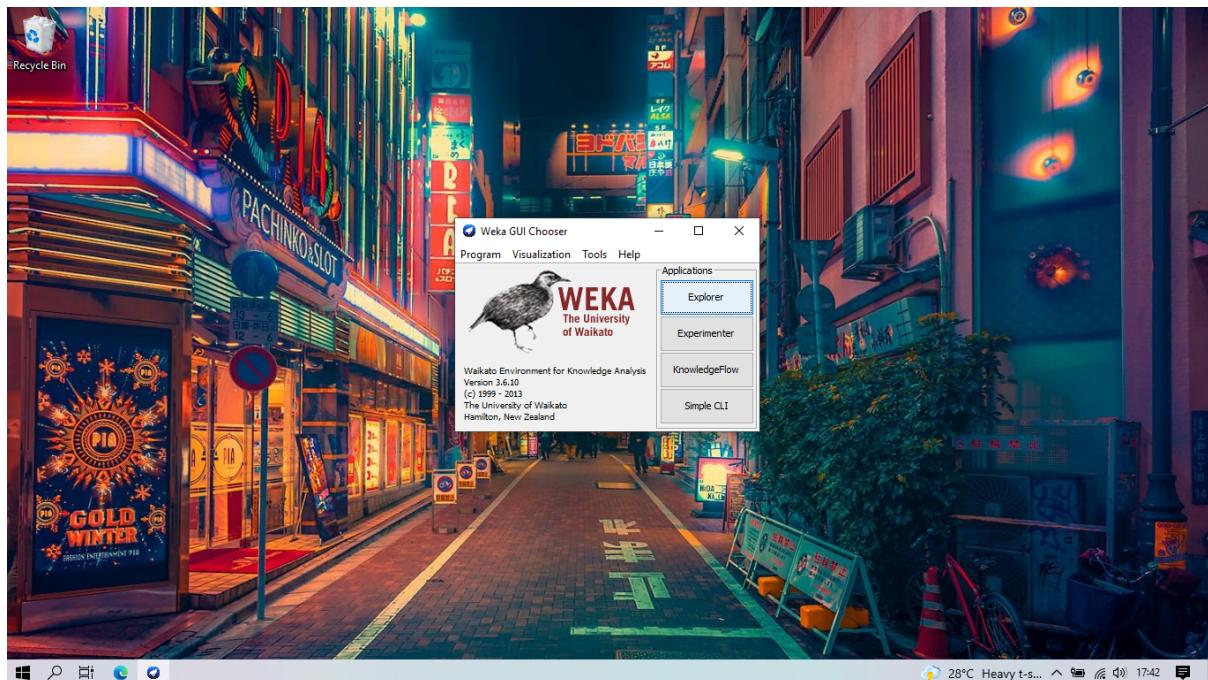
Data Mining berperan mengubah data mentah tersebut menjadi informasi yang lebih berguna dan bisa dipakai untuk memprediksi apa yang akan terjadi berikutnya dan bisa dipakai di dunia nyata. Dan sering dipakai di dunia nyata misal di supermarket untuk membuat harga khusus untuk tiap pembeli.

Jika dibandingkan antara Machine Learning dan Data Mining, Data Mining merupakan aplikasi yang digunakan, sedangkan Machine Learning adalah algoritma yang digunakan untuk melakukan Data Mining.

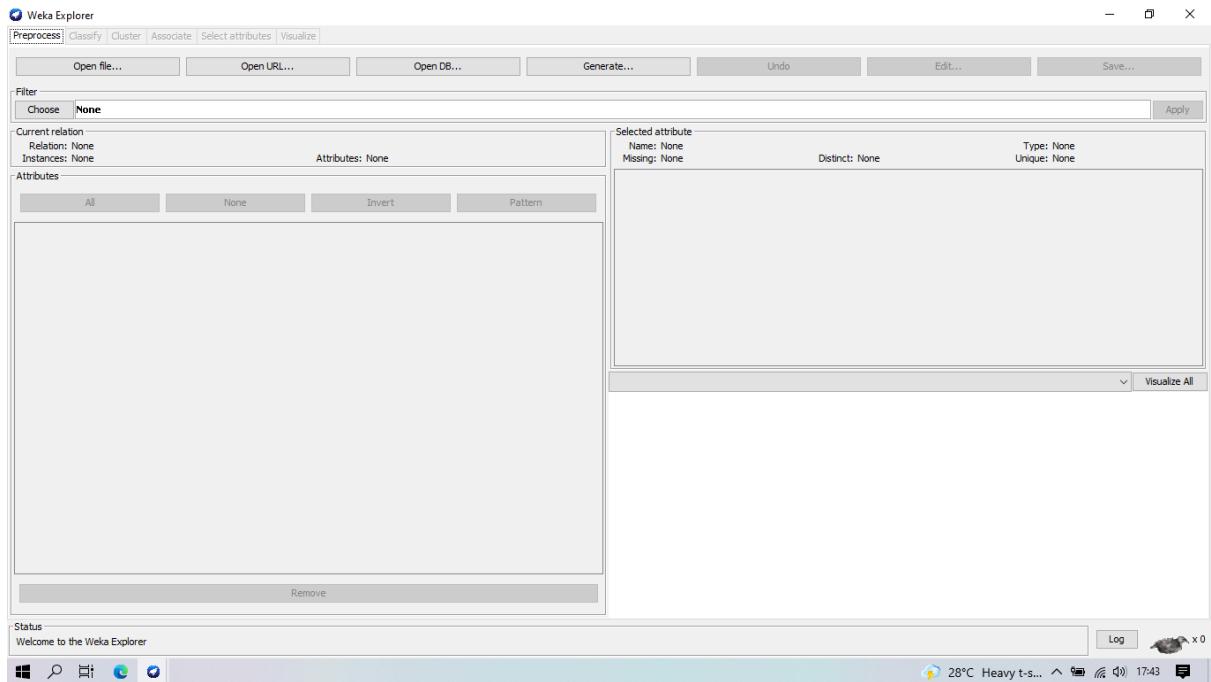
Dalam melakukan data mining, terdapat software yang khusus digunakan untuk mempermudah proses data mining, yaitu WEKA. WEKA (Waikato Environment for Knowledge Analysis) adalah data mining toolkit atau workbench yang bisa dijalankan di Windows, Mac, atau bahkan Linux yang bisa didownload gratis karna bersifat open source software.

1.2 Exploring the Explorer

Buka Aplikasi Weka, kemudian muncul tampilan awal aplikasi Weka.

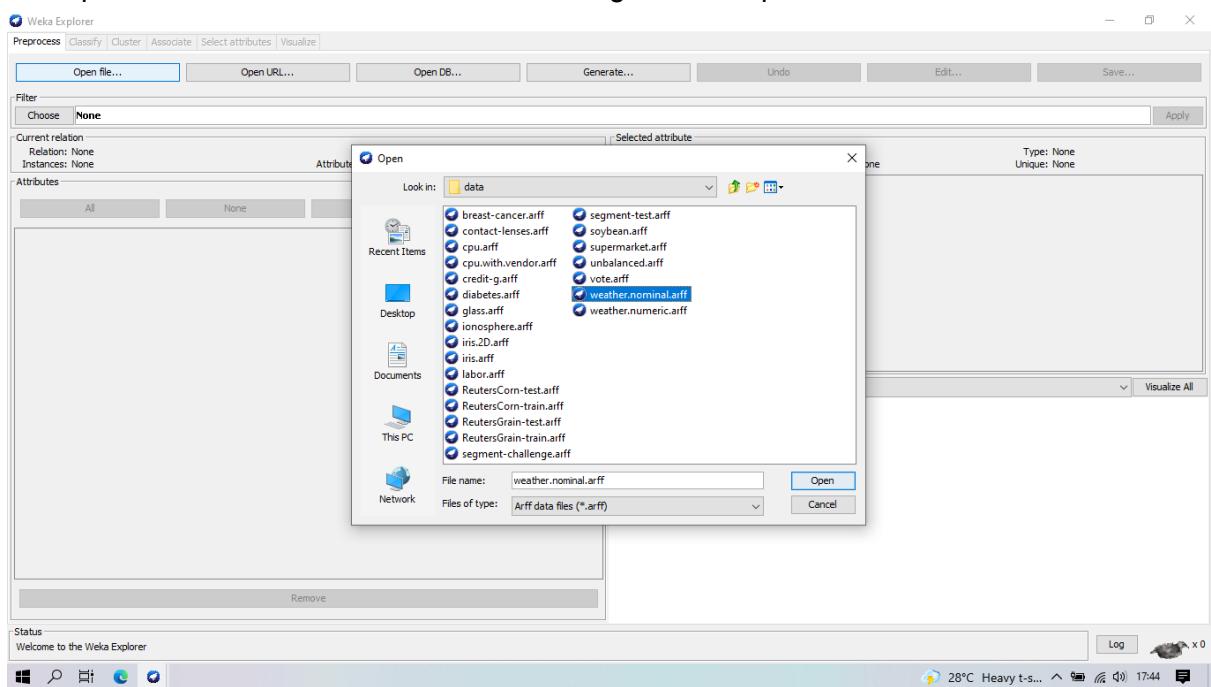


Klik Explorer, sehingga muncul tampilan Explorer.

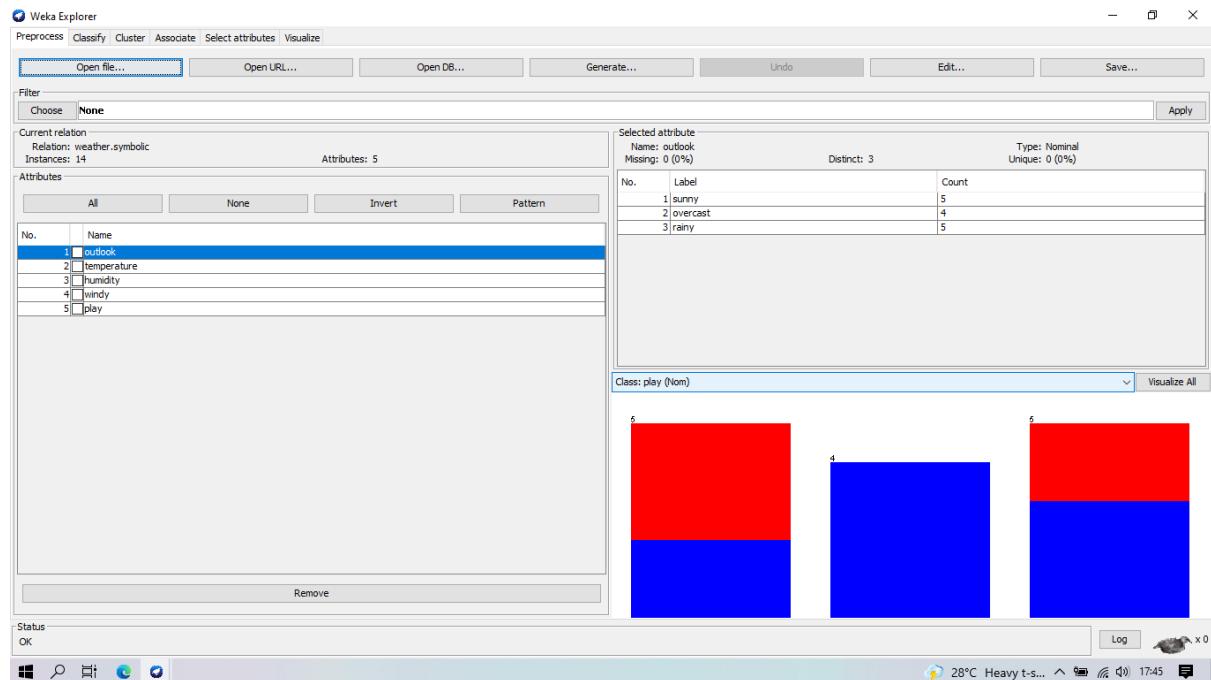


1.3 Exploring Datasets

Klik Open File untuk menambah dataset. Sebagai contoh, pilih weather-nominal.arff.



Klik Open, kemudian muncul tampilan berupa atribut, label pada setiap atribut, dan visualisasinya.



File data set berformat arff, buka file tersebut untuk melihat struktur isi dari file dataset tersebut.

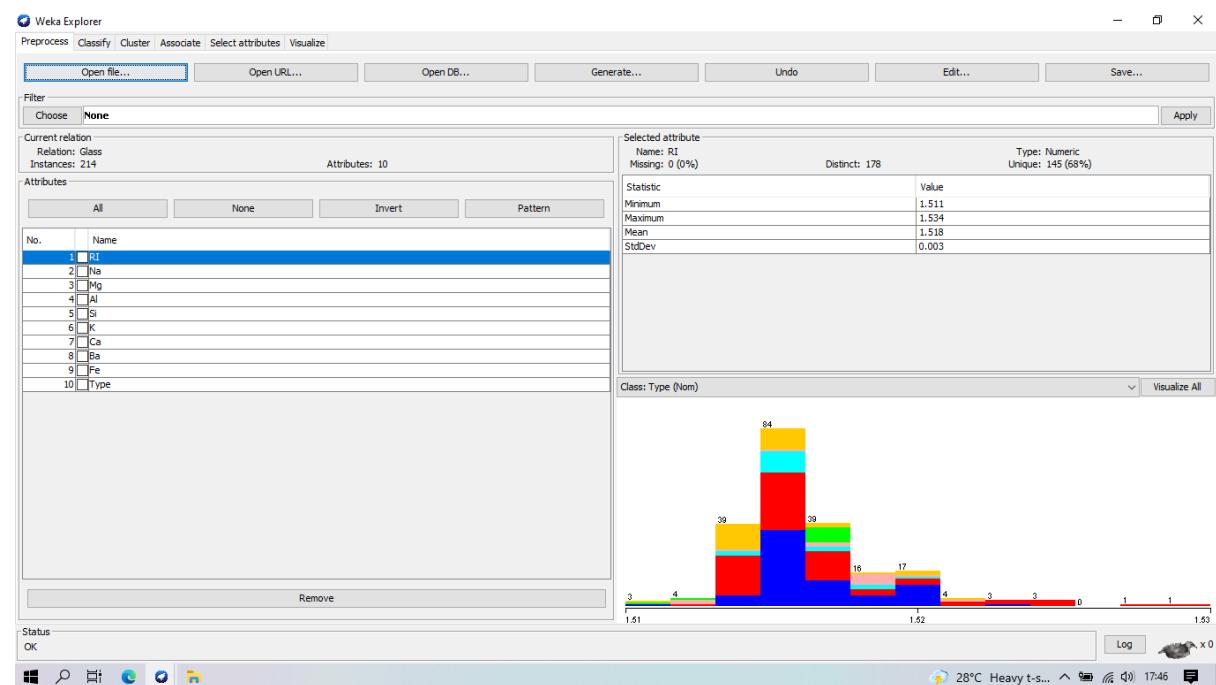
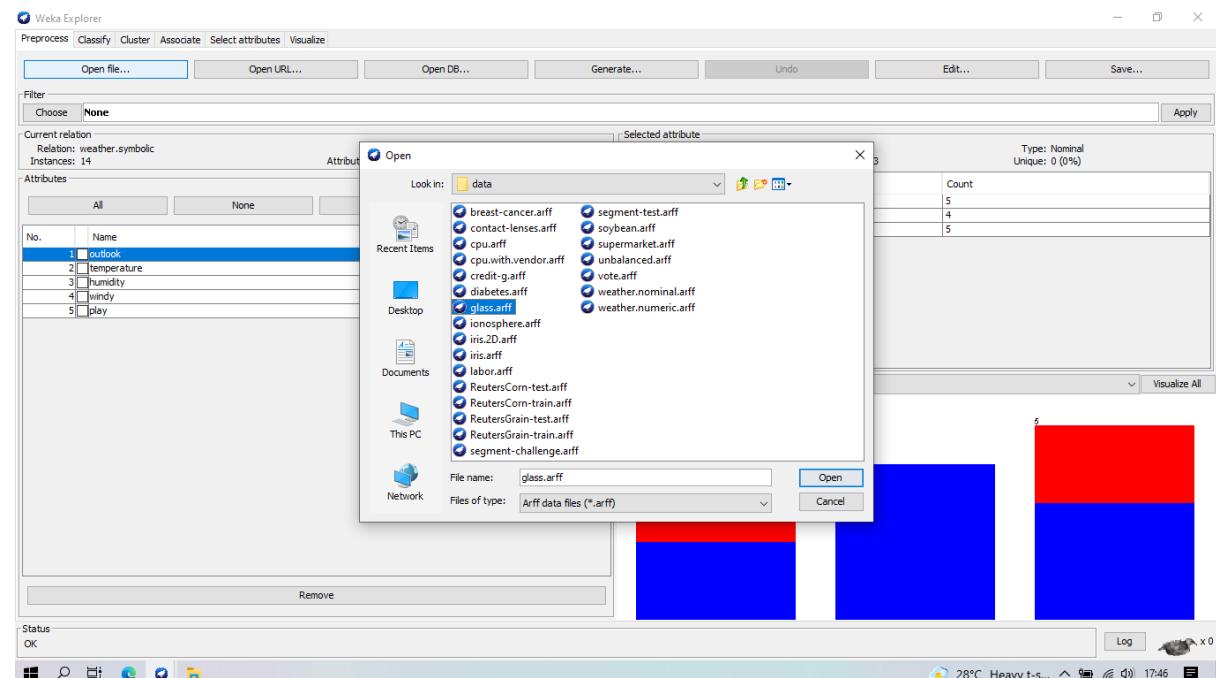
```
weather.nominal.arff - Notepad
File Edit Format View Help
@relation weather.symbolic

@attribute outlook {sunny, overcast, rainy}
@attribute temperature {hot, mild, cool}
@attribute humidity {high, normal}
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

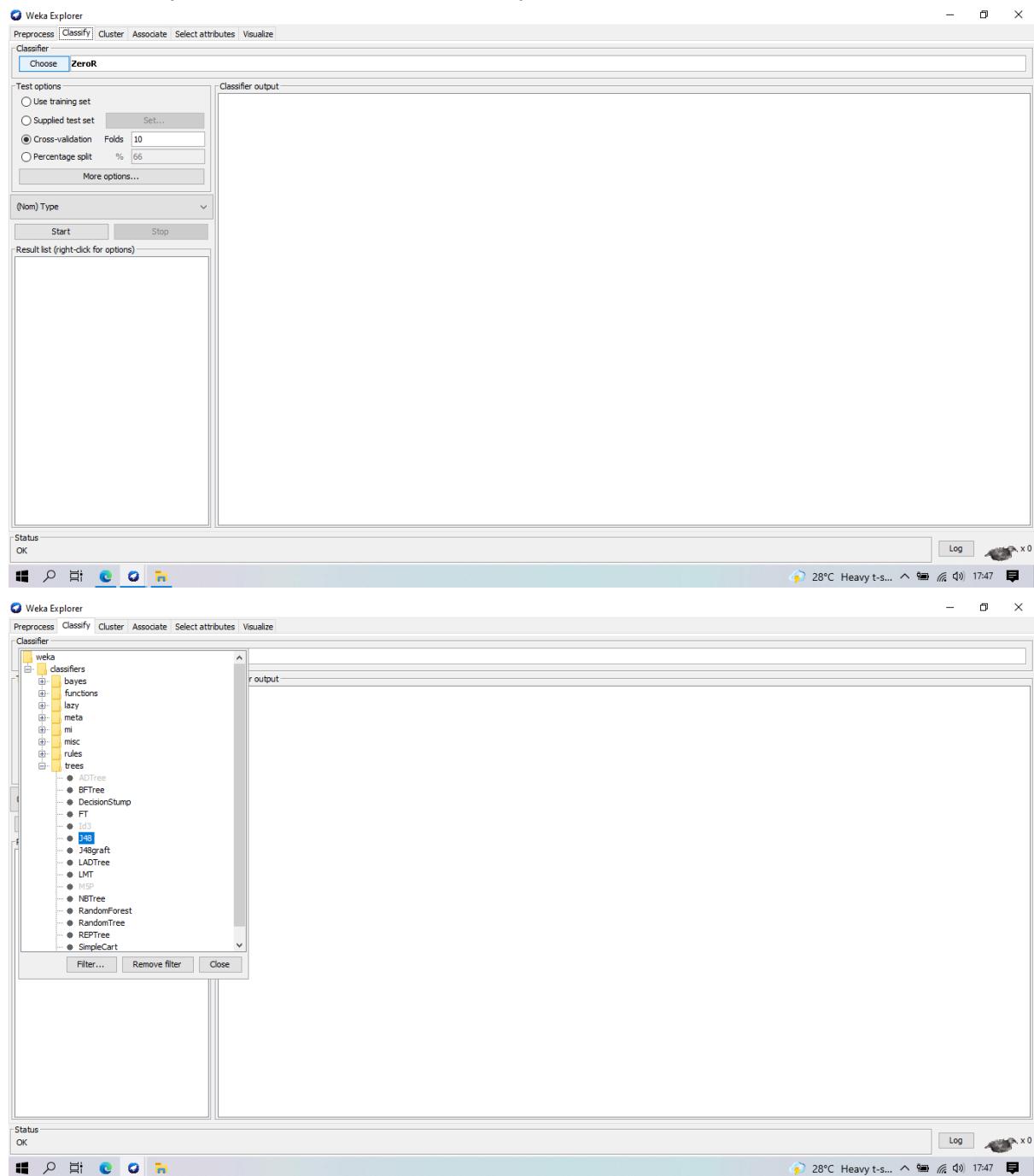
@data
sunny,hot,high,FALSE,no
sunny,hot,high,TRUE,no
overcast,hot,high,FALSE,yes
rainy,mild,high,FALSE,yes
rainy,cool,normal,FALSE,yes
rainy,cool,normal,TRUE,no
overcast,cool,normal,TRUE,yes
sunny,mild,high,FALSE,no
sunny,cool,normal,FALSE,yes
rainy,mild,normal,FALSE,yes
rainy,mild,normal,TRUE,yes
sunny,mild,normal,TRUE,yes
overcast,mild,high,TRUE,yes
overcast,hot,normal,FALSE,yes
rainy,mild,high,TRUE,no
```

