**Data Warehousing and Data Mining Lab**

|  |  |
| --- | --- |
| **NAME** | **A.ANANTH KUMAR** |
| **REG. NO.** | **19Y003** |
| **ROLL NO.** | **62CA025** |
| **SUBJECT CODE** | **17CA380** |

**Experiment-1**

**Aim:**

To perform various commands in PL/SQL .

**Tools:** Oracle 10g

**Procedure:**

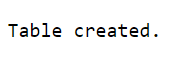
1) Open SQL 8.0 tool and it will display the login window.

2) Enter the username = ‘SCOTT’ and password = ‘TIGER’

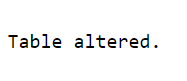
3) After the successfully login try to write down the SQL queries in correct syntax and run them successfully.

**Output:**

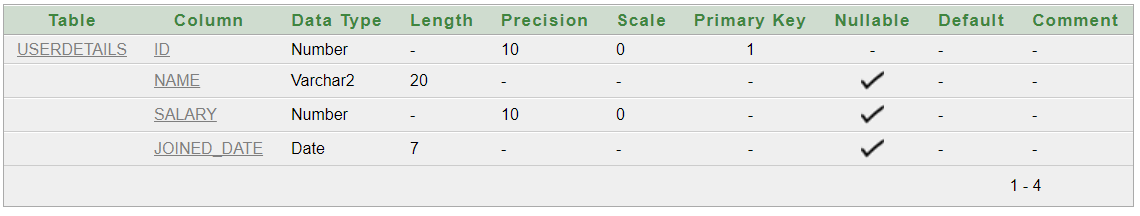
create table userdetails(id number(10) primary key, name varchar2(20));



alter table userdetails add (salary number(10), joined\_date date);



desc userdetails;



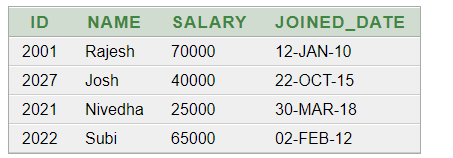
insert into userdetails values(2001, 'rajesh', 70000, '12-jan-10');

insert into userdetails values(2027, 'josh', 40000, '22-oct-15');

insert into userdetails values(2021, 'nivedha', 25000, '30-mar-18');

insert into userdetails values(2022, 'subi', 65000, '02-feb-12');

select \*from userdetails;

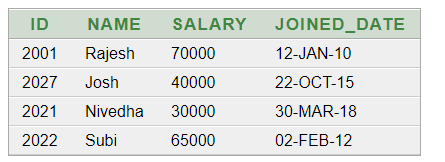


update userdetails

set salary = '30000'

where name = 'nivedha'

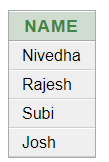
select \*from userdetails;



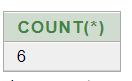
insert into userdetails values(2076, 'nivedha', 10000, '10-jun-20');

insert into userdetails values(2016, 'rajesh', 20000, '27-jun-19');

select unique name from userdetails;



select count(\*) from userdetails;



select sum(salary) as "salary"

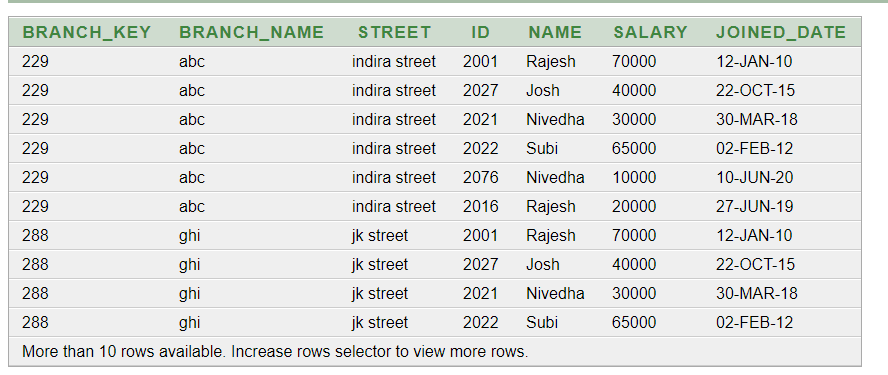
from userdetails

where salary > 50000

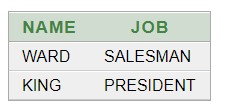
group by joined\_date;



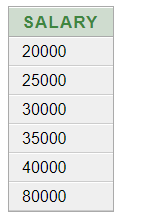
select \* from branch\_dimension cross join userdetails

****

select name,job from employee\_details e where salary>(select avg(salary)from salary s where e.id=s.id);



select salary from employee\_details group by salary having count(distinct salary)>=all(select count(distinct salary)from employee\_details group by salary);



**Result:**

The given SQL commands were successfully executed.

**Experiment-2**

**Aim:**

To perform multi-dimensional data model using SQL queries. E.g. Star, snowflake and Fact constellation schemas.

**Tools:** Oracle 10g

**Procedure:**

1) Open Oracle 8i/9i/10G tool and it will display the login window where one has to enter the login details.

2) Create the fact and the required dimensions tables as per the given business problem. There are three basic types of the multidimensional data model. They are Star, snowflake and Fact constellation schemas

3) As per the guidelines given in the theory draw all the three dimensional models.

