MODULE 9

CLASSIFICATION: DECISION TREE

Sort by :

Hamzah L200154013

\

Informatics Study Program

Faculty of Communication and Informatics

Muhammadiyah University of Surakarta

Praktikum Steps

Decision Tree Implementations using Weka.

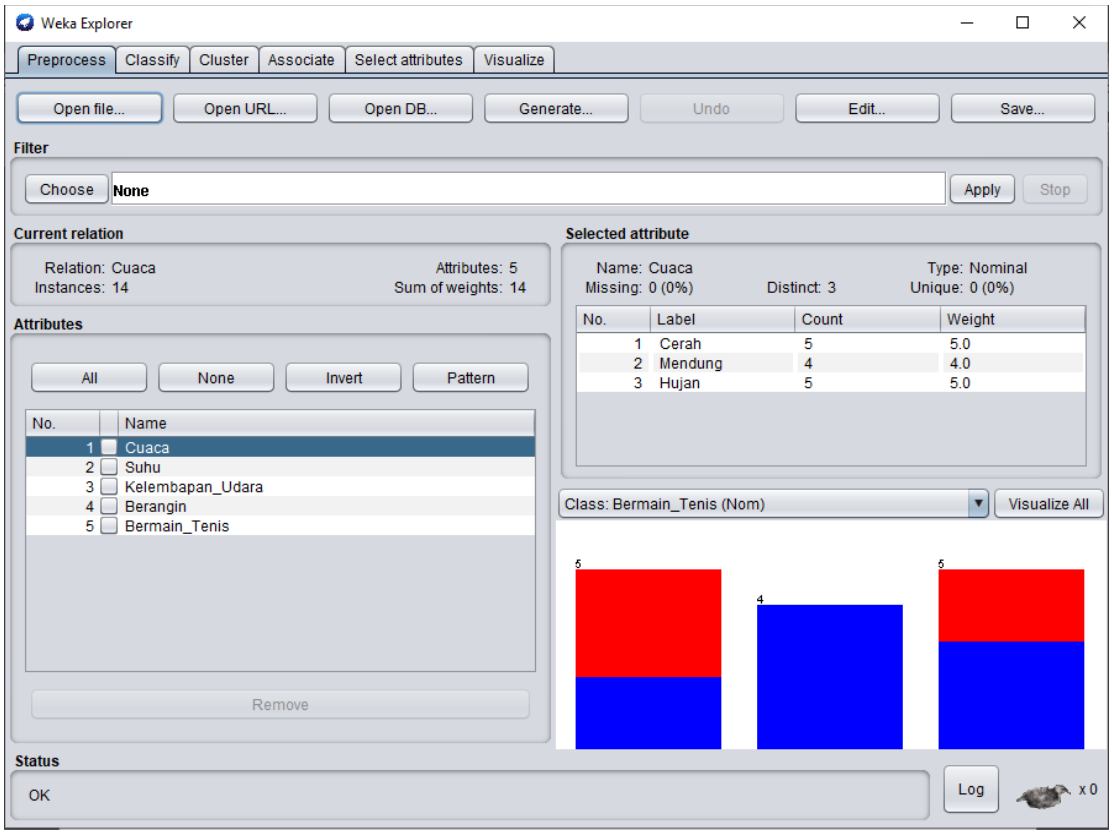
1. Prepare **Cuaca.arff** file in module 7, the file will be used as a training data.