1.4 Building a Classifier

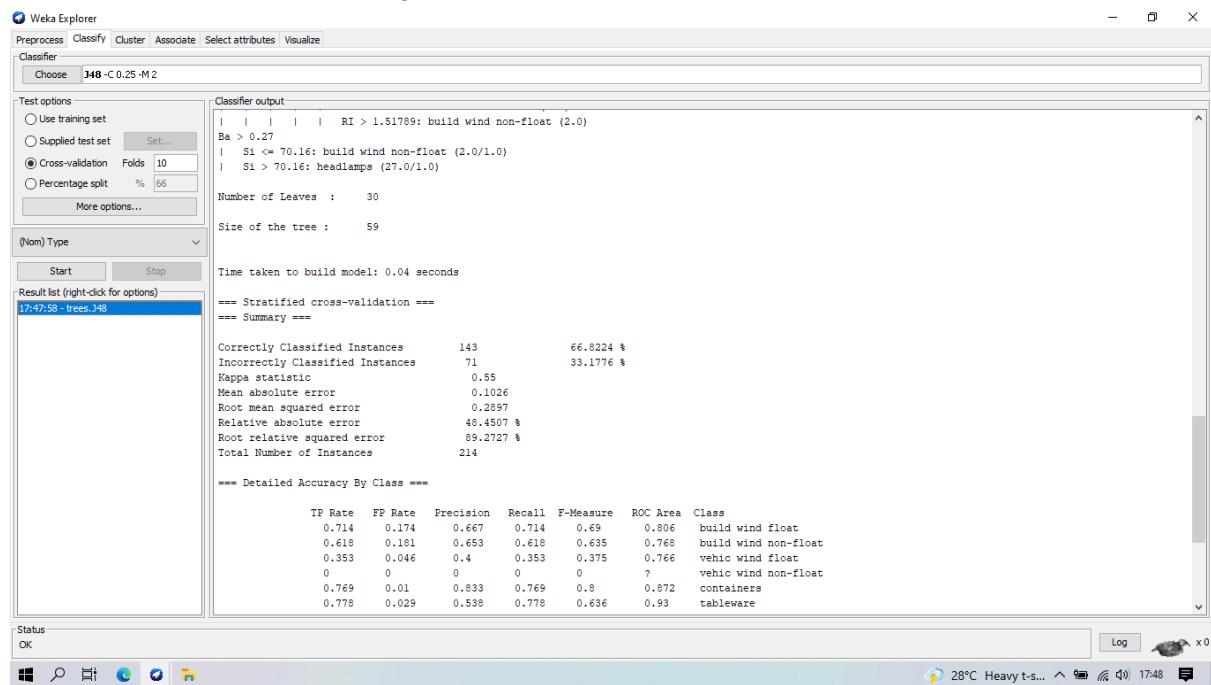
Buka file dataset glass.aff



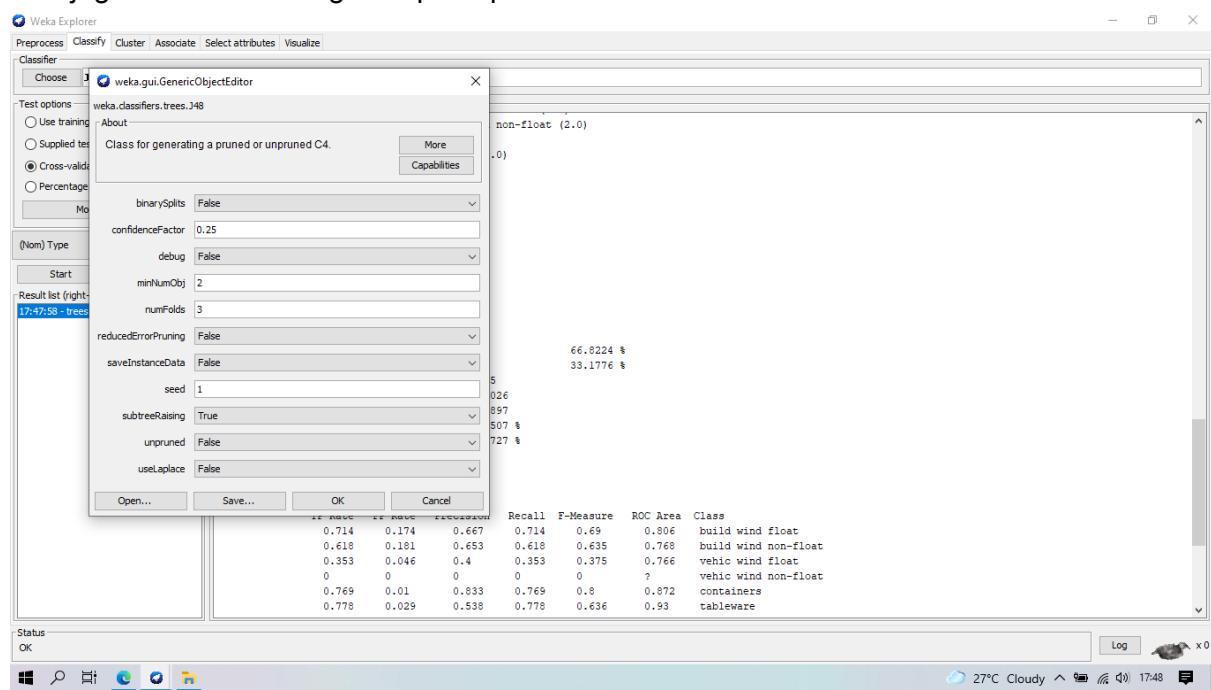
Klik tab Classify, kemudian pilih tipe classifier, yaitu Tree J48.



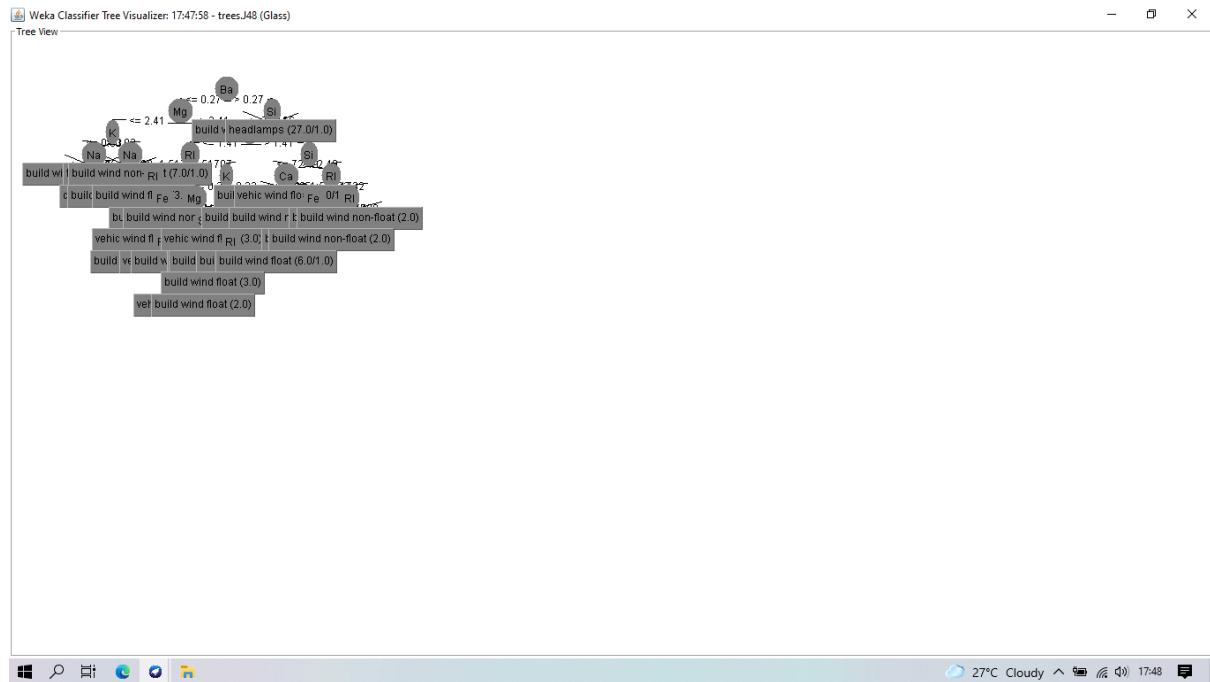
Buka hasil dari klasifikasi, dengan cara klik Start.



Bisa juga dilakukan konfigurasi pada parameter untuk J48.

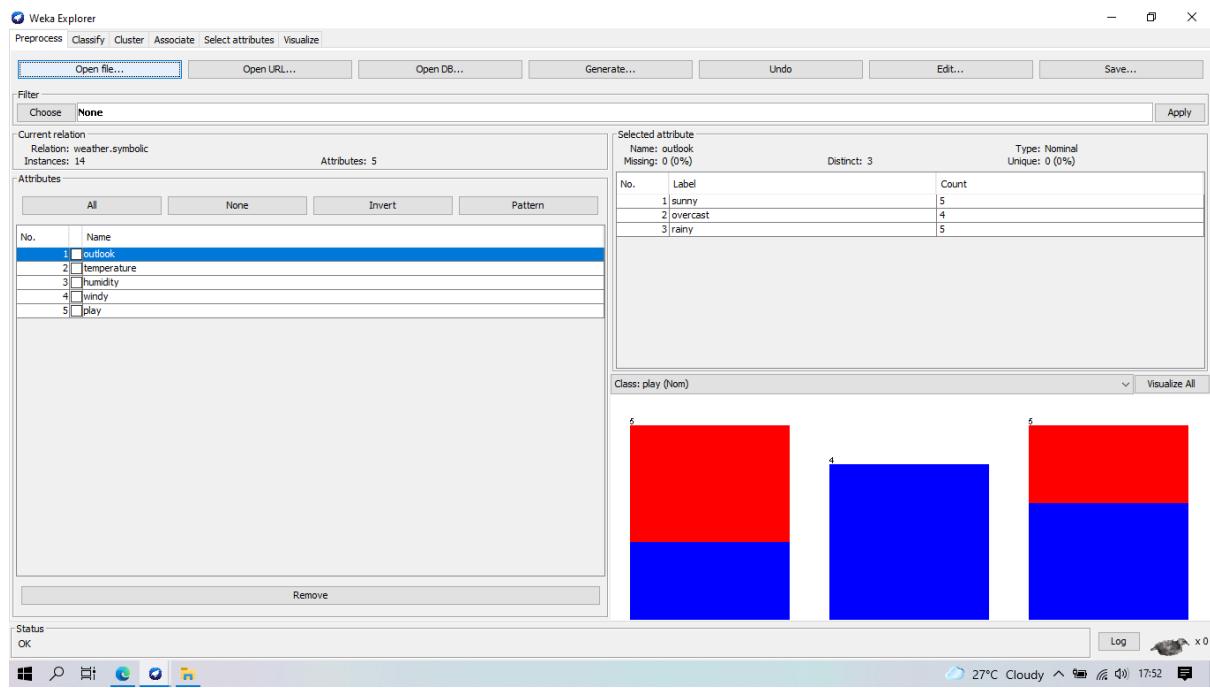


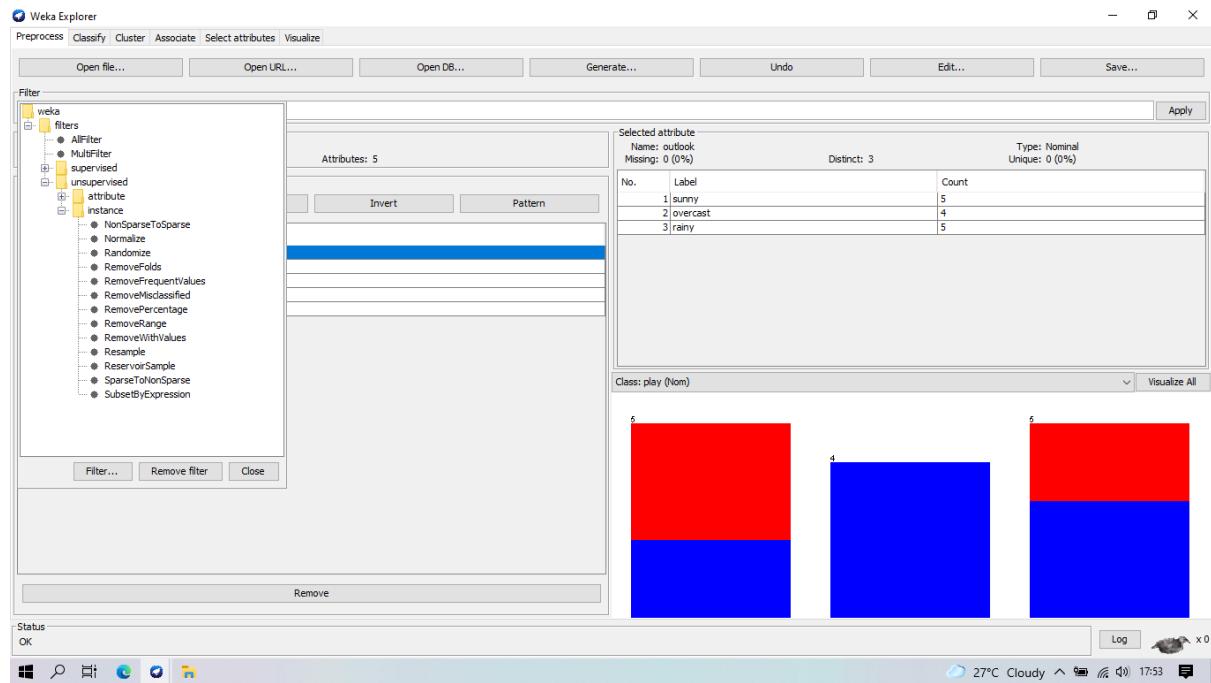
Visualisasi tree j48 untuk dataset glass.arff.