Use the following queries : *Define cube sales\_star[time, item, branch, location];*

*Dollars\_sold = sum(sales\_in\_dollars), units\_sold = count(\*).*

*Define dimension time as(time\_key,day,day\_of\_week, month, quarter, year).*

Write the same queries for all other dimensions resp. Then run the following query:

Select s.time\_key,s.item\_key, s.branch\_key, s.location\_key,

Sum(s.number\_of\_units\_sold\*s.price), sum(s.number\_of\_units\_sold)

From time t,item I, branch b, location l, sales s,

Where s.time\_key = t.time\_key and s.item\_key = i.item\_key and s.branch\_key = b.branch\_key and s.location\_key = l.location\_key Group by s.time\_key, s.item\_key, s.branch\_key, s.location\_key.

Run the queries to create the various multi-dimensional models.

**Output:**

**Sales\_Fact:**

create table sales\_fact(item\_key number(5), time\_key number(5), branch\_key number(5), location\_key(5), no\_of\_units\_sold number(5), price\_per\_unit number(5));

insert into sales\_fact(901, 100, 229, 3343, 5, 100);

insert into sales\_fact(902, 286, 244, 3362, 6, 100);

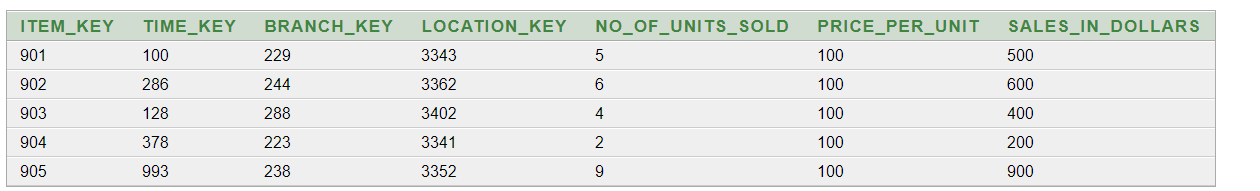
insert into sales\_fact(903, 128, 288, 3402, 4, 100);

insert into sales\_fact(904, 378, 223, 3341, 2, 100);

insert into sales\_fact(905, 993, 238, 3352, 9, 100);

update sales\_fact set sales\_in\_dollars = no\_of\_units\_sold \* price\_per\_unit;

select \*from sales\_fact ;



**Time\_Dimension:**

create table time\_dimension( time\_key number(5), day varchar2(10), day\_of\_week number(5), month varchar2(4), quarter varchar2(7), year number(4));

insert into time\_dimension values(100, 'monday', 1, 'feb', 'first', 2012);

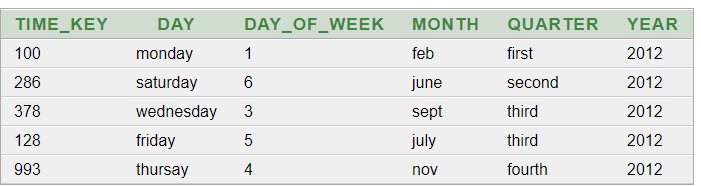
insert into time\_dimension values(286, 'saturday', 6, 'june', 'second', 2012);

insert into time\_dimension values(128, 'friday', 5, 'july', 'third', 2012);

insert into time\_dimension values(378, 'wednesday', 3, 'sept', 'third', 2012);

insert into time\_dimension values(993, 'thursay', 4, 'nov', 'fourth', 2012);

select \*from time\_dimension;



**Item\_Dimension:**

create table item\_dimension( item\_key number(5), item\_type varchar2(20), number\_of\_items\_supplied number(10));

insert into item\_dimension values(901, 'sarees', '50');

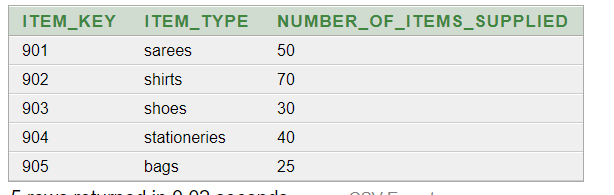
insert into item\_dimension values(902, 'shirts', '70');

insert into item\_dimension values(903, 'shoes', '30');

insert into item\_dimension values(904, 'stationeries', '40');

insert into item\_dimension values(905, 'bags', '25');

select \*from item\_dimension;



**Branch\_Dimension:**

create table branch\_dimension( branch\_key number(5), branch\_name varchar2(10), street varchar2(20));

insert into branch\_dimension values(229, 'abc', 'indira street');

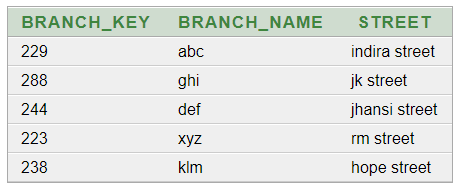
insert into branch\_dimension values(244, 'def', 'jhansi street');

insert into branch\_dimension values(288, 'ghi', 'jk street');

insert into branch\_dimension values(223, 'xyz', 'rm street');

insert into branch\_dimension values(238, 'klm', 'hope street');

select \*from branch\_dimension;



**Location\_Dimension:**

create table location\_dimension( location\_key number(5), city varchar2(20));

insert into location\_dimension values(3343, 'tiruchy');