Cuaca.arff

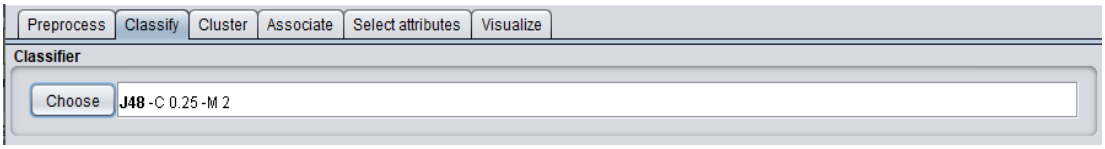
C: > Users > Hamzah > Documents > Hamzah Backup > Kuliah >

```
@relation Cuaca

@attribute Cuaca {Cerah, Mendung, Hujan}
@attribute Suhu real
@attribute Kelembapan_Udara real
@attribute Berangin {YA, TIDAK}
@attribute Bermain_Tenis {YA, TIDAK}

@data
Cerah, 85, 85, TIDAK, TIDAK
Cerah, 80, 90, YA, TIDAK
Mendung, 83, 86, TIDAK, YA
Hujan, 70, 96, TIDAK, YA
Hujan, 68, 80, TIDAK, YA
Hujan, 65, 70, YA, TIDAK
Mendung, 64, 65, YA, YA
Cerah, 72, 95, TIDAK, TIDAK
Cerah, 69, 70, TIDAK, YA
Hujan, 75, 80, TIDAK, YA
Cerah, 75, 70, YA, YA
Mendung, 72, 90, YA, YA
Mendung, 81, 75, TIDAK, YA
Hujan, 71, 91, YA ,TIDAK
```

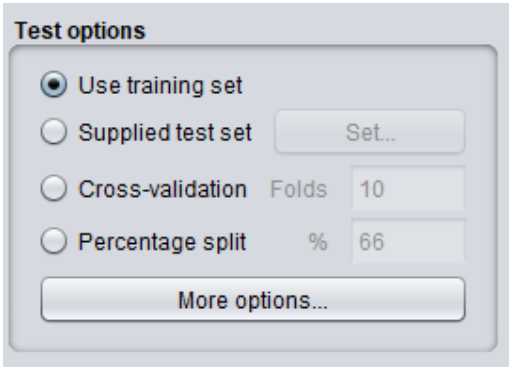
2. Open file **Cuaca.arff** using Weka – Explorer in tab preprocessing.



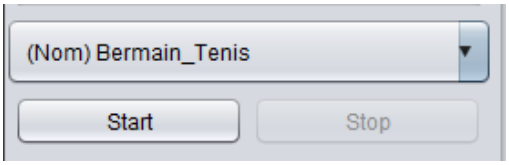
3. Open **Classify** tab, in classifier column open tree folder and choose J48.



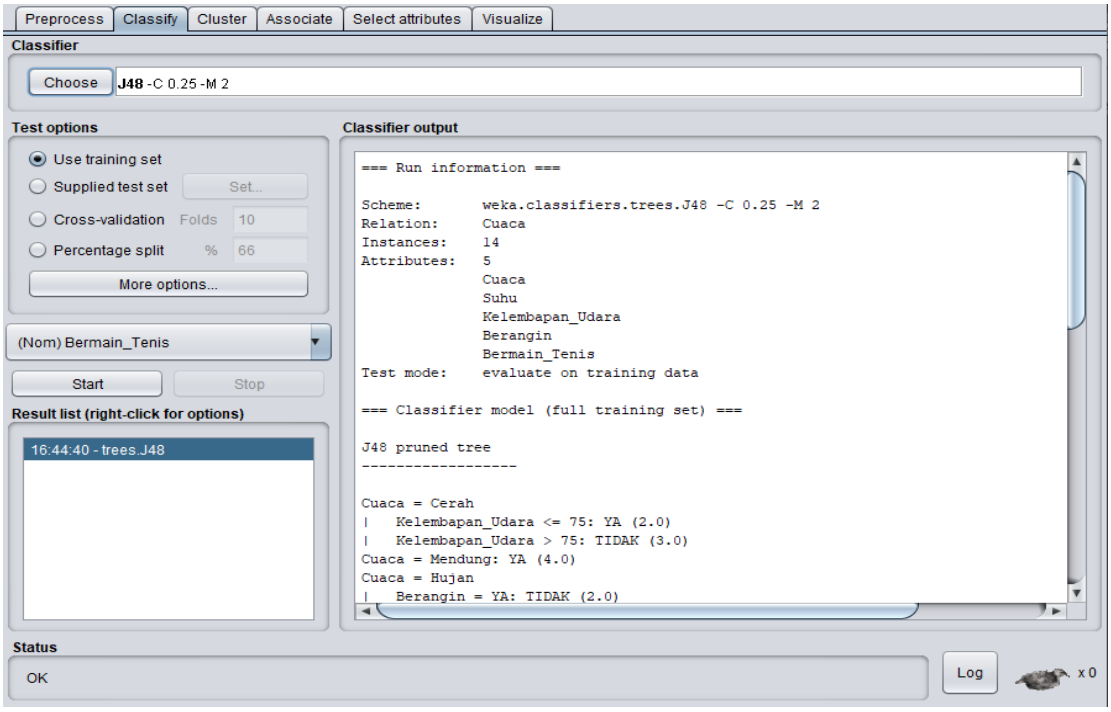
4. In **Test Options** column choose **Use Training Set**. Cuaca.arff data will be used as data training to created classification.