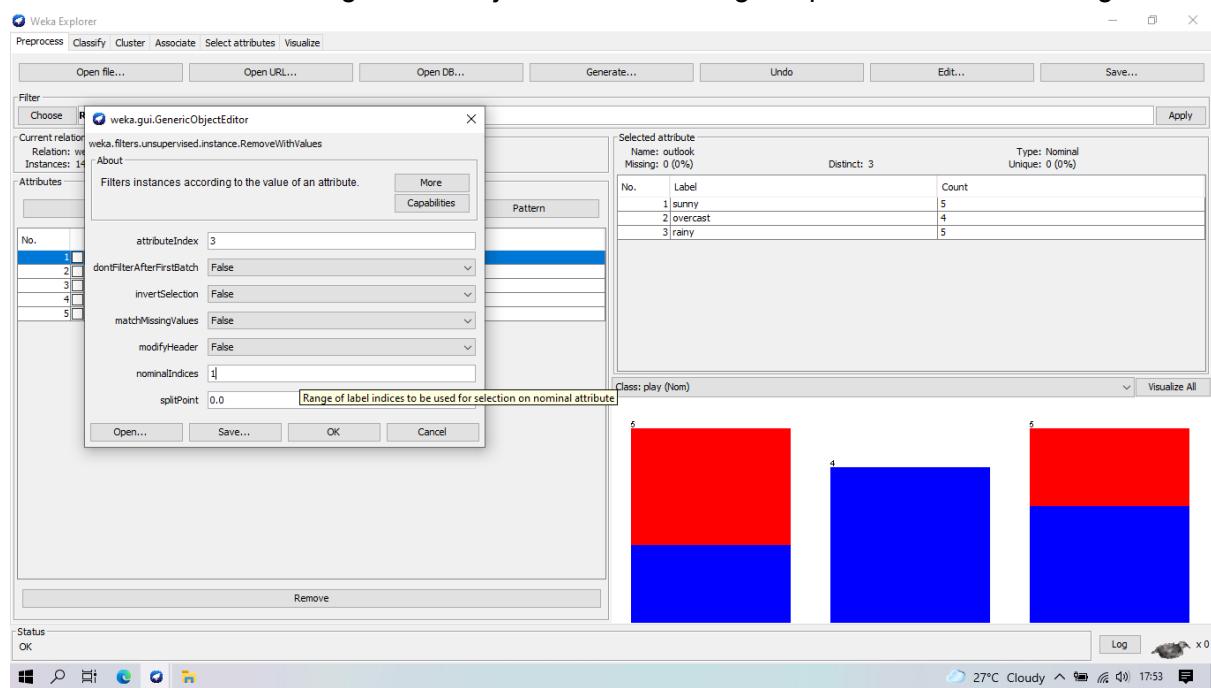
1.5 Using a Filter

Pilih dataset weather-nominal.arff, kemudian cari filter yang ingin digunakan. Untuk contoh, gunakan filter instance di folder unsupervised, yaitu removeWithValues.

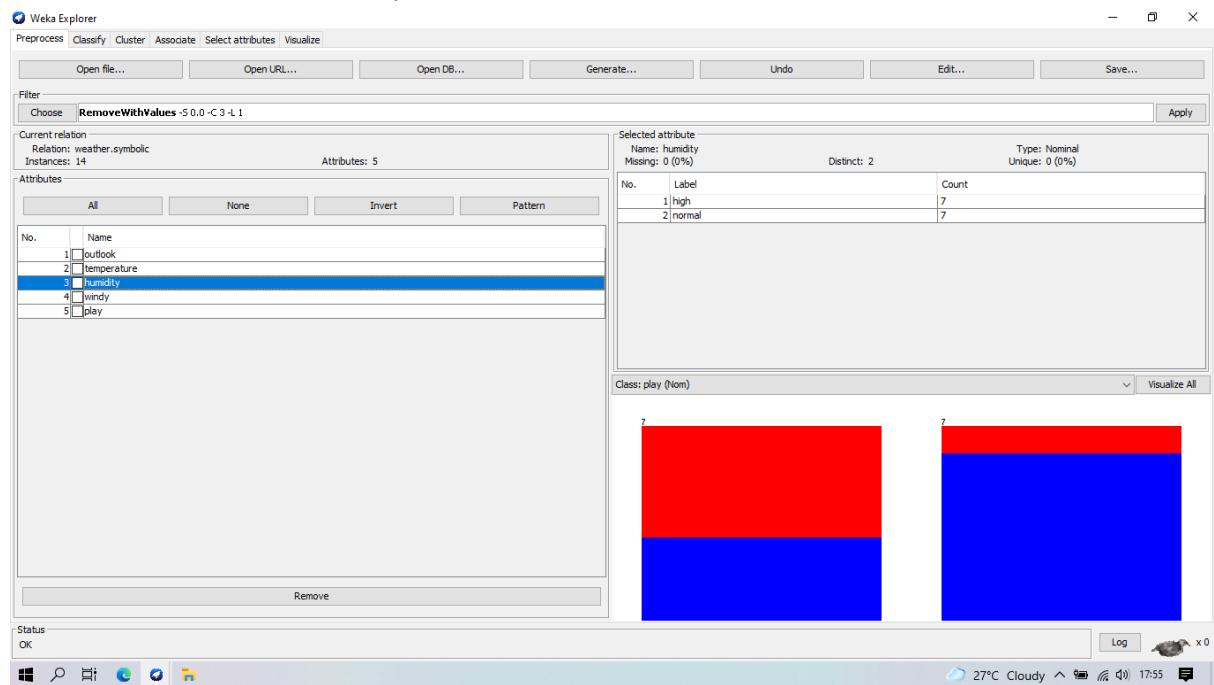




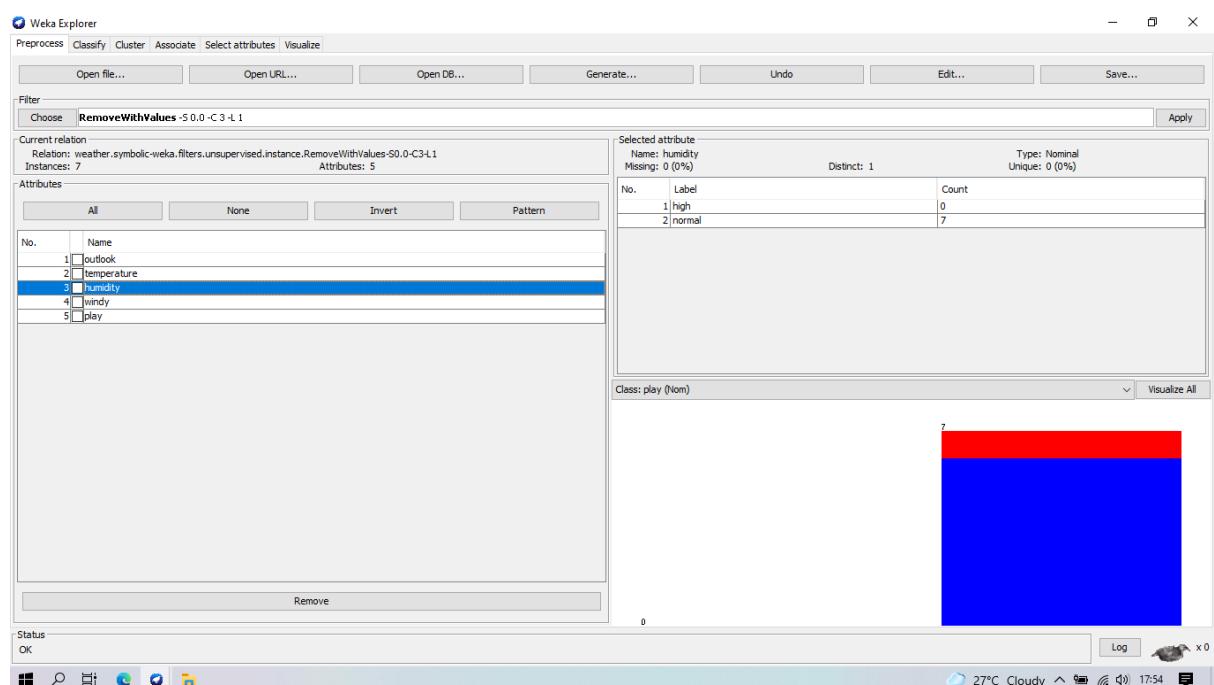
Klik RemoveWithValues agar muncul jendela untuk mengatur parameter sesuai keinginan.



Klik Open, kemudian klik Apply untuk melihat perubahan.



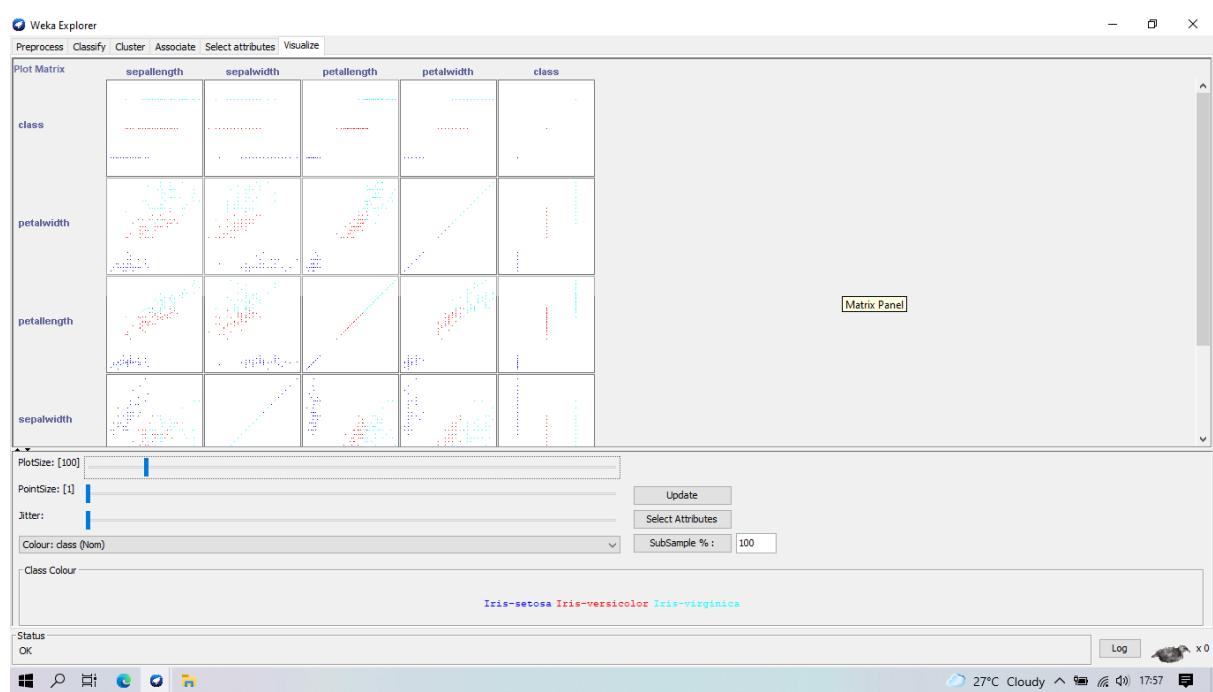
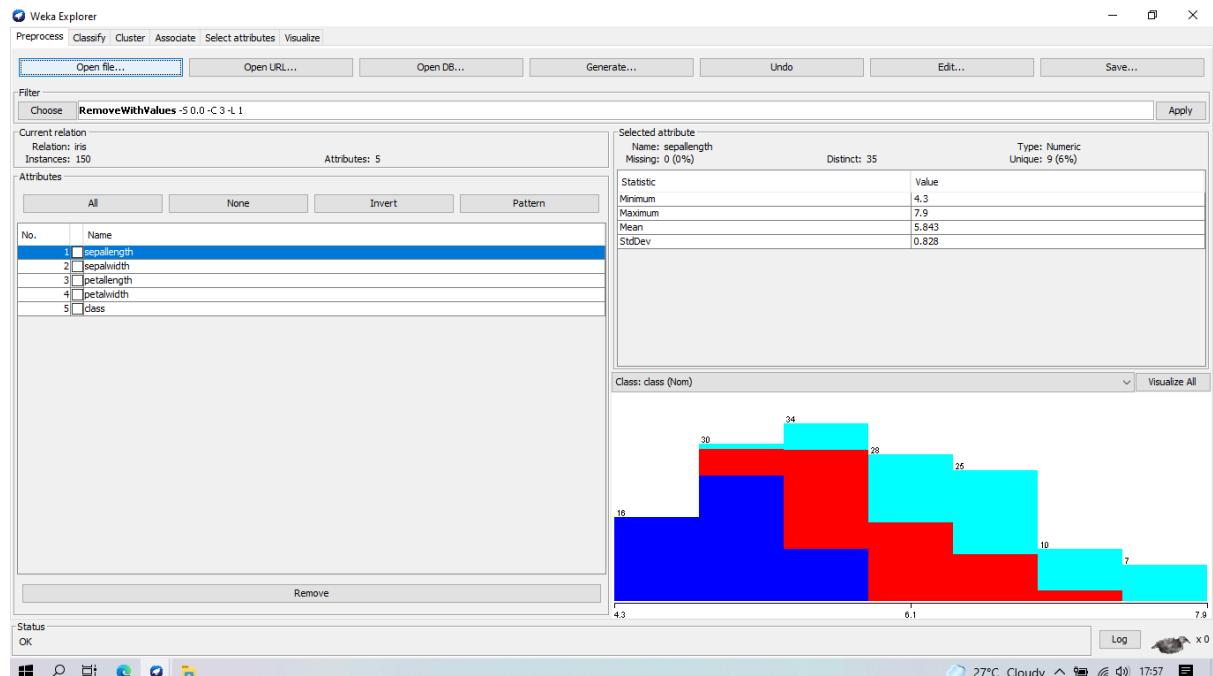
Sebelum diterapkan filter



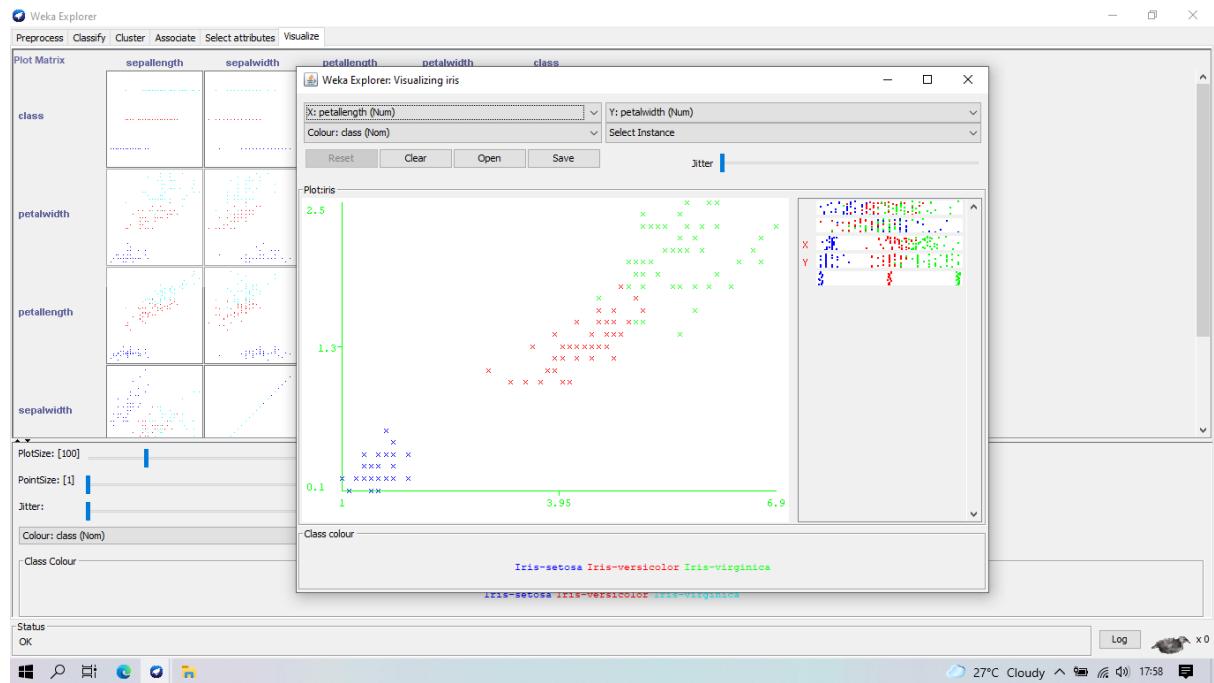
Setelah diterapkan filter

1.6 Visualizing Your Data

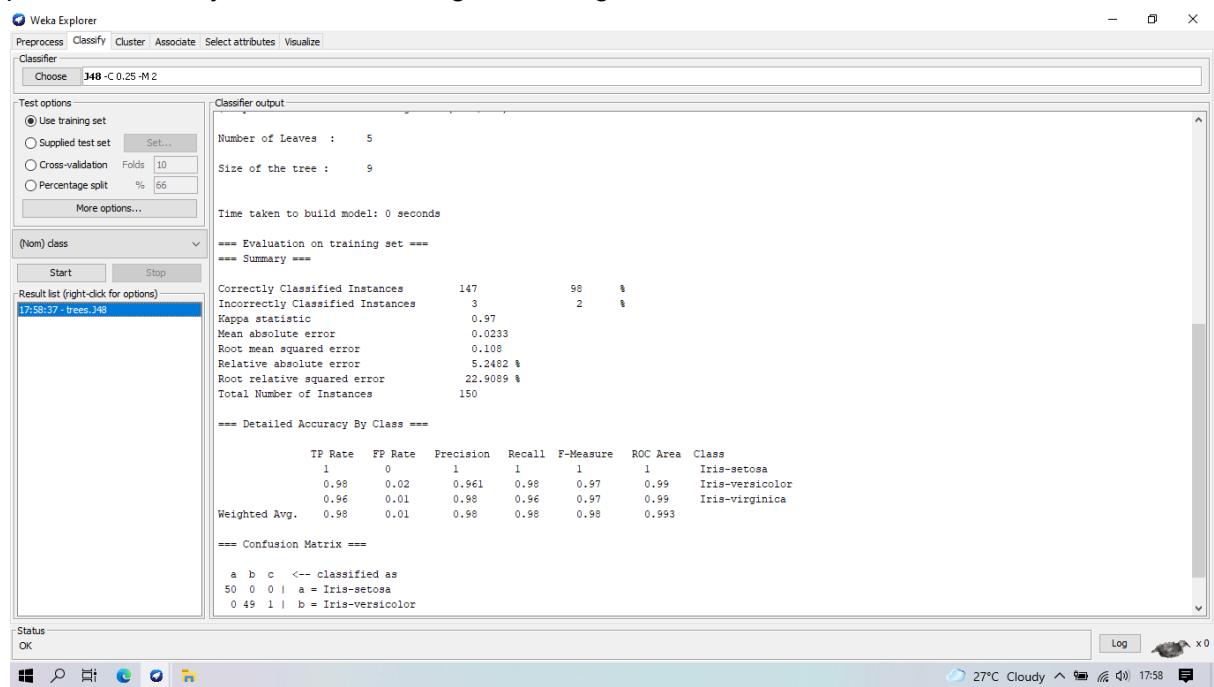
Load dataset iris.arff, Klik tab Visualize untuk melihat visualisasi dari dataset.