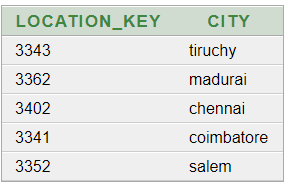
insert into location\_dimension values(3362, 'madurai');

insert into location\_dimension values(3402, 'chennai');

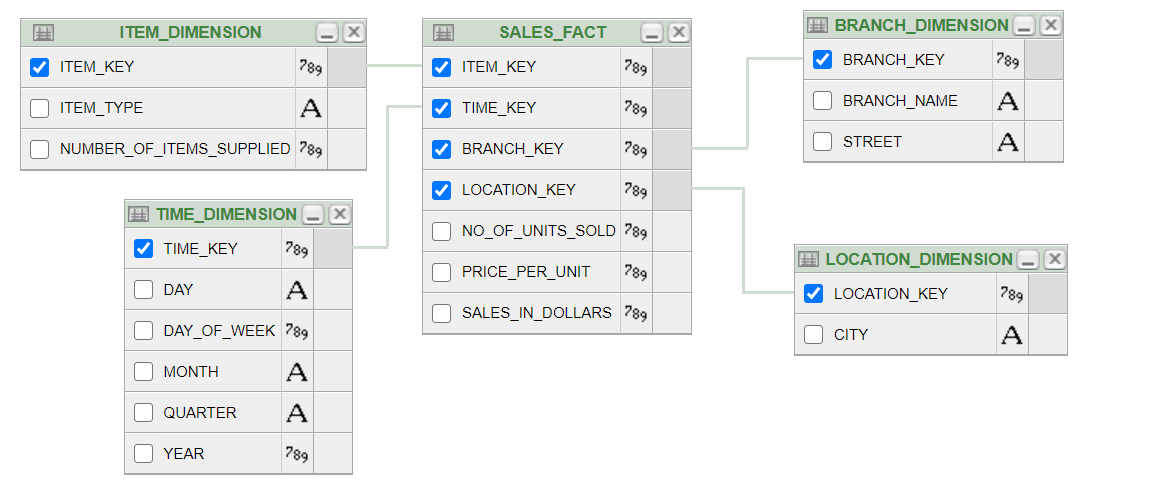
insert into location\_dimension values(3341, 'coimbatore');

insert into location\_dimension values(3352, 'salem');

select \*from location\_dimension;



**Star Schema:**



**Supply\_Dimension:**

create table supply\_dimension( supply\_key number(5), item\_type varchar2(20), number\_of\_items\_supplied number(10));

insert into supply\_dimension values(1901, 'sarees', '50');

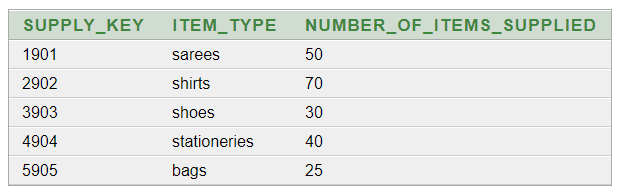
insert into supply\_dimension values(2902, 'shirts', '70');

insert into supply\_dimension values(3903, 'shoes', '30');

insert into supply\_dimension values(4904, 'stationeries', '40');

insert into supply\_dimension values(5905, 'bags', '25');

select \*from supply\_dimension;



**City\_Dimension:**

create table city\_dimension( city\_key number(5), city varchar2(20), state varchar2(15));

insert into city\_dimension values(632, 'tiruchy', 'tamilnadu');

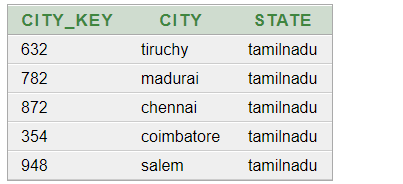
insert into city\_dimension values(782, 'madurai', 'tamilnadu');

insert into city\_dimension values(872, 'chennai', 'tamilnadu');

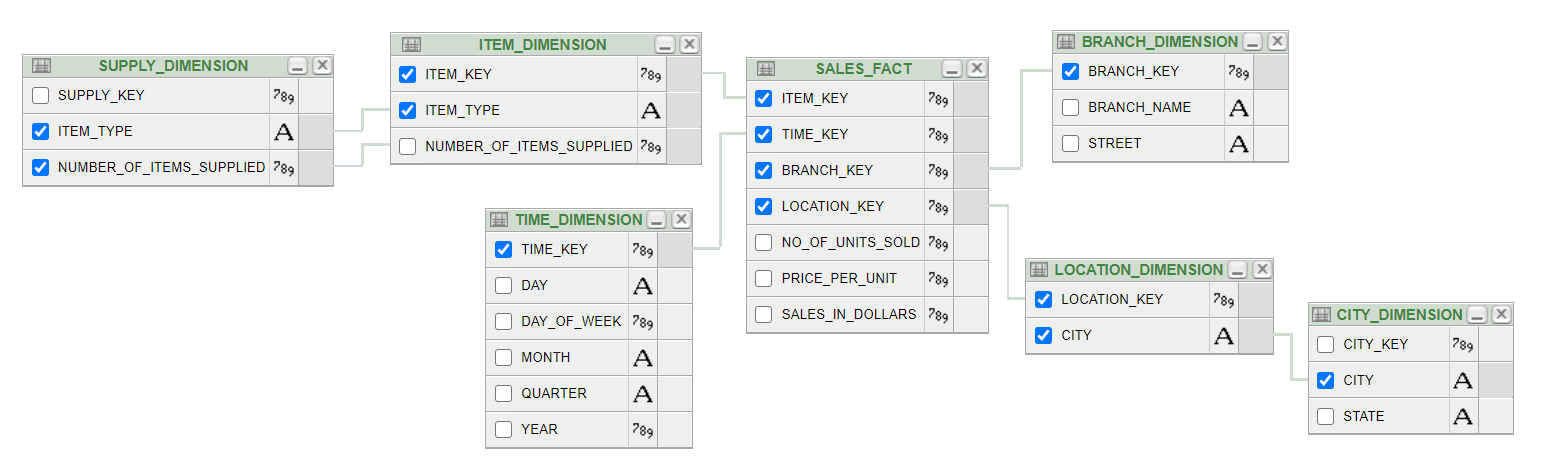
insert into city\_dimension values(354, 'coimbatore', 'tamilnadu');