5. Make sure dependant attribute is **Bermain_Tenis**. Then click start.



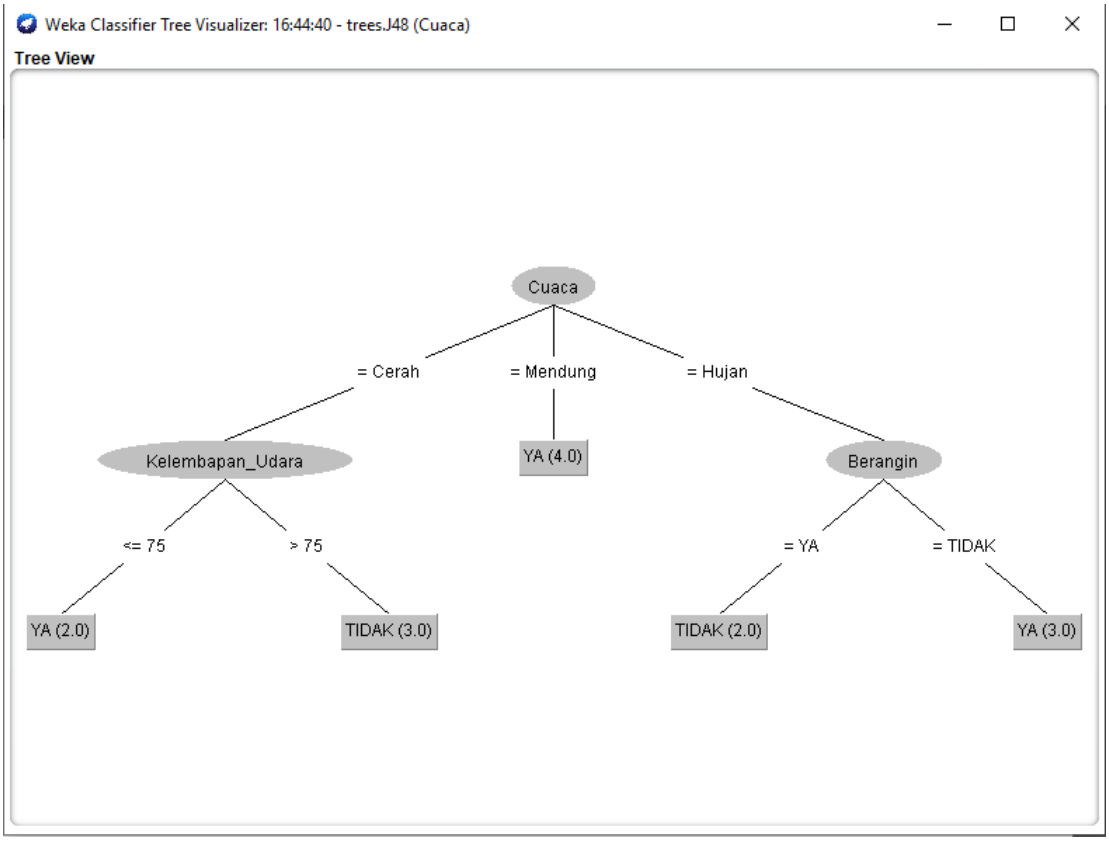
6. The result will be shown like this picture below.



```
Cuaca = Cerah
|   Kelembapan_Udara <= 75: YA (2.0)
|   Kelembapan_Udara > 75: TIDAK (3.0)
Cuaca = Mendung: YA (4.0)
Cuaca = Hujan
|   Berangin = YA: TIDAK (2.0)
|   Berangin = TIDAK: YA (3.0)

Number of Leaves  :     5

Size of the tree :      8


Time taken to build model: 0.01 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds

=== Summary ===

Correctly Classified Instances          14               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances               14
```