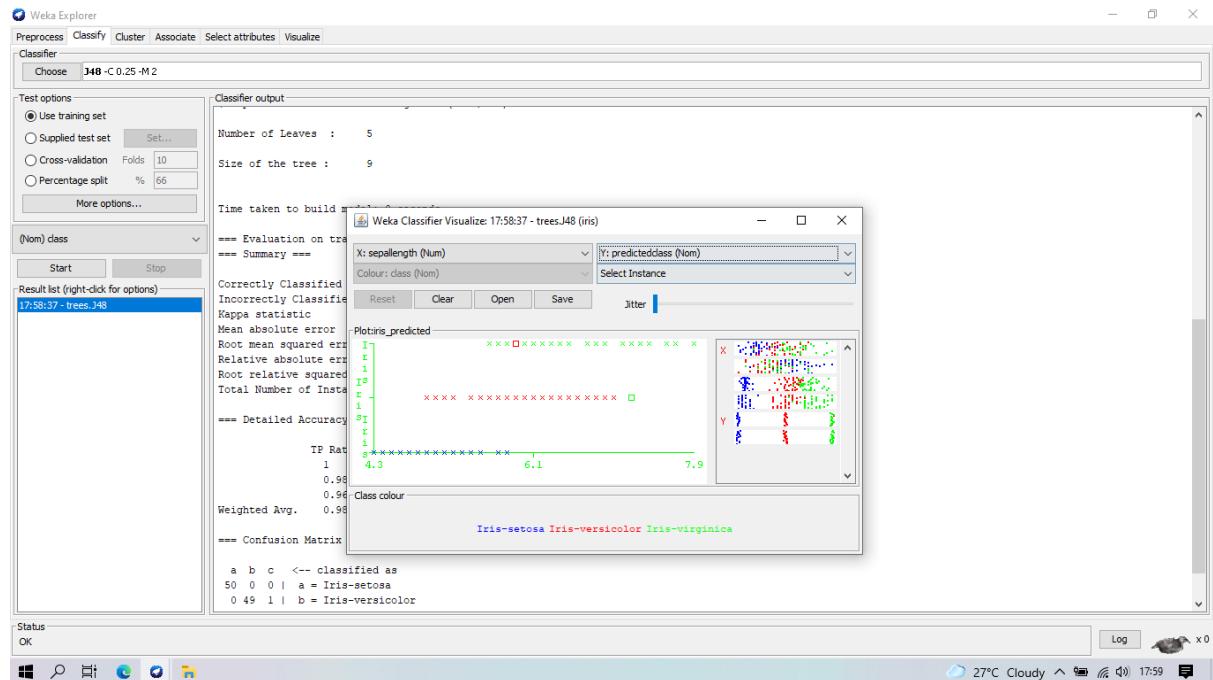
Klik salah satu visualisasinya untuk melihat lebih besar. Juga bisa dimodifikasi atribut yang ingin dijadikan sumbu X dan sumbu Y.



Selain itu, juga bisa dilihat visualisasi untuk hasil klasifikasi. Pertama lakukan klasifikasi pada tab classify. Untuk contoh, digunakan algoritma tree J48.

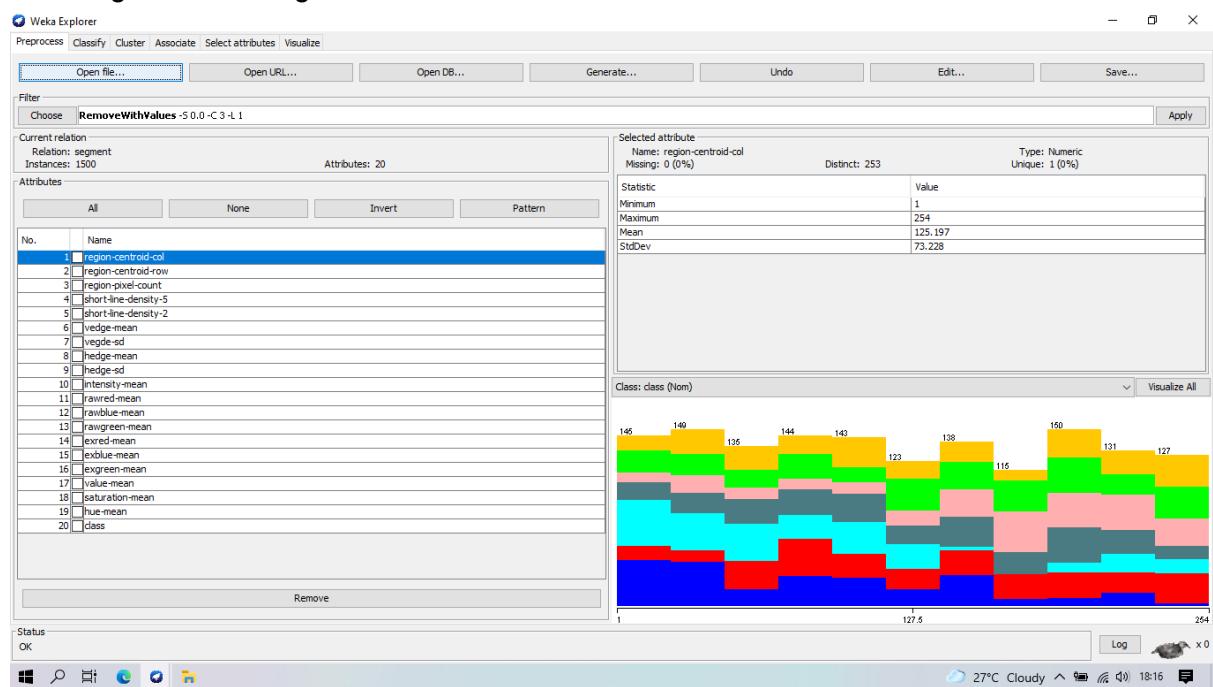


Klik kanan pada result list, kemudian pilih visualize classifier error.

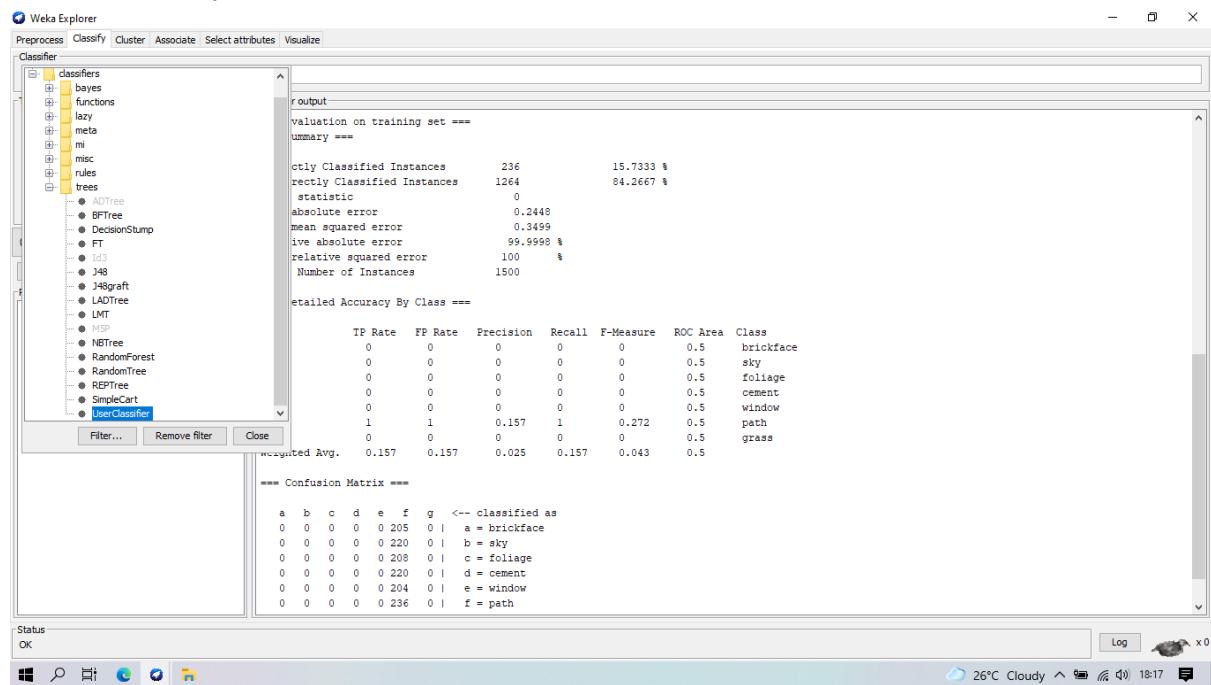


2.1 Be a Classifier!

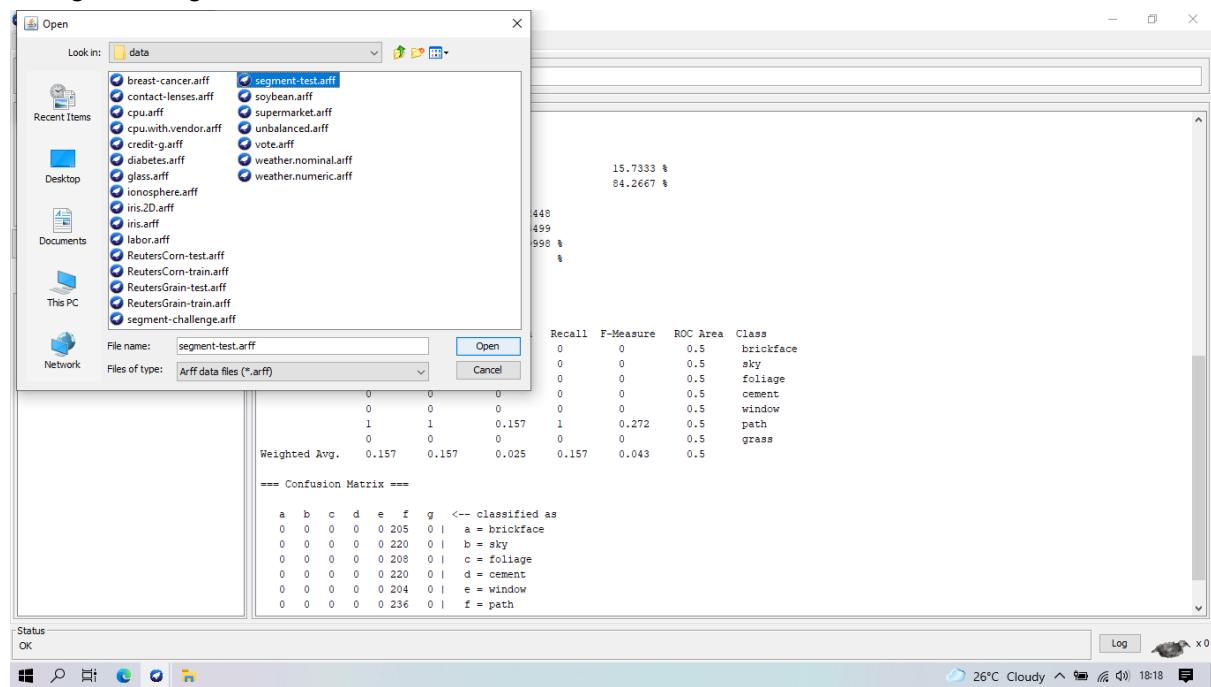
Load segment-challenge dataset.



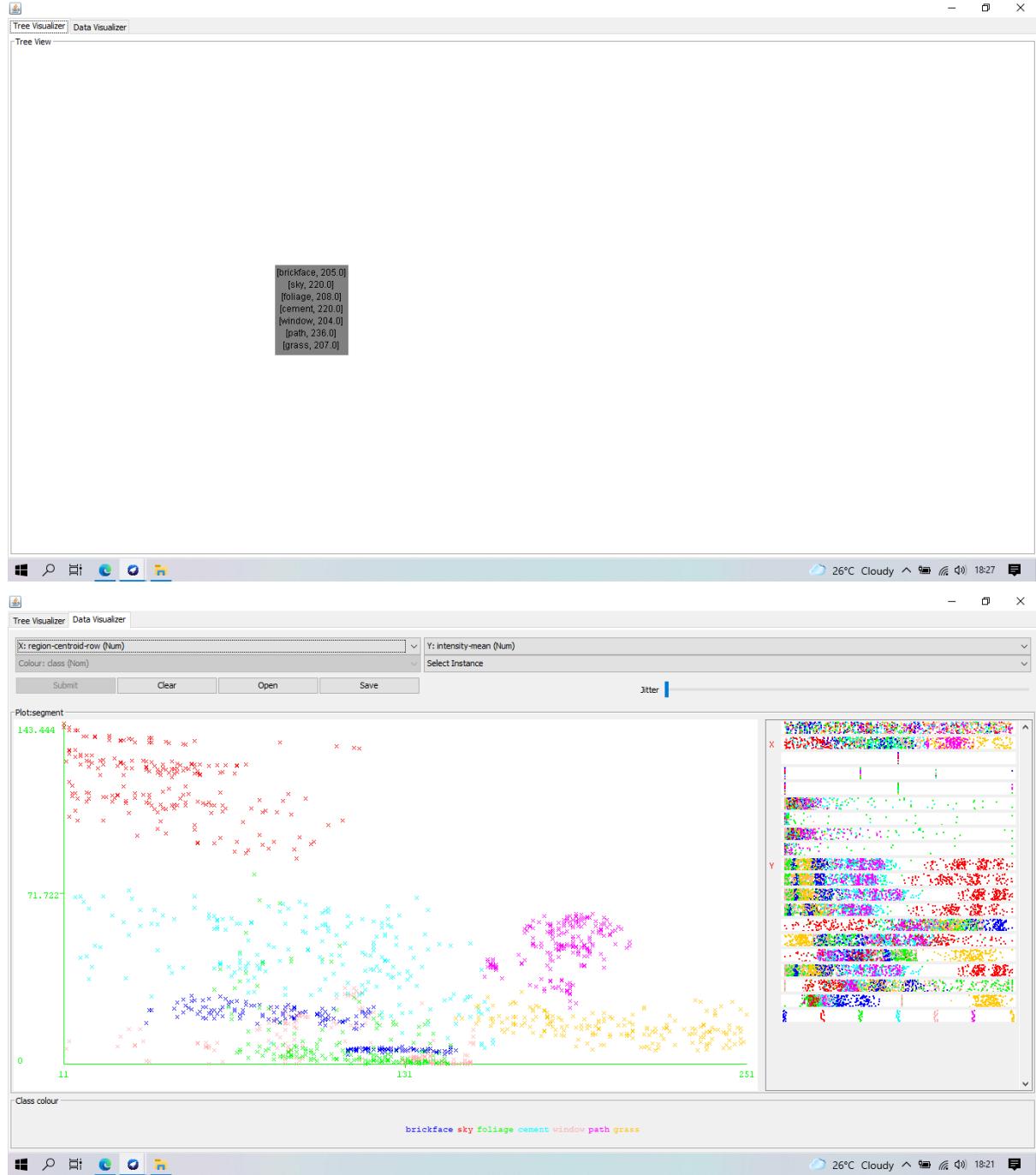
Buka tab classify, pilih userClassifier.



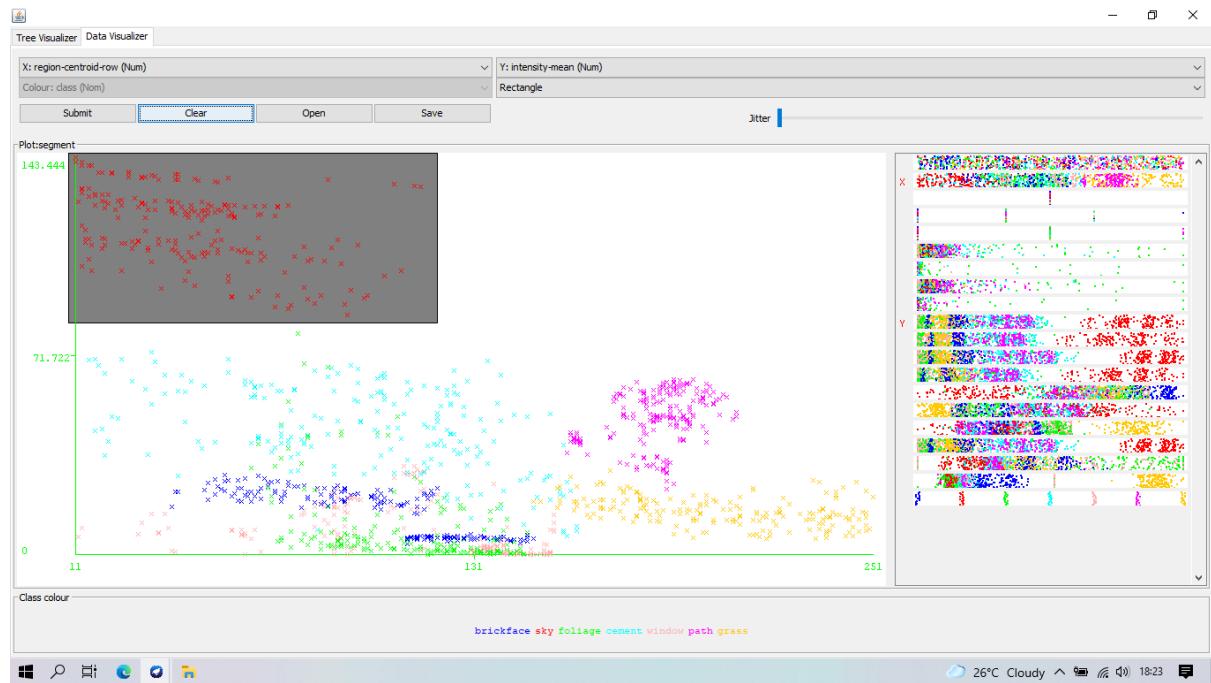
Pada test options, pilih supplied test set, kemudian klik set. Tujuannya untuk memilih dataset yang digunakan sebagai testing untuk classifier yang digunakan. Pilih segment-test.arff sebagai testing dataset.