insert into city\_dimension values(948, 'salem', 'tamilnadu');

select \*from city\_dimension;



**Snow Flake Schema:**



**Shipping\_Fact:**

create table shipping\_fact(time\_key number(5), item\_key number(5), shipper\_key number(5), from\_location varchar2(20), to\_location varchar2(20), cost\_in\_dollars number(5), unit\_shipped number(5));

insert into shipping\_fact values(100, 901, 656, 'chennai', 'tiruchy',90, 10);

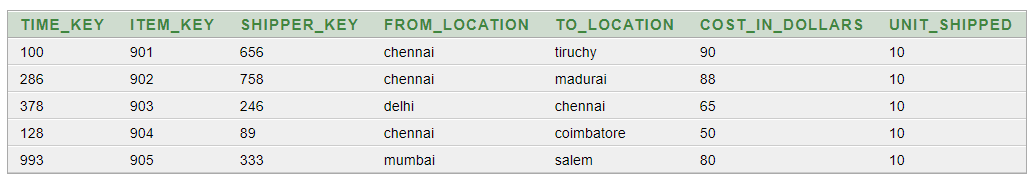
insert into shipping\_fact values(286, 902, 758,'chennai', 'madurai',88, 10);

insert into shipping\_fact values(378, 903, 246, 'delhi','chennai',65, 10);

insert into shipping\_fact values(128, 904, 089,'chennai', 'coimbatore',50, 10);

insert into shipping\_fact values(993, 905, 333, 'mumbai','salem', 80, 10);

select \*from shipping\_fact;



**Shipper\_Dimension:**

create table shipper\_dimension(shipper\_key number(5), shipper\_name varchar2(15), location\_key number(5), shipper\_type varchar2(10));

insert into shipper\_dimension values(656, 'brandyuva', 3343, 'regular');

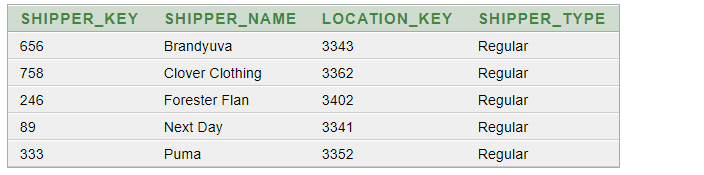
insert into shipper\_dimension values(758, 'clover clothing', 3362, 'regular');

insert into shipper\_dimension values(246, 'forester flannels', 3402, 'regular');

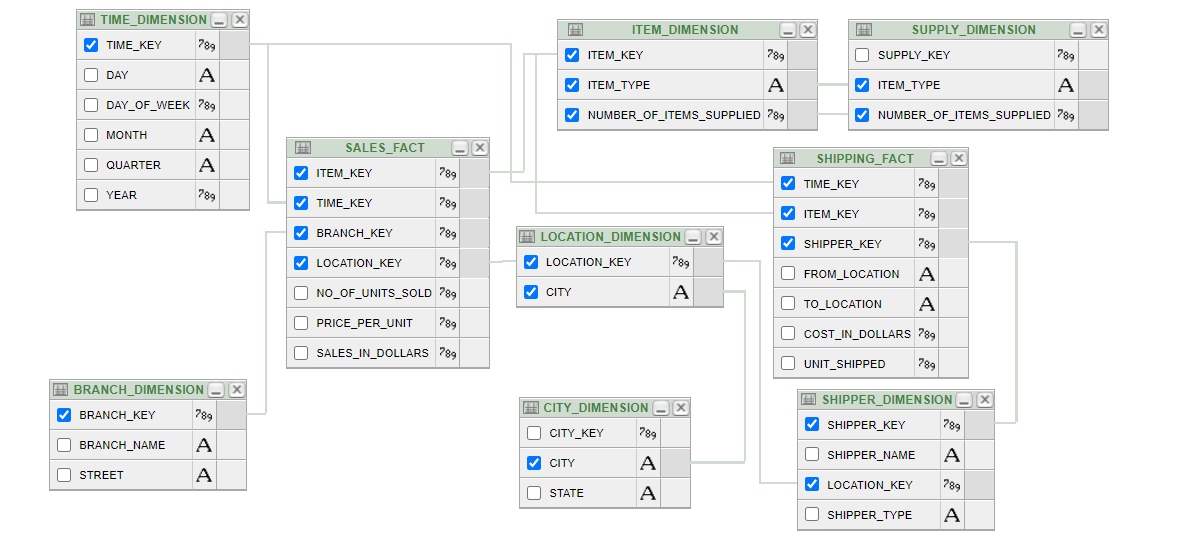
insert into shipper\_dimension values(089, 'coimbatore', 3341, 'regular');

insert into shipper\_dimension values(333, 'puma', 3352, 'regular');

select \*from shipper\_dimension;



**Fact Schema:**



**Result:**

Multi-dimensional data model using SQL queries. E.g. Star, snowflake and Fact constellation schemas is made successfully.