7. From the result we can take conclusion that :
   a. Number of leaves in the decision tree = 5
   b. Size of the tree = 8
   c. Time takes to build the model = 0.01 seconds
   d. Correctly classified instance = 100%
   e. Incorrectly classified instance = 0%

8. To see the visualization of the tree, right click **trees.j48** in result list then choose visualize tree. The tree visualization will look like the picture below.

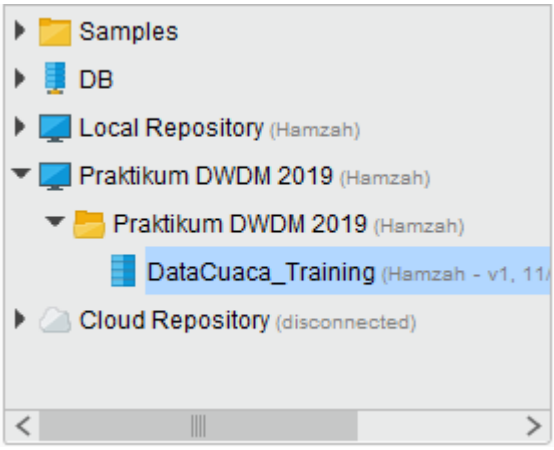


9. From the decision tree above, we can see several knots, there are
   a. Root knot = Cuaca
   b. Internals knot = Kelembapan_Udara and Berangin
   c. Leaves knot = YA, TIDAK

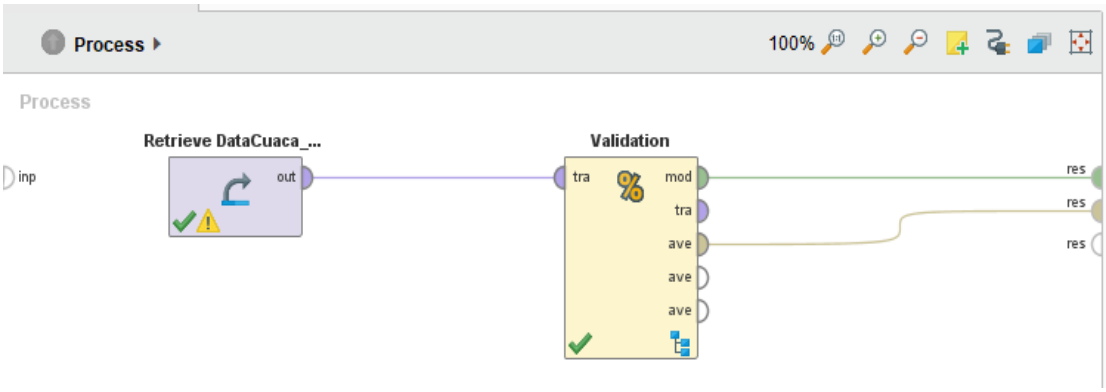
10. The classification conclusion is :
    a. Someone will be playing tennis if this condition fulfilled :
       i. Cuaca = Cerah, Kelembapan_Udara <= 5, (Others attribute value will be ignored).
       ii. Cuaca = Mendung, (Others condition will be ignored).
       iii. Cuaca = Hujan, Berangin = TIDAK, (Other attribute value will be ignored).
    b. Someone will not playing tennis if this condition fulfilled :
       i. Cuaca = Cerah, Kelembapan_Udara > 75, (The value of other attributes will be ignored)
       ii. Cuaca = Hujan, Berangin = YA, (Other attributes values will be ignored).