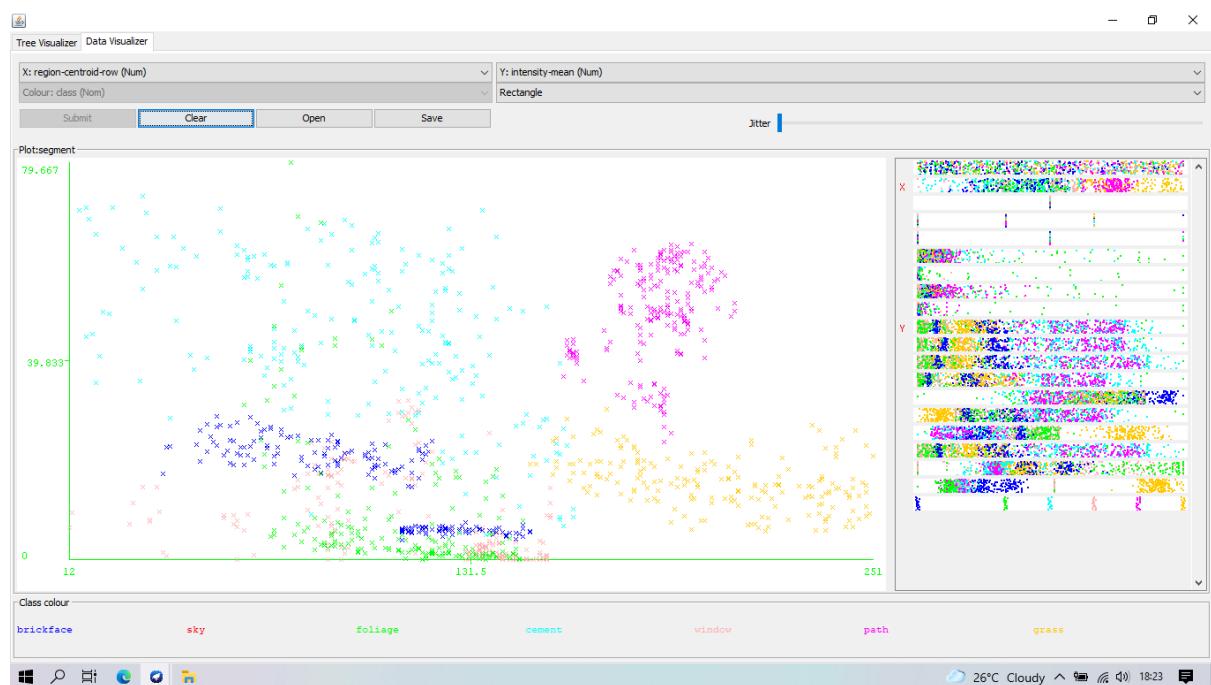
Klik start. Kemudian muncul jendela baru. Klik tab data visualizer, untuk melihat visualisasinya. Pilih sumbu X dan sumbu Y yang kira-kira sesuai sehingga nilai yang berwarna sama terkelompok cukup baik.



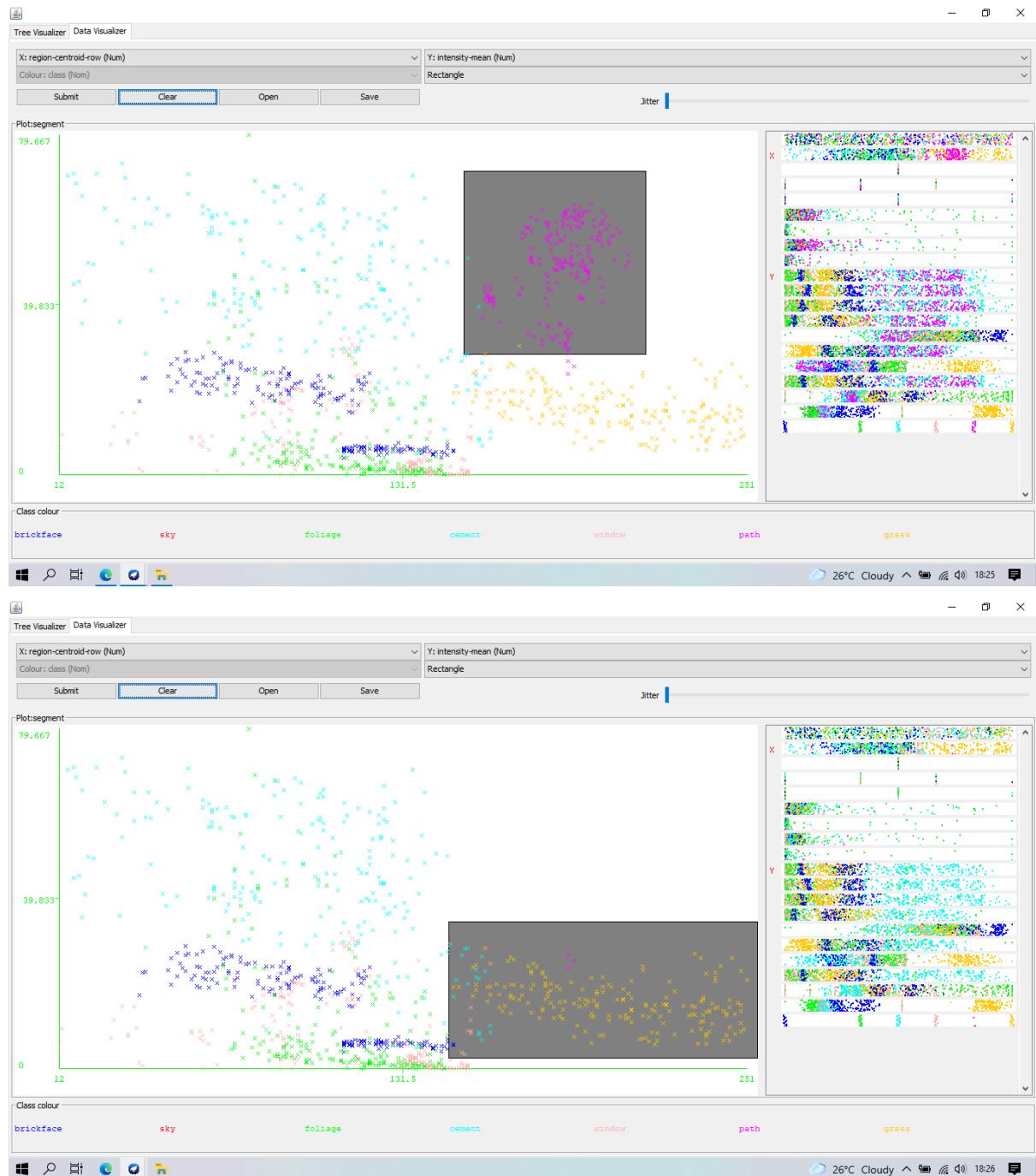
Lakukan data selection pada dropdown select instance, pilih rectangle. Kemudian pilih area dengan koordinat berwarna merah. Kemudian klik submit.



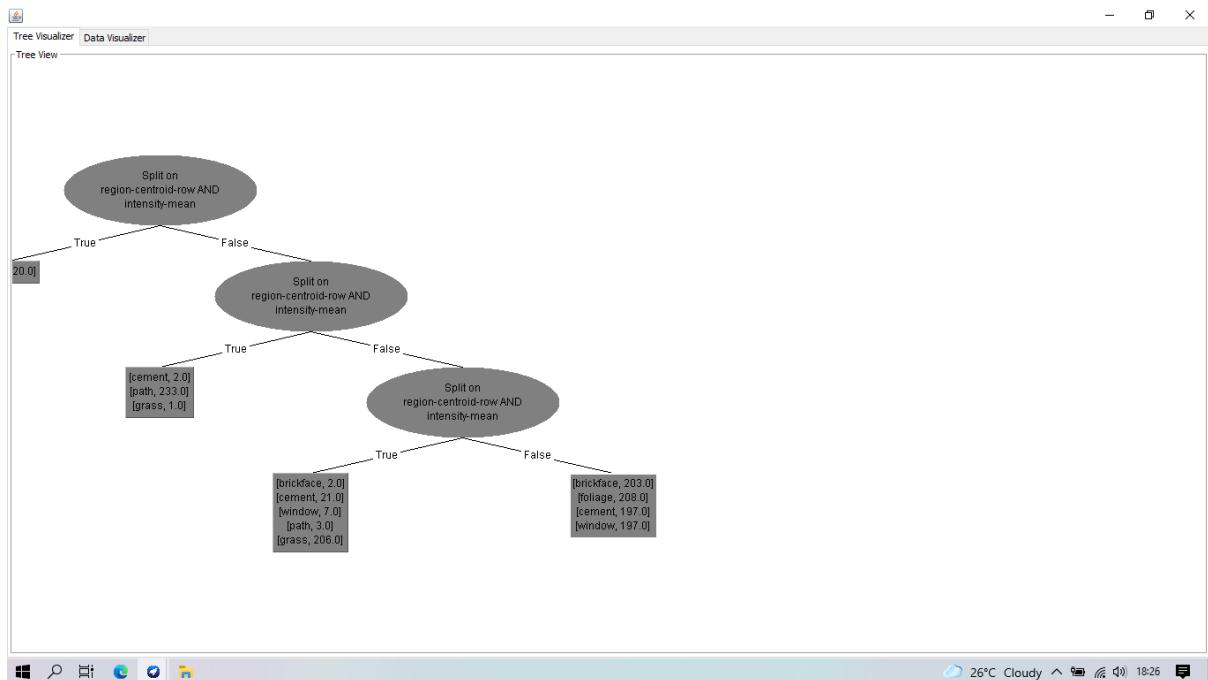
Visualisasinya akan berubah seperti gambar berikut. Hal ini dikarenakan nilai yang telah diseleksi telah diklasifikasikan, sehingga yang terlihat hanya yang belum diklasifikasi oleh User.



Lanjutkan dengan memilih warna lain, seperti ungu, biru, kuning dan seterusnya, sampai semuanya telah terkласifikasi.

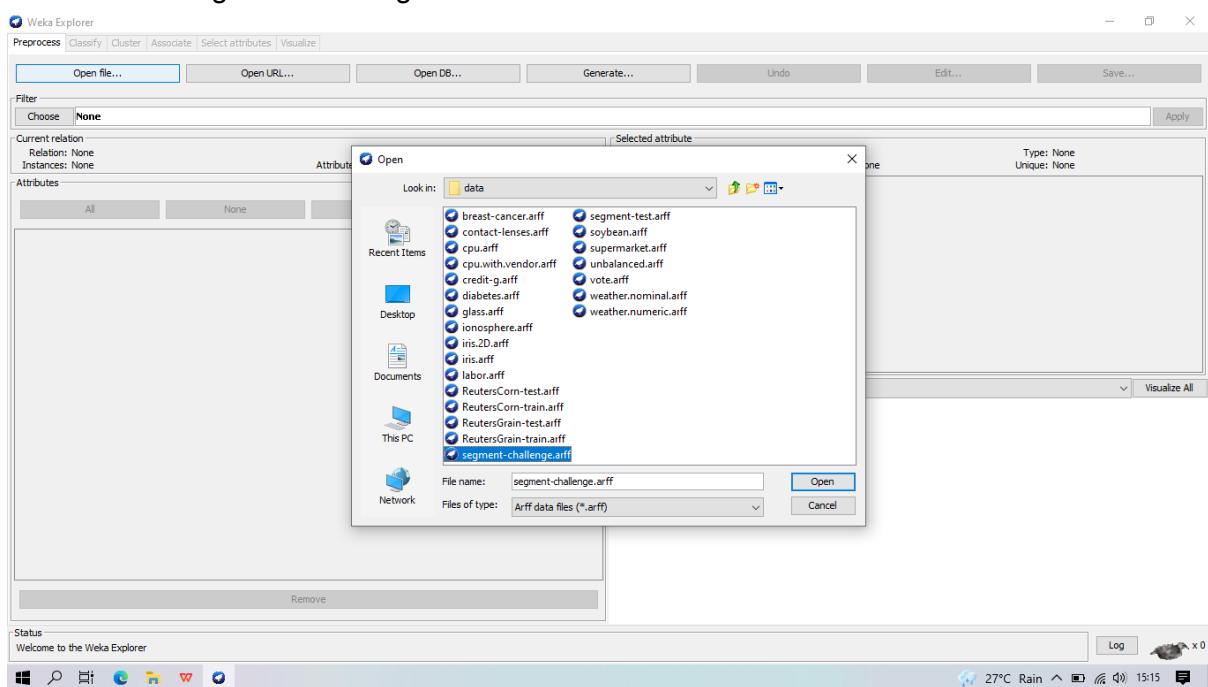


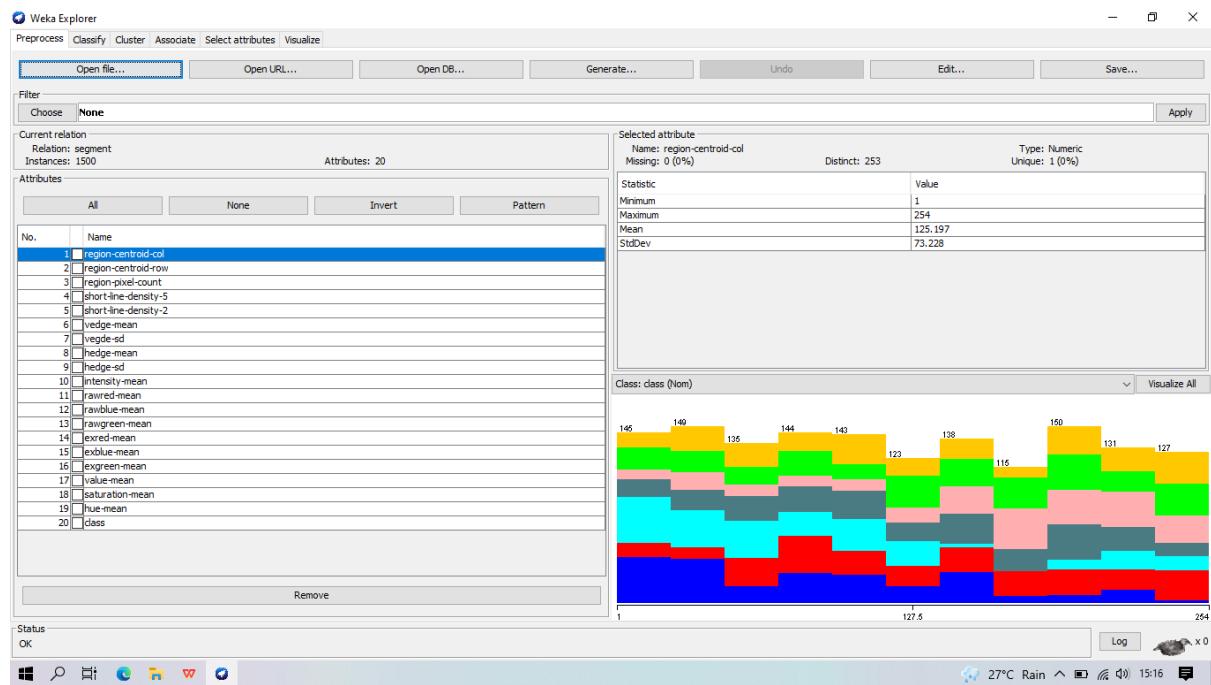
Setelah melakukan klasifikasi, klik tab Tree Visualizer. Terlihat bahwa tree yang ditampilkan berbeda dengan sebelumnya, instance telah diklasifikasikan dengan cukup baik.