**Experiment-3**

**Aim:**

To perform various OLAP operations such slice, dice, roll up, drill up, pivot.

**Tools:** Oracle 10g

**Procedure:**

1) Open SQL tool and login successfully.

2) Write down the queries to perform slice. In which one should keep one of the

dimensions as constant and other dimensions should range from min to max.

3) Write down the queries to perform the dice. In which one has to keep two of the

dimensions constant.

4) Write down the queries to perform roll-up by keeping one dimension constant and others should range from min to max. It is more like a generalization.

5) Write down the queries to perform roll-up by keeping one dimension constant and others should range from min to max. It is more like a specialization.

**Output:**

create table employee\_details(id integer primary key, name varchar2(20), job varchar2(20), salary number(10));

insert into employee\_details values(7369, 'smith','clerk', 20000);

insert into employee\_details values(7566, ' jones', ‘manager ' ,40000);

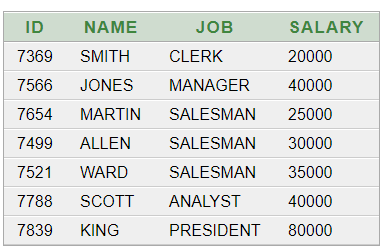
insert into employee\_details values(7654, ' martin', ‘salesman ' ,25000);

insert into employee\_details values(7499, ' allen', ‘salesman' ,30000);

insert into employee\_details values(7521, ' ward', ‘salesman' ,35000);

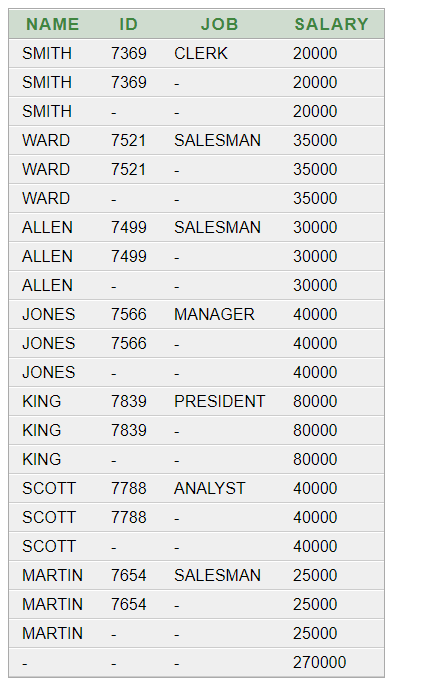
insert into employee\_details values(7788, ' scott', ‘analyst' ,40000);

insert into employee\_details values(7839, ' king ',’president' ,80000);



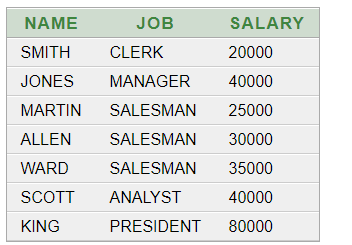
**Roll-up:**

select name, id , job, sum(salary) as salary from employee\_details group by rollup(name, id , job);



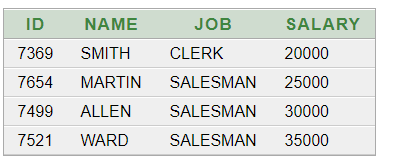
**Drill-up:**

select name, job, salary from employee\_details;



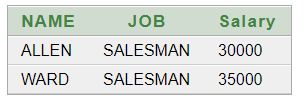
**Slice:**

select \*from employee\_details where salary<40000;



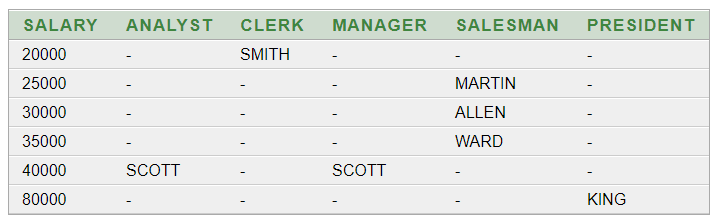
**Dice:**

select name, job , sum(salary) as "salary" from Employee\_details where (salary > 20000 and job = 'SALESMAN') group by name , job order by name asc;



**Pivot:**

select salary, max(decode(salary,'40000',name))ANALYST, max(decode(salary,'20000',name))CLERK,max(decode(salary,'40000',name))MANAGER ,max(decode(salary,'25000',name,'30000',name,'35000',name))Salesman, max(decode(salary,'80000',name))President from (select job,salary,name from Employee\_details) group by salary order by salary;



**Result:**

The given OLAP operations were successfully executed in Oracle.