Decision Tree Implementation using RapidMiner

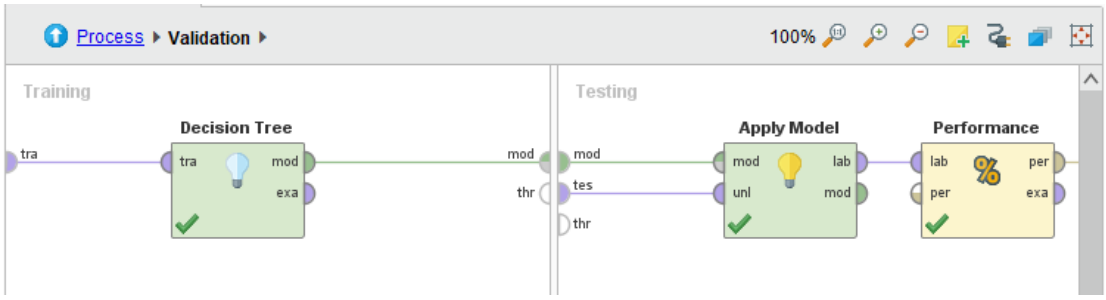
1. Open RapidMiner and start new blank process.
2. Import data training that we already made in previous module (DataCuaca_Training).



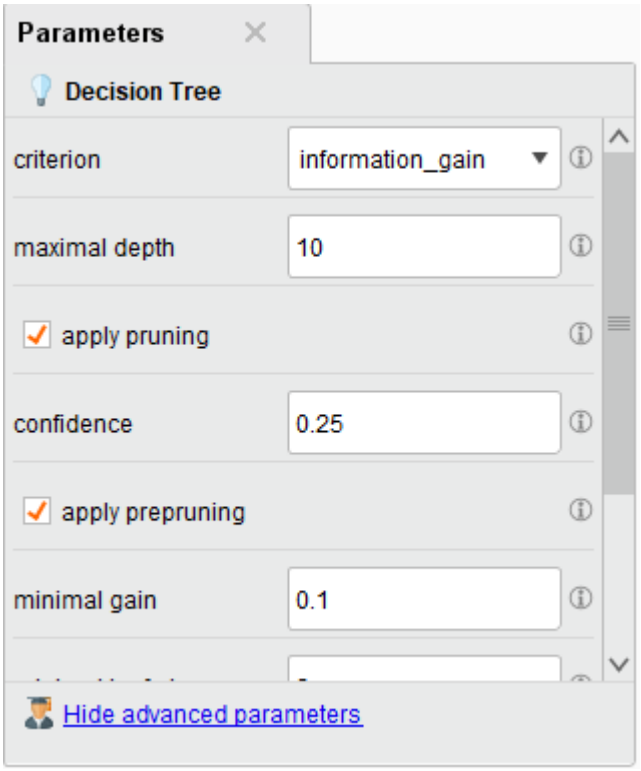
3. Drag DataCuaca_Training in Process View area, then search cross validation (x-validation in newest RapidMiner) drag it to Process View area too and connect the port like the picture below.



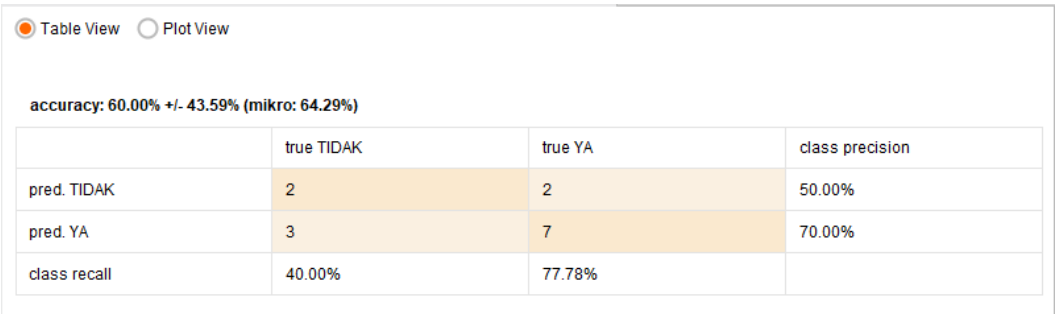
4. The next step is to make decision tree algorithm. Double click in Cross Validation (Validation) in the Process View Area. Drag operator Decision Tree, Apply Model and Perfomance arrange the operator like the picture below.