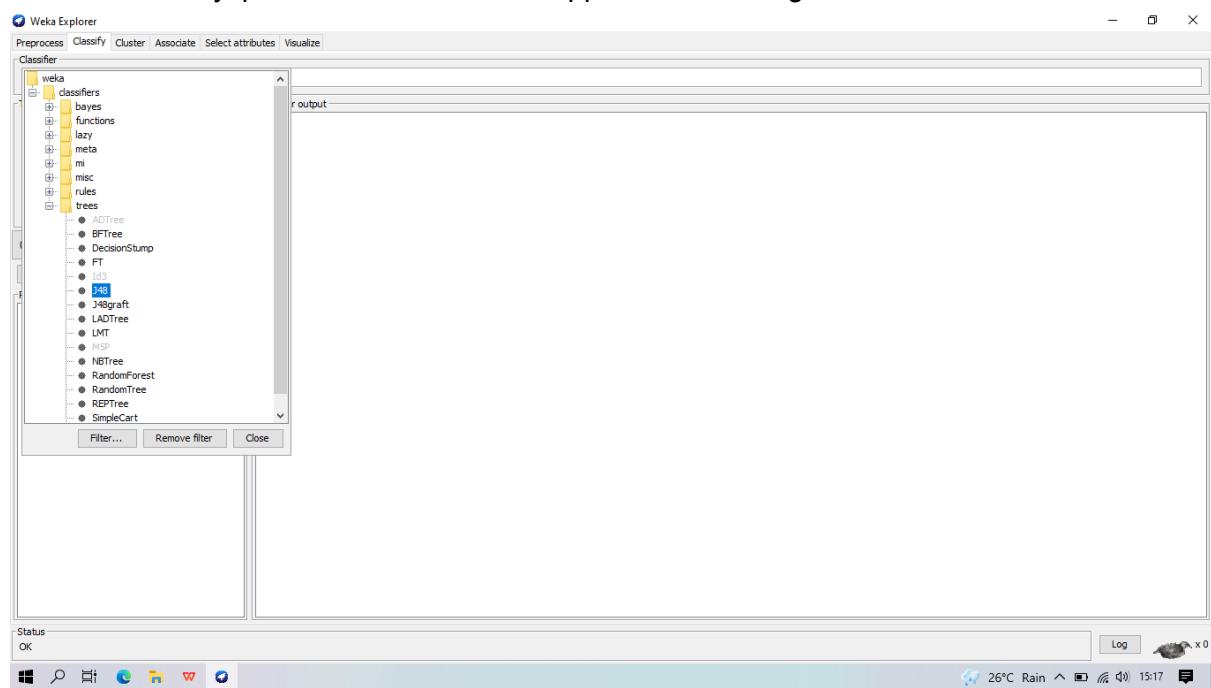
2.2 Training and Testing

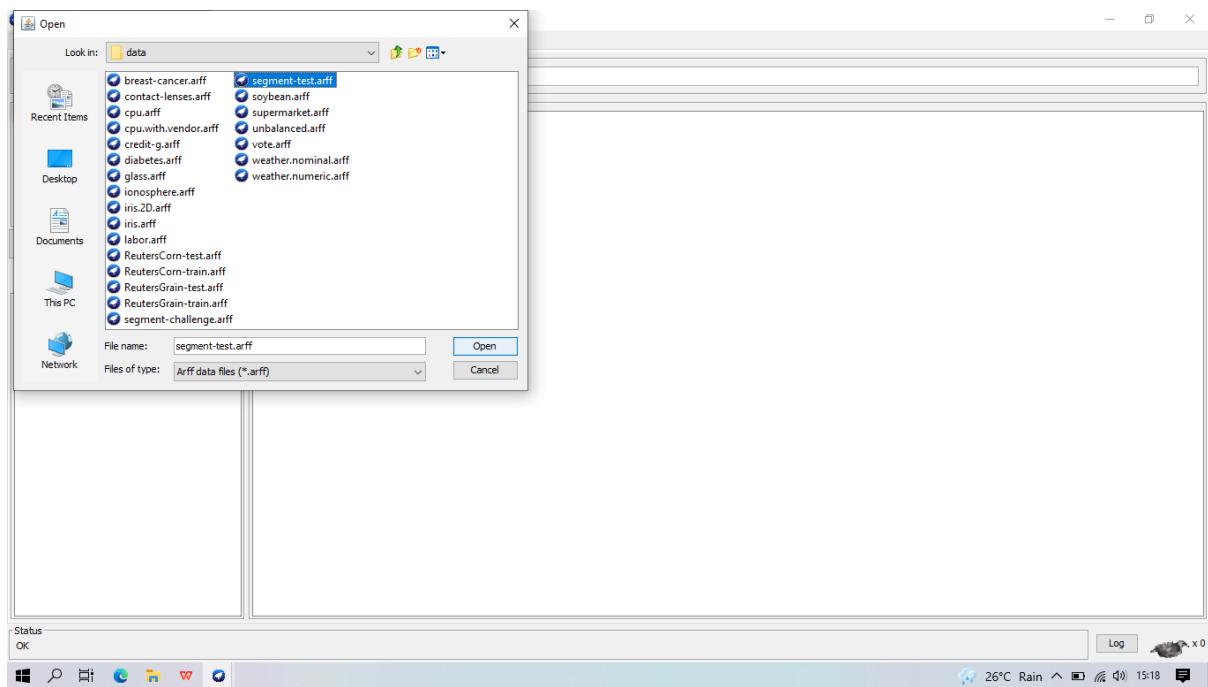
Load dataset segment-challenge.arff



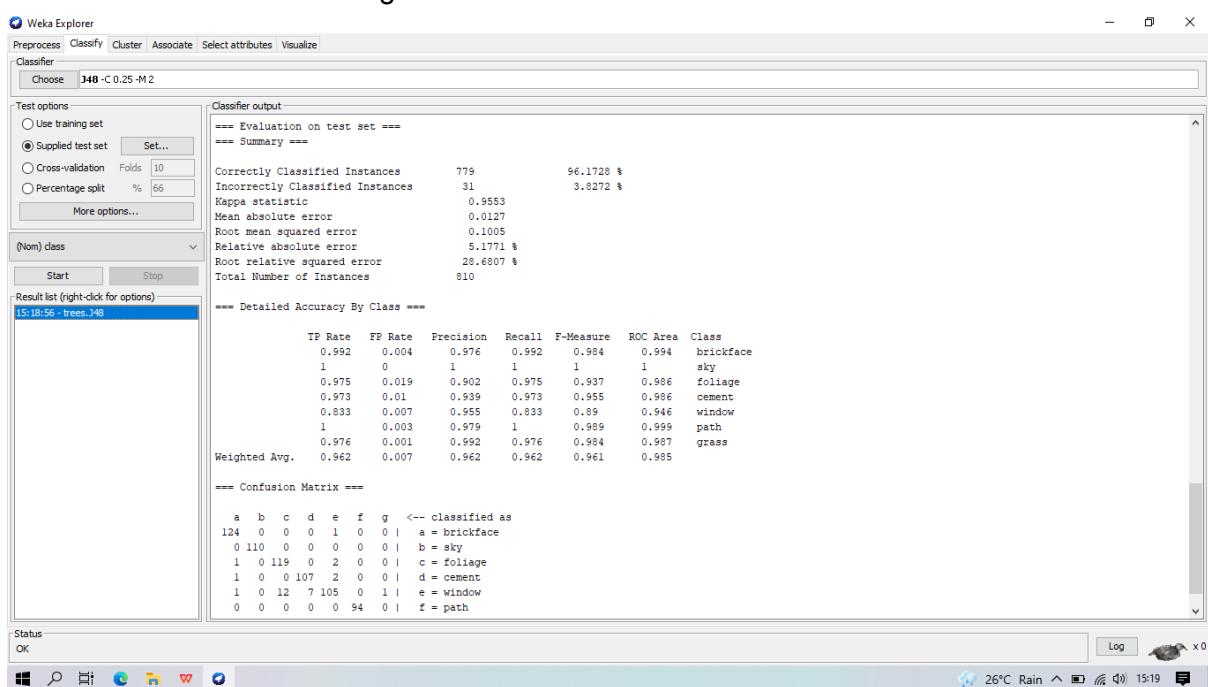


Pada tab classify, pilih classifier J48 dan supplied test set segment-test.arff.

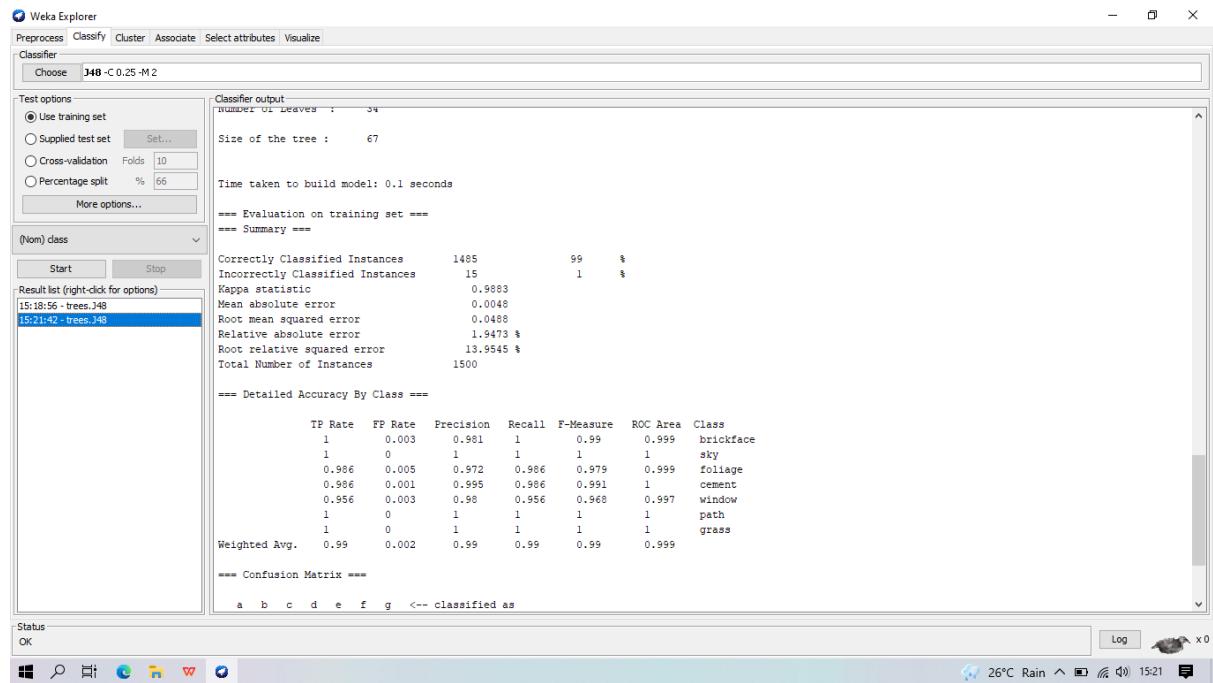




Klik Start. Akurasi hasil testing adalah 96%.



Ubah test option menjadi use training set. Akurasi berupa menjadi 99% (hal ini tidak seharusnya dilakukan dalam melakukan testing).

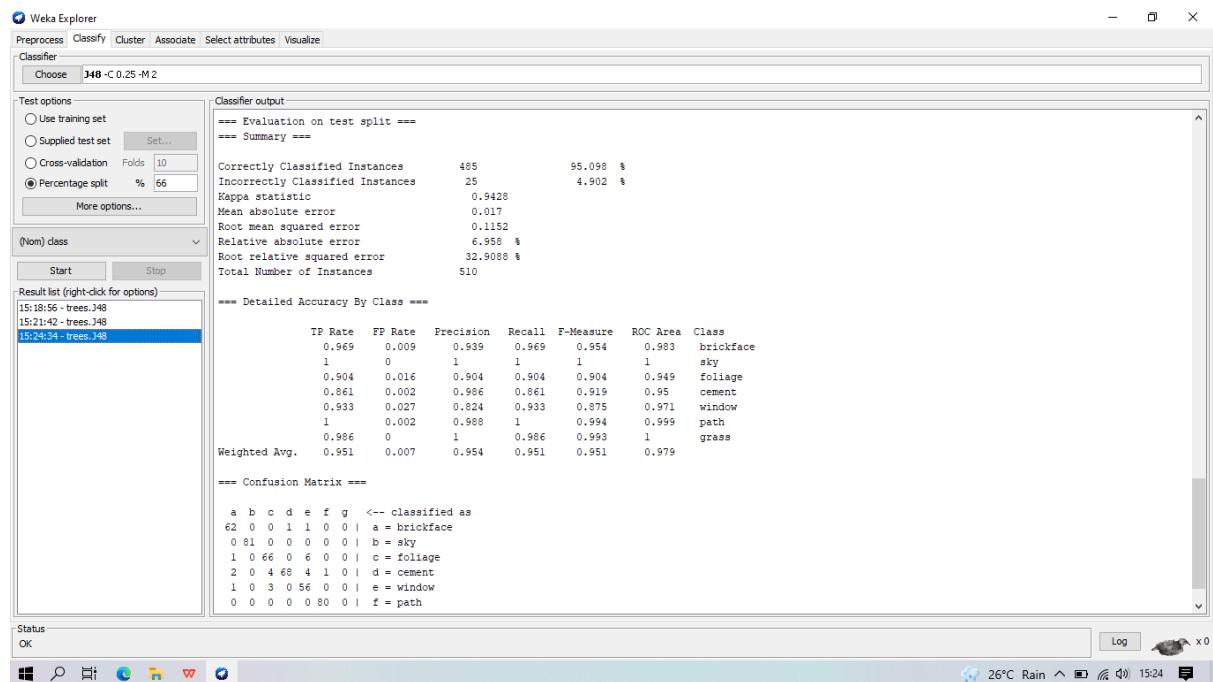


```

Weka Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Classifier Choose J48 -C 0.25 -M 2
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)
15:18:56 - trees.J48
15:21:42 - trees.J48
Classifier output
NUMBER OF LEAVES : 34
Size of the tree : 67
Time taken to build model: 0.1 seconds
*** Evaluation on training set ***
*** Summary ***
Correctly Classified Instances 1485 99 %
Incorrectly Classified Instances 15 1 %
Kappa statistic 0.9883
Mean absolute error 0.0048
Root mean squared error 0.0488
Relative absolute error 1.9473 %
Root relative squared error 13.9545 %
Total Number of Instances 1500
*** Detailed Accuracy By Class ***
TP Rate FP Rate Precision Recall F-Measure ROC Area Class
1 0.003 0.981 1 0.99 0.999 brickface
1 0 1 1 1 1 sky
0.986 0.005 0.972 0.986 0.979 0.999 foliage
0.986 0.001 0.995 0.986 0.991 1 cement
0.956 0.003 0.98 0.956 0.968 0.997 window
1 0 1 1 1 1 path
1 0 1 1 1 1 grass
Weighted Avg. 0.99 0.002 0.99 0.99 0.999 0.999
*** Confusion Matrix ***
a b c d e f g <-- classified as

```

Jika ingin menggunakan satu dataset saja, pilih percentage split pada test option, kemudian diatur persentase training set. Sebagai contoh, persentase diatur menjadi 66%. Akurasi yang didapatkan 95%.



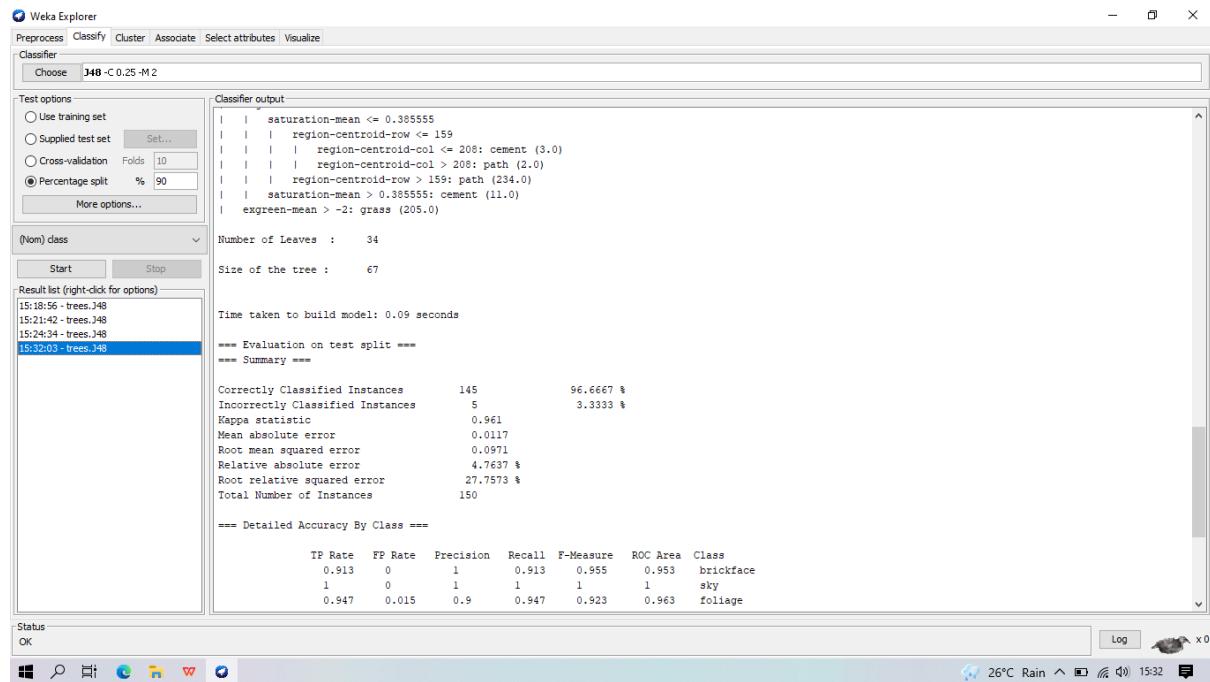
```

Weka Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Classifier Choose J48 -C 0.25 -M 2
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)
15:18:56 - trees.J48
15:21:42 - trees.J48
15:24:34 - trees.J48
Classifier output
*** Evaluation on test split ***
*** Summary ***
Correctly Classified Instances 485 95.098 %
Incorrectly Classified Instances 25 4.902 %
Kappa statistic 0.9428
Mean absolute error 0.017
Root mean squared error 0.1152
Relative absolute error 6.958 %
Root relative squared error 32.9088 %
Total Number of Instances 510
*** Detailed Accuracy By Class ***
TP Rate FP Rate Precision Recall F-Measure ROC Area Class
0.969 0.009 0.939 0.969 0.954 0.983 brickface
1 0 1 1 1 1 sky
0.904 0.016 0.904 0.904 0.904 0.949 foliage
0.861 0.002 0.986 0.861 0.919 0.95 cement
0.933 0.027 0.824 0.933 0.875 0.971 window
1 0.002 0.988 1 0.994 0.999 path
0.986 0 1 0.986 0.993 1 grass
Weighted Avg. 0.951 0.007 0.954 0.951 0.951 0.979
*** Confusion Matrix ***
a b c d e f g <-- classified as
62 0 0 1 1 0 0 | a = brickface
0 91 0 0 0 0 0 1 | b = sky
1 0 66 0 6 0 0 1 | c = foliage
2 0 4 68 4 1 0 1 | d = cement
1 0 3 0 56 0 0 1 | e = window
0 0 0 0 80 0 1 | f = path

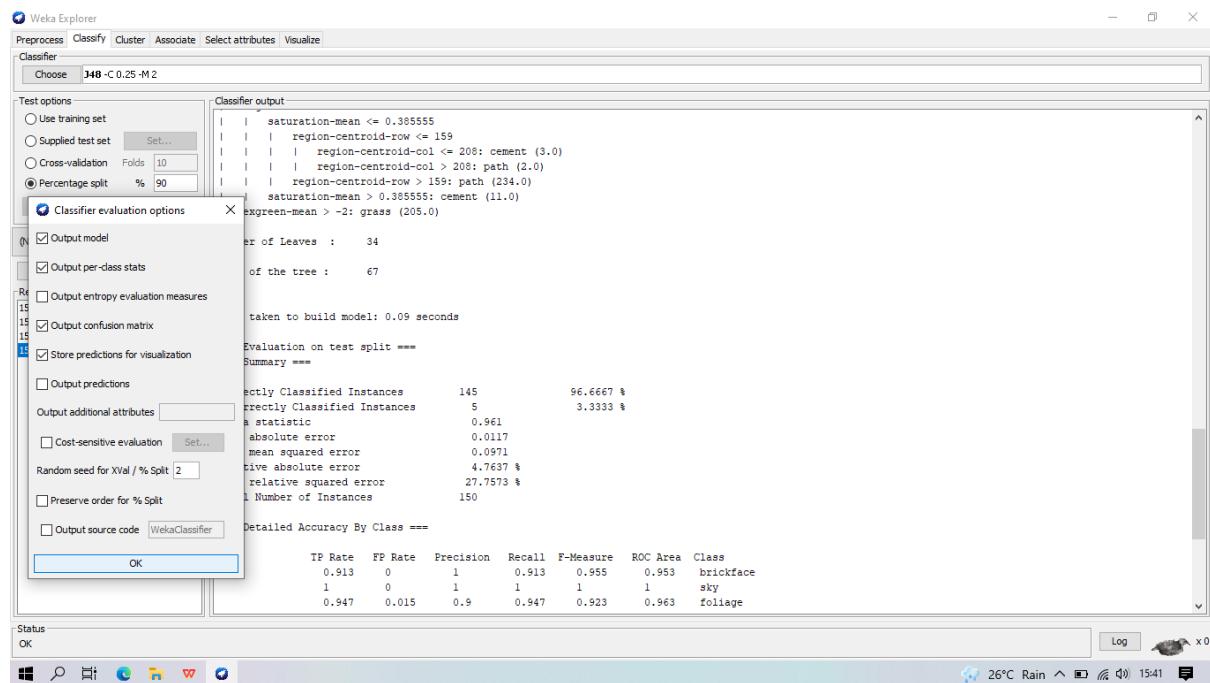
```