**Experiment-4**

**Aim:**

To perform Preprocessing in data mining.

**Tools:** Weka mining tool.

**Procedure:**

**Apply discretization:**

* load the *sick* dataset and look at the attributes
* classify using NB, evaluating with cross-validation
* apply the supervised discretization filter and look at the effect (in the Preprocess panel)
* apply unsupervised discretization with different numbers of bins and look at the effect
* use the FilteredClassifier with NB and supervised discretization, evaluating with cross-validation
* repeat using unsupervised discretization with different numbers of bins
* compare and interpret the results.

**Apply feature selection using CfsSubsetEval**

* load the *mushroom* dataset and apply J48, IBk and NB, evaluating with crossvalidation
* select attributes using CfsSubsetEval and GreedyStepwise search
* interpret the results
* use AttributeSelectedClassifier (with CfsSubsetEval and GreedyStepwise
* search) for classifiers J48, IBk and NB, evaluating with cross-validation
* interpret the results.

**Apply feature selection using WrapperSubsetEval:**

* load the *vote* dataset and apply J48, IBk and NB, evaluating with crossvalidation
* select attributes using WrapperSubsetEval with InfoGainAttributeEval and

RankSearch, with the J48 classifier

* interpret the results
* use AttributeSelectedClassifier (with WrapperSubsetEval, InfoGainAttributeEval and RankSearch) with classifiers J48, IBk and NB, evaluating with cross-validation
* interpret the results.

**Sampling a dataset:**

* load the *letter* dataset and examine a particular (numeric) attribute
* apply the Resample filter to select half the dataset
* examine the same attribute and comment on the results.

**Output:**

**Apply discretization:**

=== Run information ===

Scheme: weka.classifiers.bayes.NaiveBayes

Relation: weka.datagenerators.classifiers.classification.RDG1-S\_1\_-n\_100\_-a\_10\_-c\_2\_-N\_0\_-I\_0\_-M\_1\_-R\_10

Instances: 100

Attributes: 11

a0

a1

a2

a3

a4

a5

a6

a7

a8

a9

class

Test mode: 10-fold cross-validation

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

Naive Bayes Classifier

Class

Attribute c0 c1

(0.66) (0.34)

=============================

a0

false 31.0 17.0

true 37.0 19.0

[total] 68.0 36.0

a1

false 33.0 20.0

true 35.0 16.0

[total] 68.0 36.0

a2

false 42.0 18.0

true 26.0 18.0

[total] 68.0 36.0

a3

false 32.0 9.0

true 36.0 27.0

[total] 68.0 36.0

a4

false 36.0 20.0

true 32.0 16.0

[total] 68.0 36.0

a5

false 43.0 3.0

true 25.0 33.0

[total] 68.0 36.0

a6

false 36.0 21.0

true 32.0 15.0

[total] 68.0 36.0

a7

false 36.0 16.0

true 32.0 20.0

[total] 68.0 36.0

a8

false 31.0 23.0

true 37.0 13.0

[total] 68.0 36.0

a9

false 35.0 16.0

true 33.0 20.0

[total] 68.0 36.0

Time taken to build model: 0 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 77 77 %

Incorrectly Classified Instances 23 23 %

Kappa statistic 0.4838

Mean absolute error 0.3108

Root mean squared error 0.3921

Relative absolute error 68.9937 %

Root relative squared error 82.6609 %

Total Number of Instances 100

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.833 0.353 0.821 0.833 0.827 0.484 0.820 0.870 c0