5. Click on Decision Tree operator and make sure the creation in parameters columns set as information_gain (additional : maximal depth = 10). Then run the process

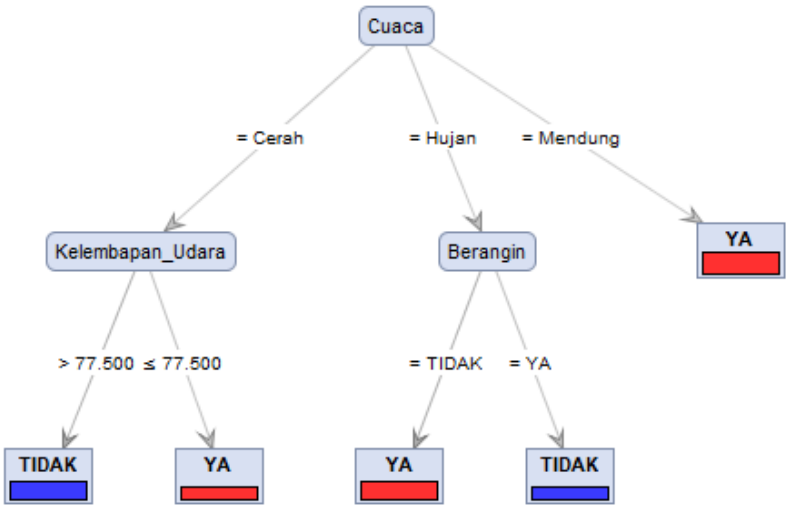


6. There will be 2 result of process classification :
   a. PerformanceVector (Performance), show the level of accuracy, precision, recall and any others in the form of table and plot view.



	true TIDAK	true YA	class precision
pred. TIDAK	2	2	50.00%
pred. YA	3	7	70.00%
class recall	40.00%	77.78%	

b. Tree (Decision Tree), show decision tree result from the classification process. The decision tree that produced is similar when using weka application.



Task

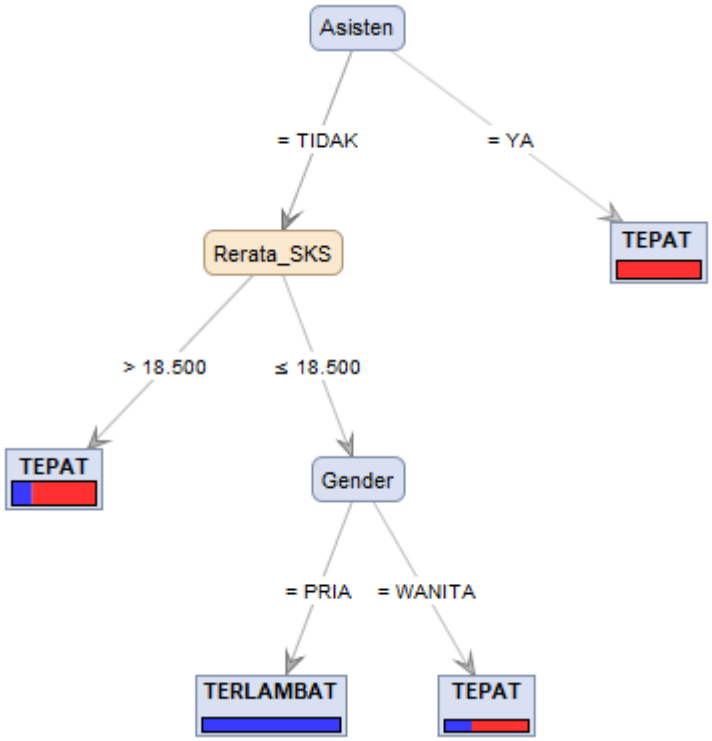
1. Using decision tree in activity 9.4.2 (using RapidMiner). The value of Bermain_Tenis will be :