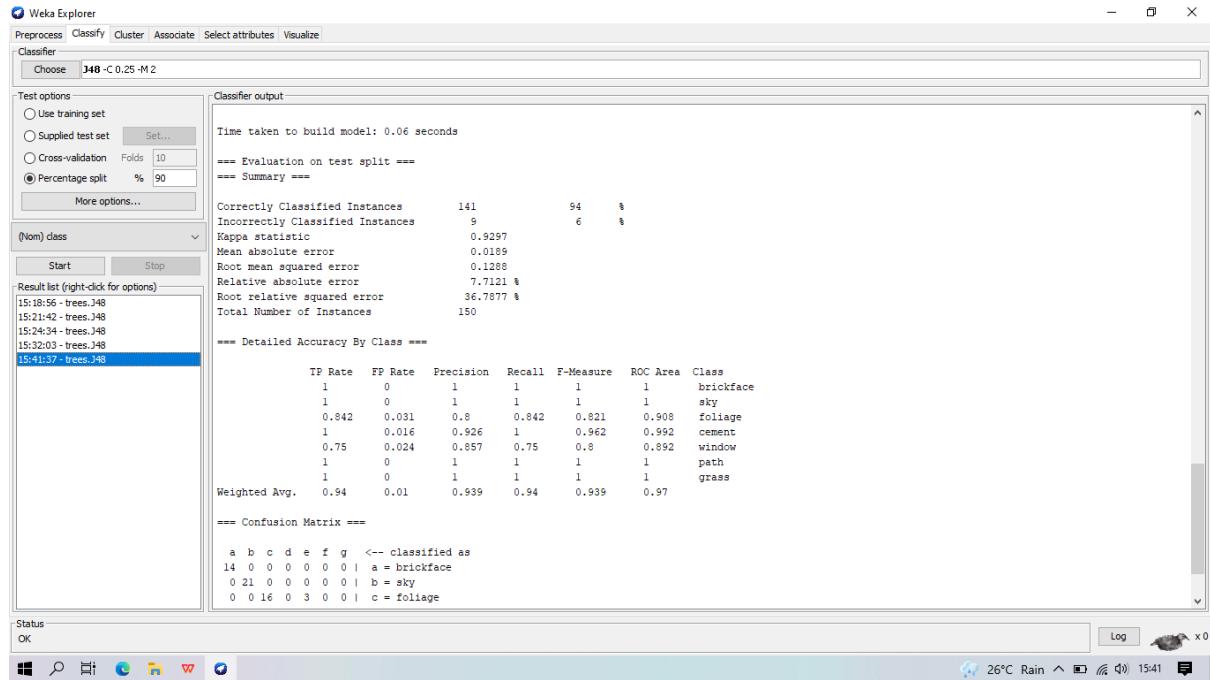
2.3 Repeated Training and Testing

Pada tahap ini tetap menggunakan dataset yang sama dengan tahap sebelumnya yaitu segment-challenge.arff dan classifier yang sama, J48. Karena dataset terdiri 1500 instances, percentage split diatur menjadi 90% (90% untuk training data, 10% untuk testing data). Akurasi yang didapatkan adalah 96,67%



Untuk lebih lanjut, klik more options. Atur random seed for cross-validation menjadi angka yang diinginkan, sebagai contoh, random seed diatur menjadi 2 (Defaultnya bernilai 1). Kemudian klik Start, Akurasi yang didapatkan adalah 94%.





Kemudian diubah lagi nilai random seed for cross-validation sesuai keinginan dan lihat perbedaan akurasi yang didapatkan. Setelah dicoba diubah nilai random seed dari 1-10. Didapatkan hasil sebagai berikut.

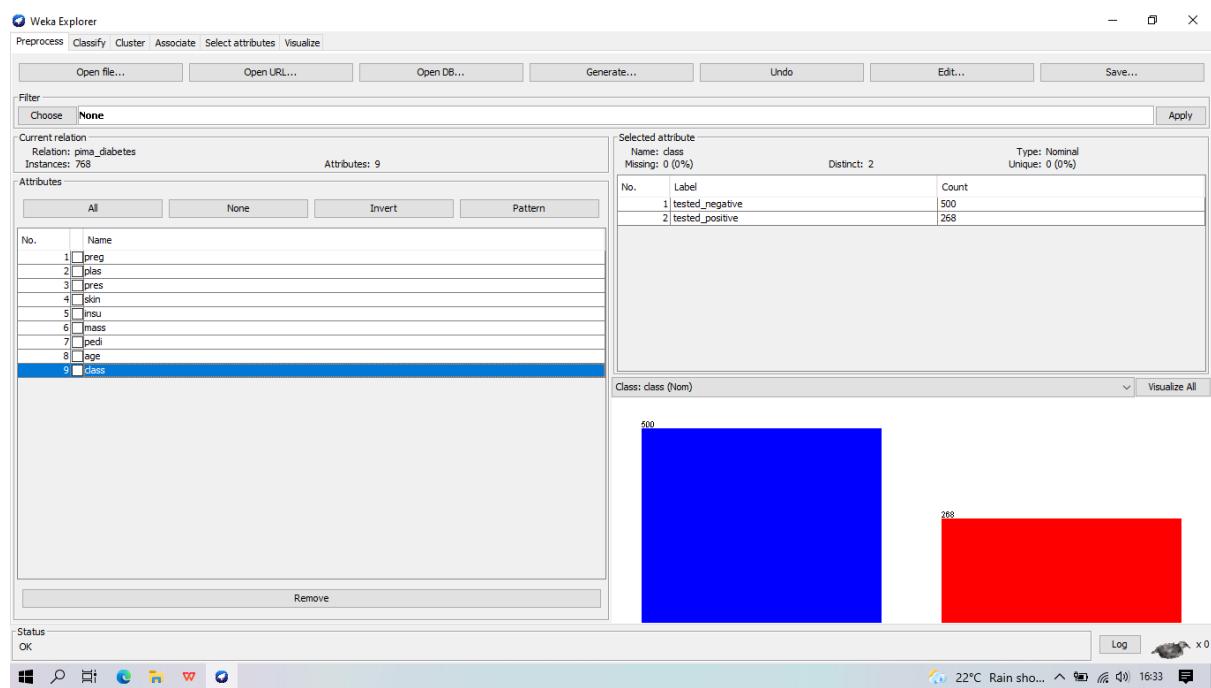
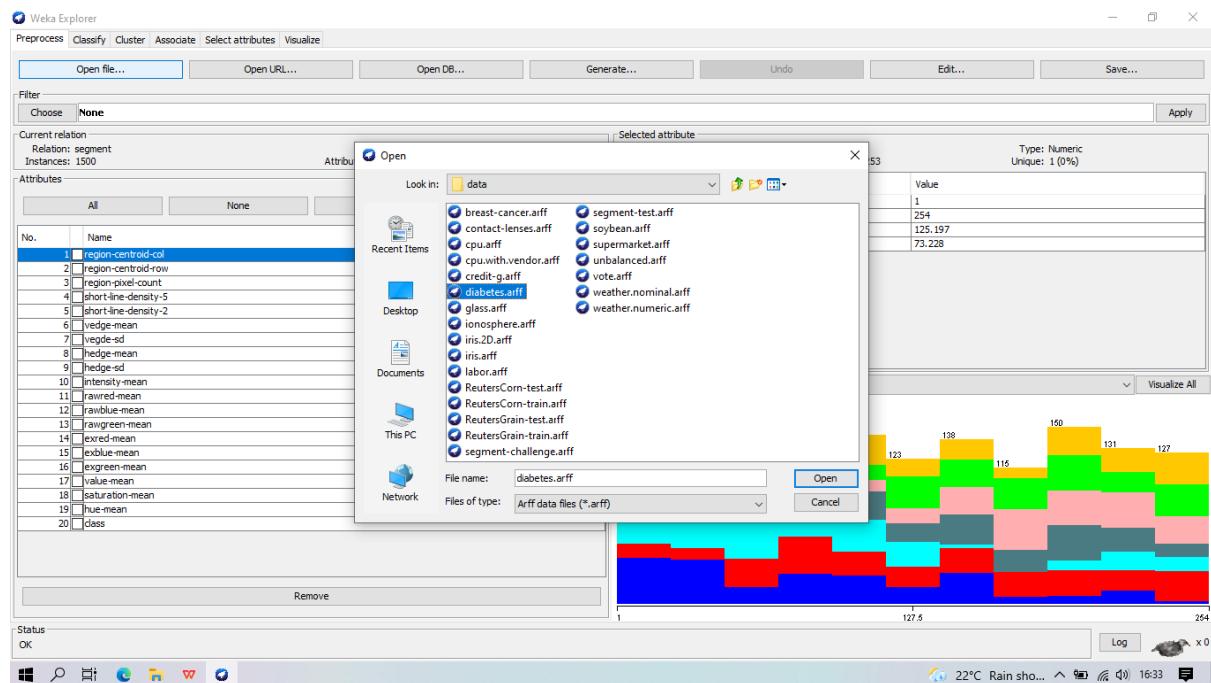
Sample mean	$\bar{x} = \frac{\sum x_i}{n}$	0.967
Variance	$\sigma^2 = \frac{\sum (x_i - \bar{x})^2}{n-1}$	0.940
Standard deviation	σ	0.940
		0.967
		0.920
		0.947
		0.933
		0.947
		$\bar{x} = 0.949, \sigma = 0.018$

Berdasarkan sampel data tersebut dapat dicari nilai rata-rata sampel dan standar deviasinya. Rata-rata akurasinya adalah 94,9% dan standar deviasinya adalah 1,8%. Sehingga dapat disimpulkan bahwa nilai akurasi berada diantara 93-97%.

2.4 Baseline Accuracy

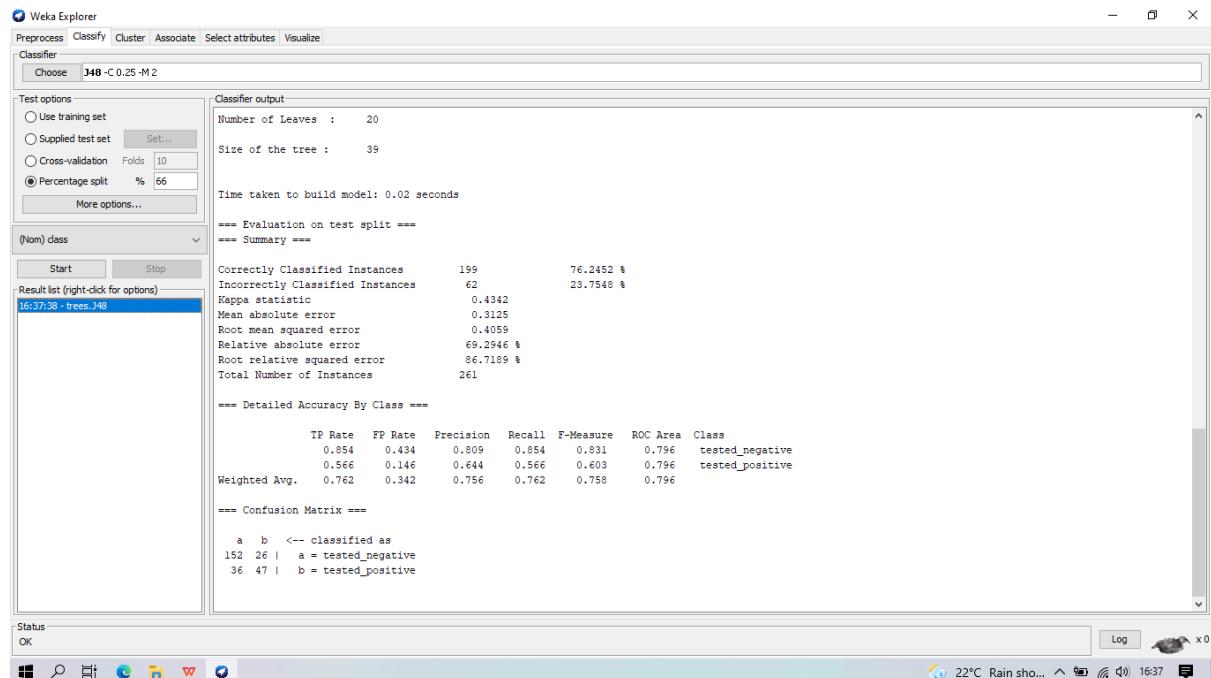
Dalam membuat model pada suatu dataset, jangan menggunakan classifier sembarang, perlu diperhatikan dataset yang digunakan dan selalu lakukan percobaan *baseline accuracy* (menggunakan classifier *ZeroR*). Dimana classifier yang akan digunakan nantinya harus memiliki akurasi lebih besar dari akurasi classifier *ZeroR*.

Load dataset diabetes.arff



Gunakan berbagai macam classifier untuk membuat model dari dataset diabetes. Sebagai contoh, gunakan classifier trees - J48, bayes - NaiveBayes, lazy - IBk, dan rules - PART. serta perlu juga dilakukan pengecekan *baseline accuracy* menggunakan ZeroR. (percentage split sebesar 66% dan random seed bernilai 1).

Berikut hasil yang didapatkan.



Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose **J48 - C 0.25 - M 2**

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)

16:37:38 - trees.J48

Classifier output

Number of Leaves : 20

Size of the tree : 39

Time taken to build model: 0.02 seconds

==== Evaluation on test split ===

==== Summary ===

	Correctly Classified Instances	199	76.2452 %
Incorrectly Classified Instances	62	23.7548 %	
Kappa statistic	0.4342		
Mean absolute error	0.3125		
Root mean squared error	0.4059		
Relative absolute error	69.2946 %		
Root relative squared error	86.7189 %		
Total Number of Instances	261		

==== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.854	0.434	0.809	0.854	0.831	0.796	0.796	tested_negative
0.566	0.146	0.444	0.566	0.603	0.796	0.796	tested_positive
Weighted Avg.	0.762	0.342	0.756	0.762	0.758	0.796	

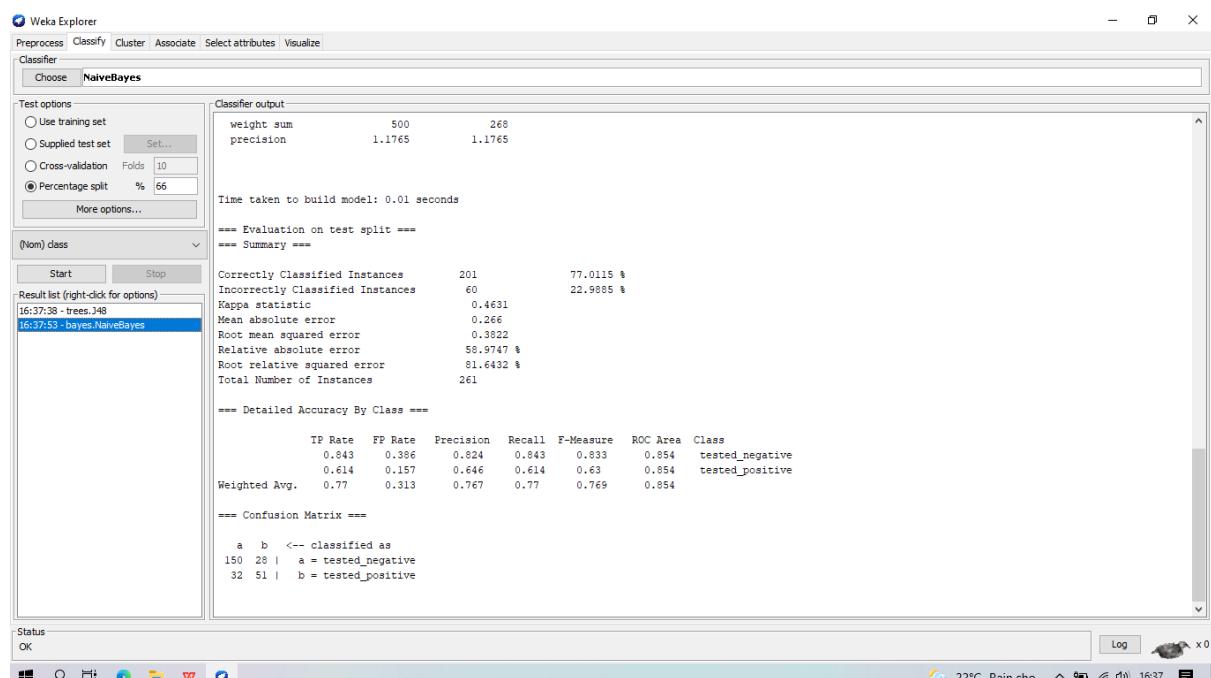
==== Confusion Matrix ===

	a	b	<- classified as
a	152	26	a = tested_negative
b	36	47	b = tested_positive

Status OK

22°C Rain sh... 16:37 Log x0

Menggunakan classifier J48 (Akurasi 76%)



Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose **NaiveBayes**

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)

16:37:38 - trees.J48

16:37:53 - bayes.NaiveBayes

Classifier output

	weight sum	500	268
precision	1.1765	1.1765	

Time taken to build model: 0.01 seconds

==== Evaluation on test split ===

==== Summary ===

	Correctly Classified Instances	201	77.0115 %
Incorrectly Classified Instances	60	22.9885 %	
Kappa statistic	0.4631		
Mean absolute error	0.266		
Root mean squared error	0.3822		
Relative absolute error	58.9747 %		
Root relative squared error	81.6432 %		
Total Number of Instances	261		

==== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.843	0.386	0.824	0.843	0.833	0.854	0.854	tested_negative
0.614	0.157	0.646	0.614	0.63	0.63	0.63	tested_positive
Weighted Avg.	0.77	0.313	0.767	0.77	0.769	0.854	

==== Confusion Matrix ===

	a	b	<- classified as
a	150	28	a = tested_negative
b	32	51	b = tested_positive

Status OK

22°C Rain sh... 16:37 Log x0

Menggunakan classifier NaiveBayes (Akurasi 77%)

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose: **IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -R "weka.core.EuclideanDistance -R first-last"**

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)

- 16:37:38 - trees.J48
- 16:37:53 - bayes.NaiveBayes
- 16:38:21 - lazy.IBk**

Classifier output

```

IB1 instance-based classifier
using 1 nearest neighbour(s) for classification

Time taken to build model: 0 seconds

*** Evaluation on test split ===
*** Summary ===

Correctly Classified Instances      190      72.7969 %
Incorrectly Classified Instances    71       27.2031 %
Kappa statistic                   0.3788
Mean absolute error               0.2729
Root mean squared error           0.5205
Relative absolute error           60.5137 %
Root relative squared error      111.2011 %
Total Number of Instances         261

*** Detailed Accuracy By Class ===

      TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area   Class
      0.792     0.41      0.806      0.792     0.799      0.691     tested_negative
      0.59      0.208     0.57       0.59      0.58       0.691     tested_positive
Weighted Avg.      0.728     0.345     0.731      0.728     0.729      0.691

*** Confusion Matrix ===

      a   b   <-- classified as
141  37 |  a = tested_negative
34   49 |  b = tested_positive

```

Status

OK

Log x0

22°C Rain sh... 16:38

Menggunakan classifier IBk (Akurasi 73%)

Weka Explorer

Preprocess Classify Cluster Associate Select attributes Visualize

Classifier

Choose: **PART -M 2 -C 0.25 -Q 1**

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)

- 16:37:38 - trees.J48
- 16:37:53 - bayes.NaiveBayes
- 16:38:21 - lazy.IBk
- 16:38:37 - rules.PART**

Classifier output

```

: tested_positive (252.0/105.0)

Number of Rules : 13

Time taken to build model: 0.02 seconds

*** Evaluation on test split ===
*** Summary ===

Correctly Classified Instances      194      74.3295 %
Incorrectly Classified Instances    67       25.6705 %
Kappa statistic                   0.4423
Mean absolute error               0.3056
Root mean squared error           0.4018
Relative absolute error           67.7579 %
Root relative squared error      85.8448 %
Total Number of Instances         261

*** Detailed Accuracy By Class ===

      TP Rate   FP Rate   Precision   Recall   F-Measure   ROC Area   Class
      0.758     0.289     0.849      0.758     0.801      0.792     tested_negative
      0.711     0.242     0.578      0.711     0.638      0.792     tested_positive
Weighted Avg.      0.743     0.274     0.763      0.743     0.749      0.792

*** Confusion Matrix ===

      a   b   <-- classified as
135  43 |  a = tested_negative
24   59 |  b = tested_positive

```

Status

OK

Log x0

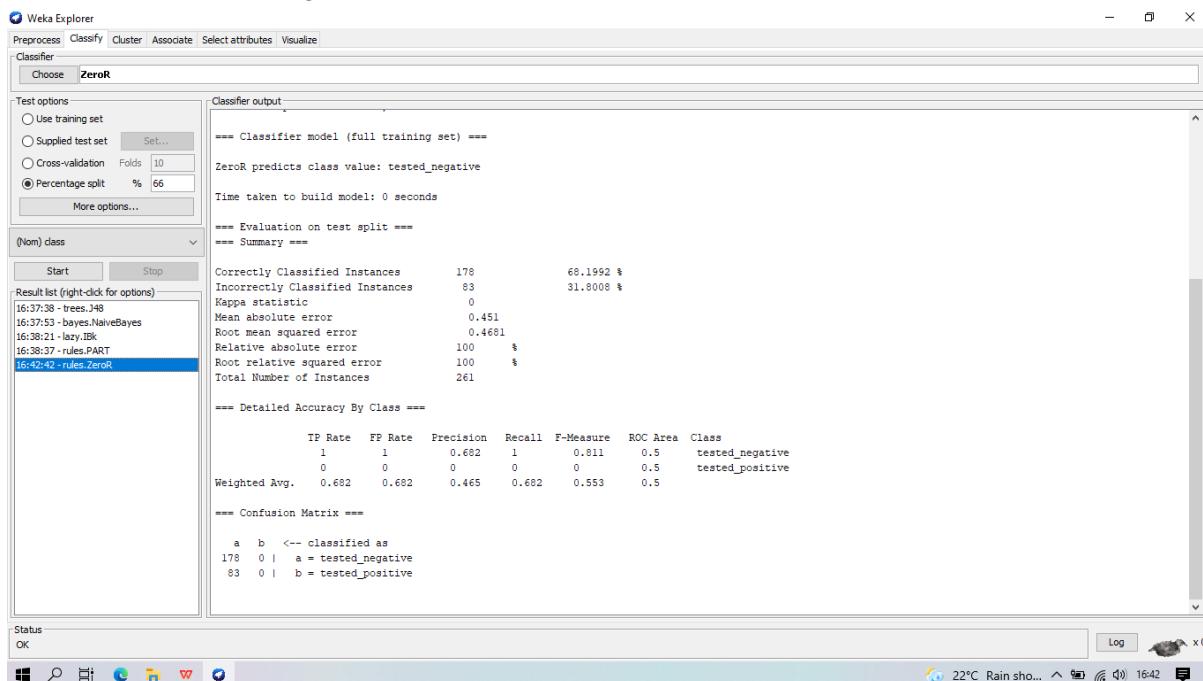
22°C Rain sh... 16:38

Menggunakan classifier PART (akurasi 74%)

Setelah dicoba setiap classifier, berikut akurasi yang dihasilkan

- **trees > J48** **76%**
- **bayes > NaiveBayes** **77%**
- **lazy > IBk** **73%**
- **rules > PART** **74%**

Kemudian lakukan pengecekan pada ZeroR.



The screenshot shows the Weka Explorer interface with the 'Classifier' tab selected. Under 'Choose', 'ZeroR' is selected. In the 'Test options' section, 'Cross-validation' is chosen with 'Folds' set to 10. The 'Result list (right-click for options)' shows several classifiers: 'trees.J48', 'bayes.NaiveBayes', 'lazy.IB', 'rules.PART', and '16:42:42 - rules.zeroR'. The 'Classifier output' pane displays the results for 'rules.zeroR'. It includes:

- Classifier model (full training set)
- ZeroR predicts class value: tested_negative
- Time taken to build model: 0 seconds
- Evaluation on test split
- Summary
- Correctly Classified Instances: 178 (68.1992 %)
- Incorrectly Classified Instances: 83 (31.8008 %)
- Kappa statistic: 0
- Mean absolute error: 0.451
- Root mean squared error: 0.4681
- Relative absolute error: 100 %
- Root relative squared error: 100 %
- Total Number of Instances: 261
- Detailed Accuracy By Class:

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
1	1	0.682	1	0.811	0.5	0.5	tested_negative
0	0	0	0	0	0.5	0.5	tested_positive
Weighted Avg.	0.682	0.682	0.465	0.682	0.553	0.5	

- Confusion Matrix:

		a	b
		-- classified as	
178		0	a = tested_negative
83		0	b = tested_positive

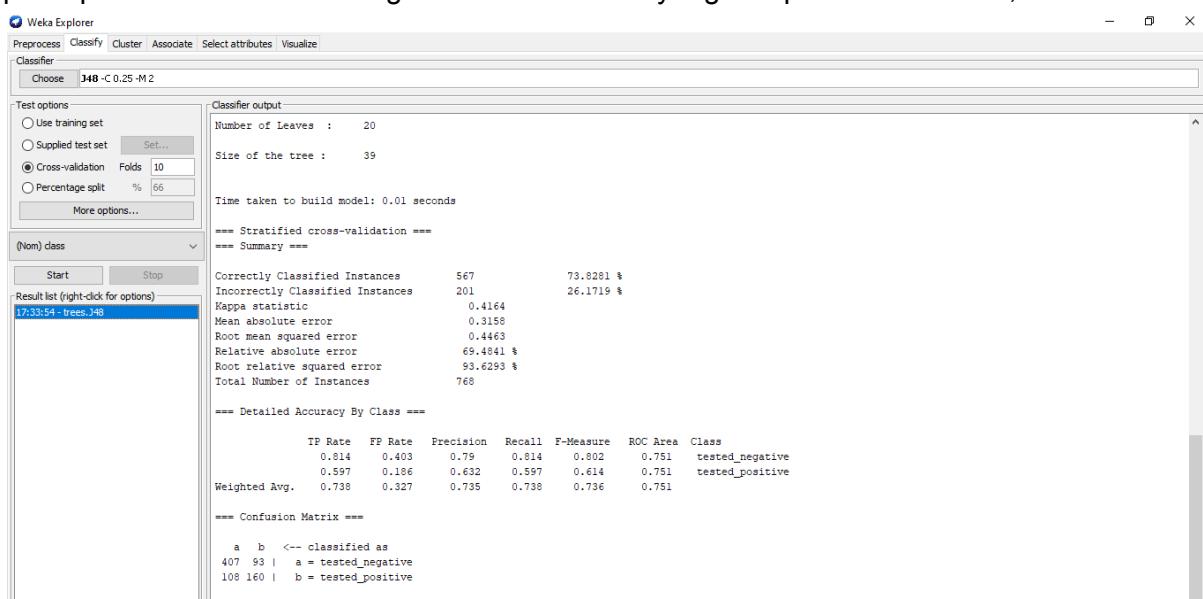
Akurasi yang didapatkan adalah 68%. Jika akurasi yang didapatkan menggunakan suatu classifier tertentu diatas akurasi ketika menggunakan classifier ZeroR, maka classifier tersebut bisa digunakan.

2.5 Cross-validation

Cross-validation (CV) adalah metode statistik yang dapat digunakan untuk mengevaluasi kinerja model atau algoritma dimana data dipisahkan menjadi dua subset. Cross-validation lebih baik dari repeated holdout. Hal tersebut akan dibuktikan pada praktik 2.6.

2.6 Cross-validation Results

Load dataset diabetes.arff, kemudian classifier J48 untuk membuat model. Pada test option, pilih opsi cross-validation dengan nilai 10. Akurasi yang didapatkan adalah 73,8%



The screenshot shows the Weka Explorer interface with the 'Classifier' tab selected. Under 'Choose', 'J48 - C 0.25 - M 2' is selected. In the 'Test options' section, 'Cross-validation' is chosen with 'Folds' set to 10. The 'Result list (right-click for options)' shows 'trees.J48'. The 'Classifier output' pane displays the results for 'trees.J48'. It includes:

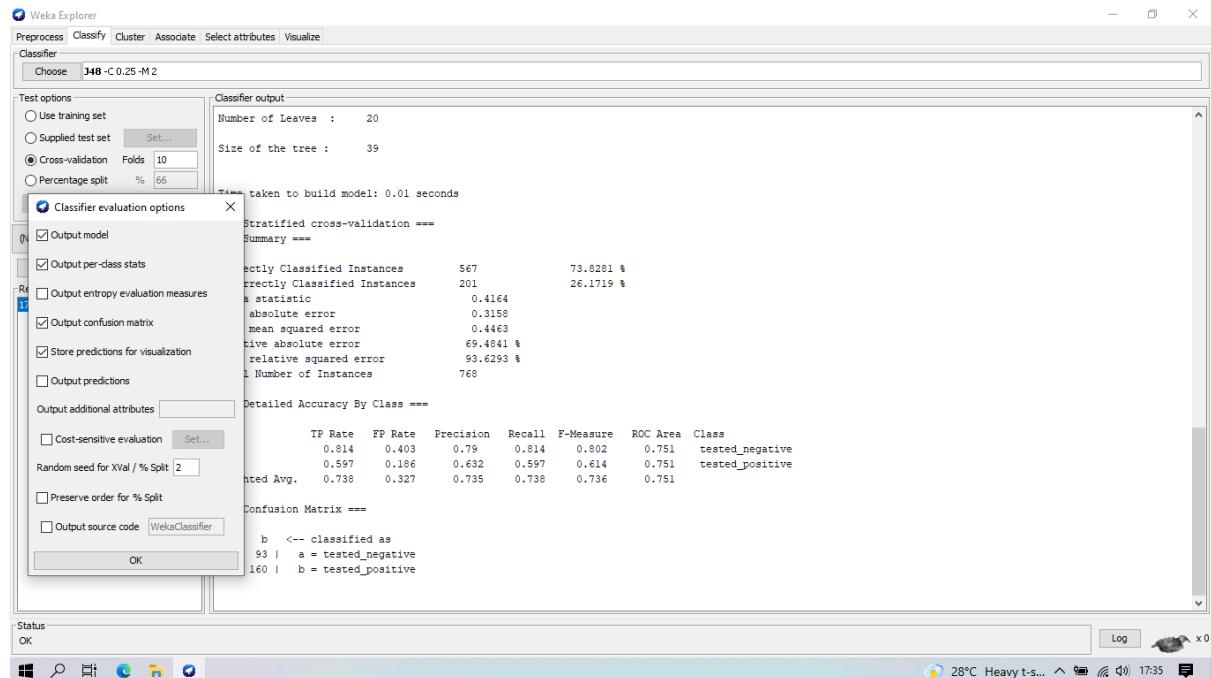
- Number of Leaves : 20
- Size of the tree : 39
- Time taken to build model: 0.01 seconds
- Stratified cross-validation
- Summary
- Correctly Classified Instances: 567 (73.8281 %)
- Incorrectly Classified Instances: 201 (26.1719 %)
- Kappa statistic: 0.4164
- Mean absolute error: 0.3158
- Root mean squared error: 0.4463
- Relative absolute error: 69.4841 %
- Root relative squared error: 93.6293 %
- Total Number of Instances: 768
- Detailed Accuracy By Class:

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0	0.814	0.403	0.79	0.814	0.802	0.751	tested_negative
1	0.597	0.186	0.632	0.597	0.614	0.751	tested_positive
Weighted Avg.	0.738	0.327	0.735	0.738	0.736	0.751	

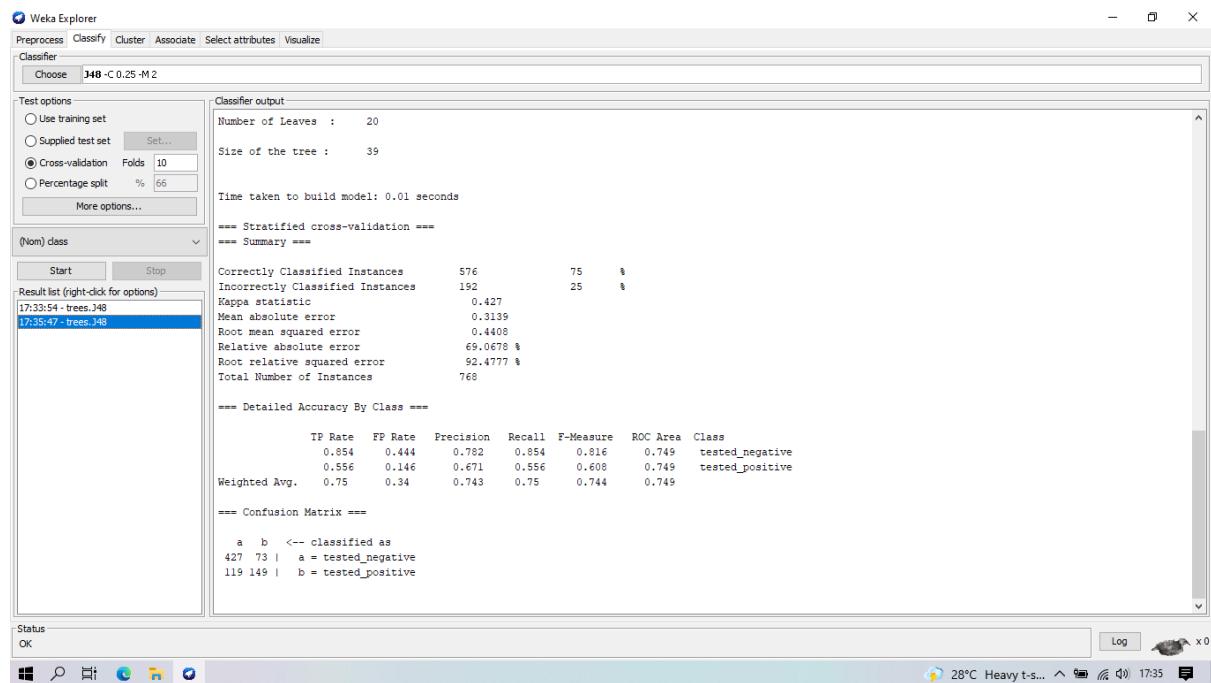
- Confusion Matrix:

		a	b
		-- classified as	
407		93	a = tested_negative
108		160	b = tested_positive

Tetap menggunakan classifier dan test option yang sama, lakukan perubahan pada random seed nya menjadi 2, 3, 4 dan seterusnya. Sebagai contoh, random cross-validation diubah menjadi 2.



Hasilnya mendapatkan akurasi sebesar 75%.



Setelah dicoba dengan random seed berbeda, dari 1 sampai 10, didapatkan data sebagai berikut beserta pembandingnya dengan metode repeated holdout.

		holdout (10%)	cross-validation (10-fold)
Sample mean	$\bar{x} = \frac{\sum x_i}{n}$	75.3 77.9 80.5 74.0	73.8 75.0 75.5 75.5
Variance	$\sigma^2 = \frac{\sum (x_i - \bar{x})^2}{n - 1}$	71.4 70.1 79.2 71.4	74.4 75.6 73.6 74.0
Standard deviation	σ	80.5 67.5	74.5 73.0
		$\bar{x} = 74.8$ $\sigma = 4.6$	$\bar{x} = 74.5$ $\sigma = 0.9$

Menggunakan cross-validation lebih baik daripada repeated holdout. Cross-validation memperkecil standar deviasi dari akurasi, dalam artian rentang akurasinya semakin kecil.