0.647 0.167 0.667 0.647 0.657 0.484 0.820 0.660 c1

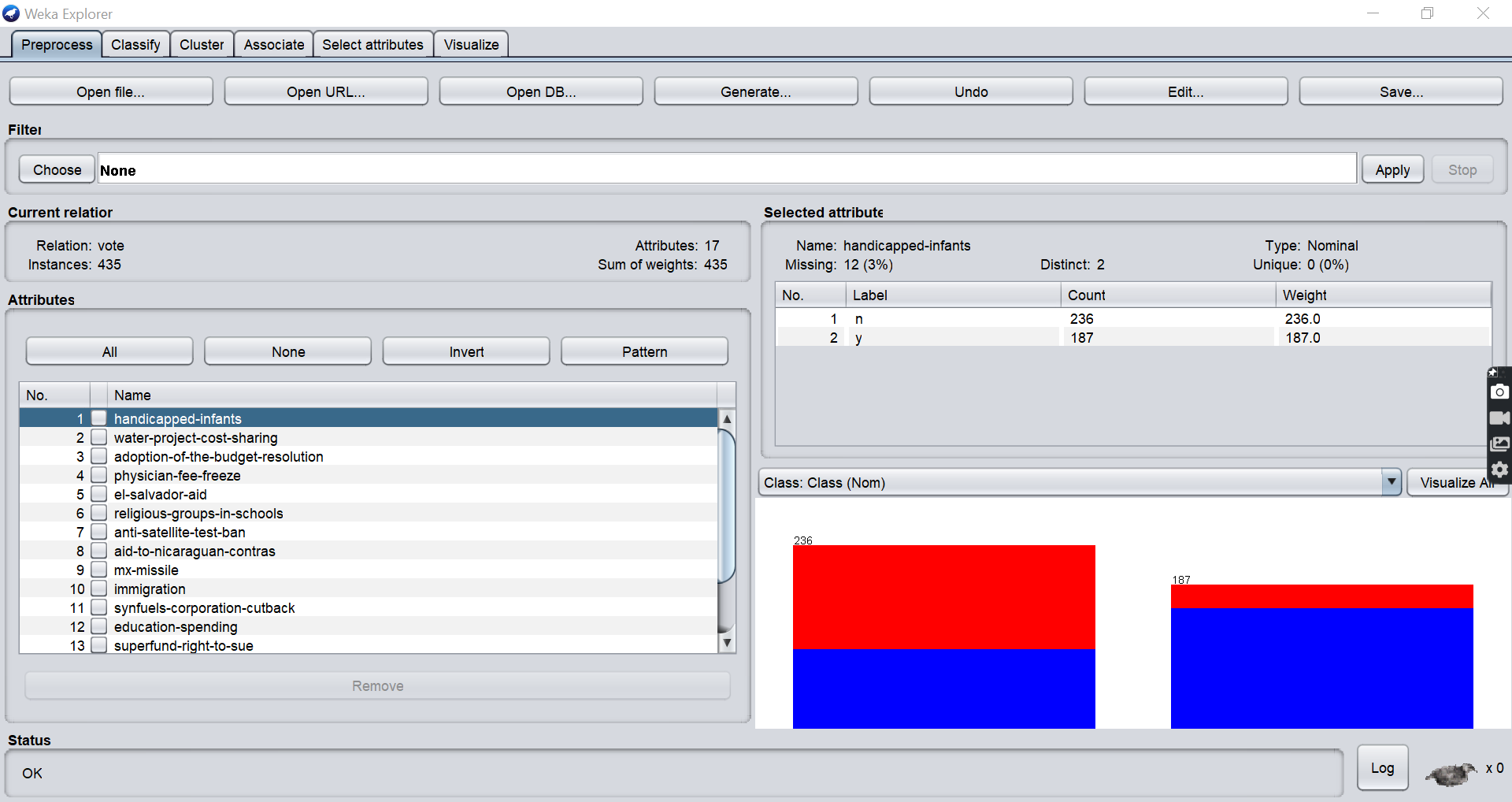
Weighted Avg. 0.770 0.290 0.768 0.770 0.769 0.484 0.820 0.799