Cuaca	Suhu	Kelembapan_Udara	Berangin	Bermain_Tenis
Cerah	75	65	TIDAK	YA
Cerah	80	68	YA	YA
Cerah	83	87	YA	TIDAK
Mendung	70	96	TIDAK	YA
Mendung	68	81	TIDAK	YA
Hujan	65	75	TIDAK	YA
Hujan	64	85	YA	TIDAK

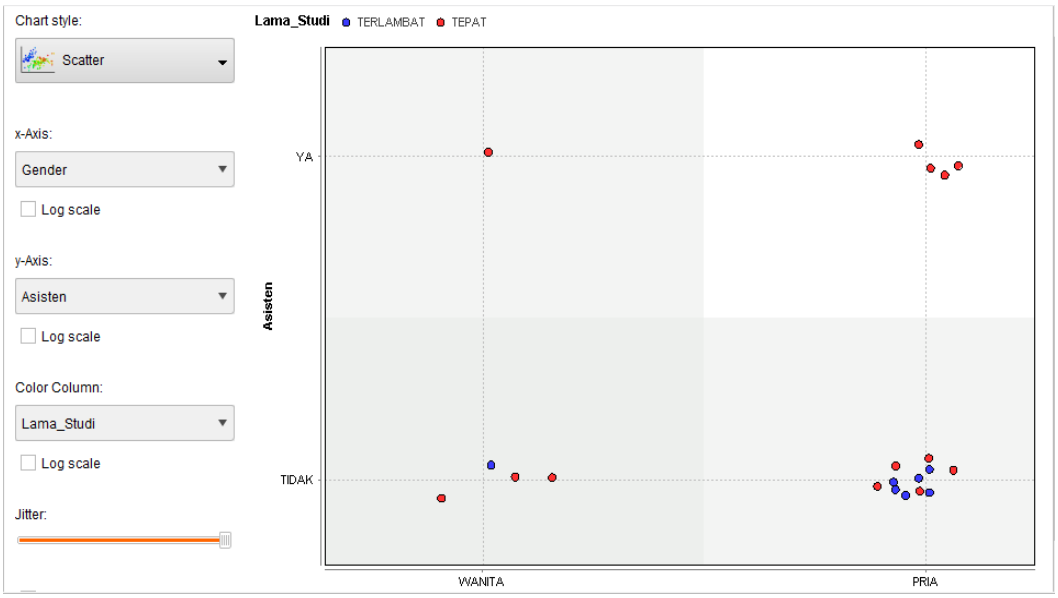
2. From ARFF file in module 7 number 1, the classifaction result of :
   i. Number of leaves in the decision tree = 5
   ii. Size of the tree = 8
   iii. Time takes to build the model = 0.01 seconds
   iv. Correctly classified instance = 100%
   v. Incorrectly classified instance = 0%

```
Number of Leaves  :     3

Size of the tree :      5


Time taken to build model: 0.01 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds

=== Summary ===

Correctly Classified Instances          17               85      %
Incorrectly Classified Instances         3               15      %
Kappa statistic                          0.6341
Mean absolute error                      0.2436
Root mean squared error                  0.349
Relative absolute error                 53.0693 %
Root relative squared error             73.1456 %
Total Number of Instances               20
```

3. From file excel that we made in module 6 number 1 as data training, the result of decision tree classification is :
   a. Decision Tree Result



   b. Perspective Plot View with scatter model, X-Axis = Gender, Y-Axis = Asisten, dan Color Column = Lama_Studi.



4. Classification result that formed based on their condition :
    a. Someone will graduate on time if this condition fulfilled :
        i. Asisten = YA, (Others attribute value will be ignored).
        ii. Asisten = TIDAK, Rerata_SKS = > 18.5 (Others condition will be ignored).
        iii. Asisten = TIDAK, Rerata_SKS = <= 18.5, Gender = Wanita (Other attribute value will be ignored).
    b. Someone will graduate late if this condition fulfilled :
        i. Asisten = TIDAK, Rerata_SKS = <= 18.5, Gender = Pria (The value of other attributes will be ignored).