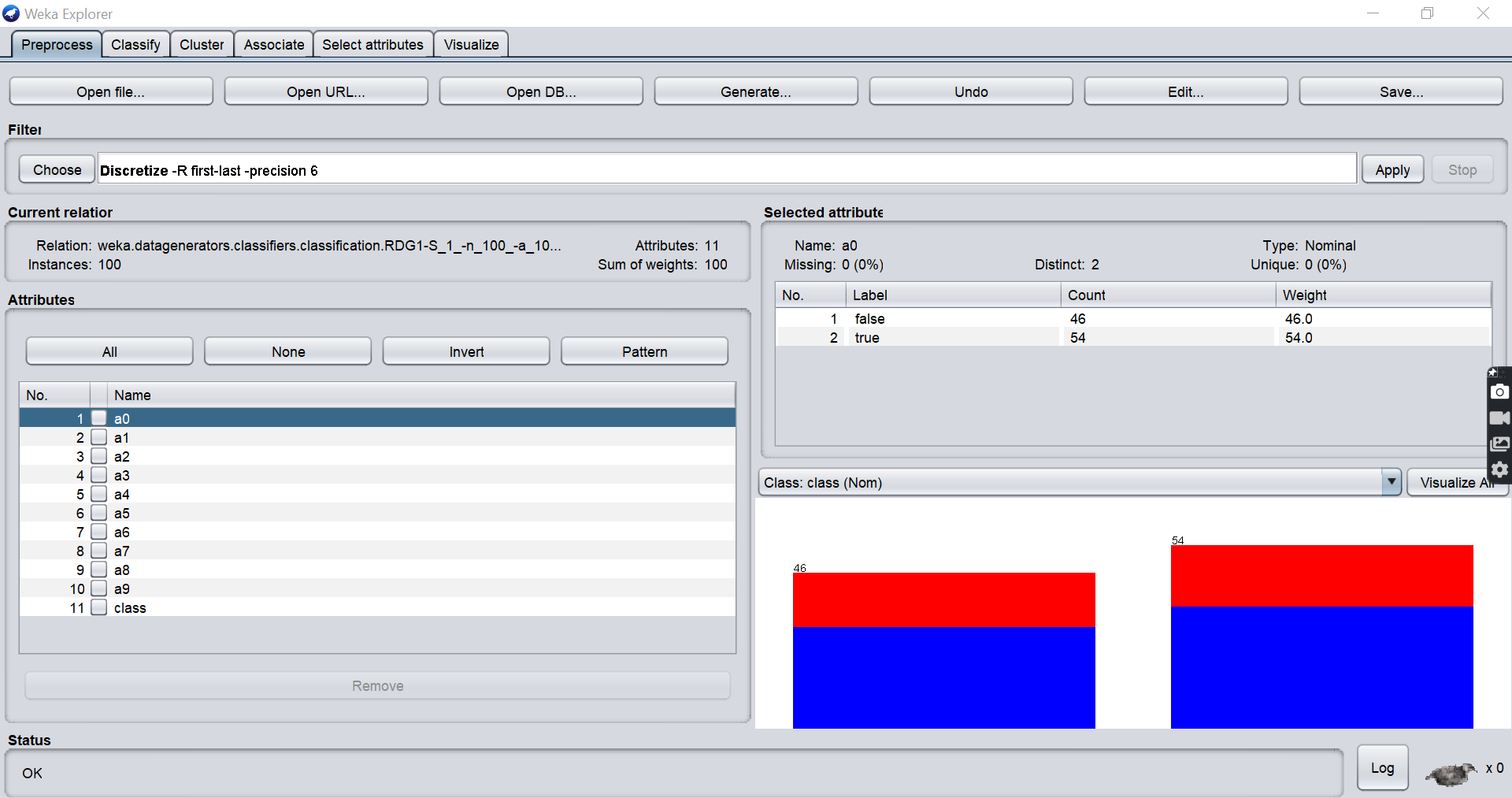
=== Confusion Matrix ===

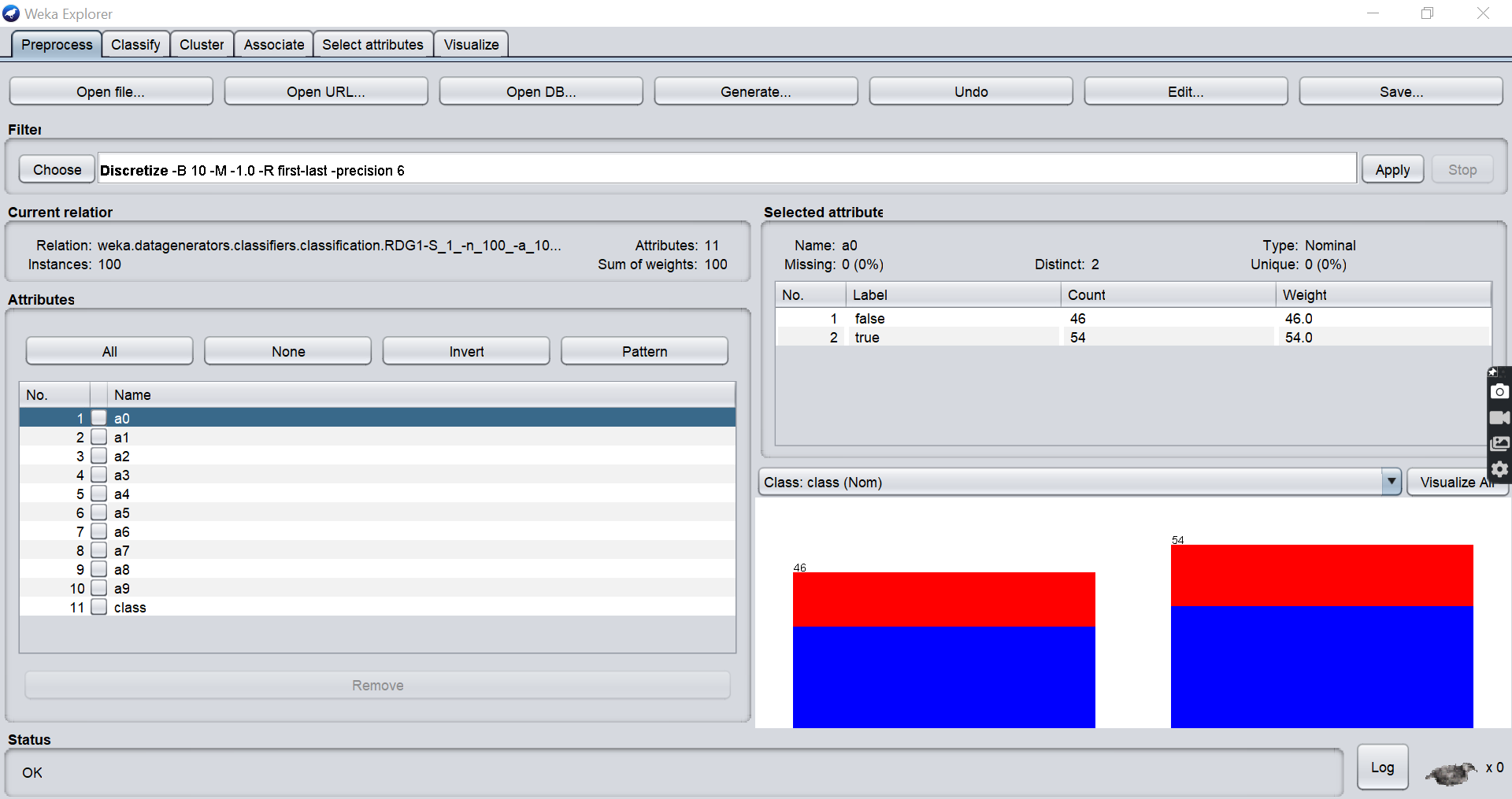
a b <-- classified as

55 11 | a = c0

12 22 | b = c1







**Apply feature selection using CfsSubsetEval**

**J48:**

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: vote

Instances: 435

Attributes: 17

handicapped-infants

water-project-cost-sharing

adoption-of-the-budget-resolution

physician-fee-freeze

el-salvador-aid

religious-groups-in-schools

anti-satellite-test-ban

aid-to-nicaraguan-contras

mx-missile

immigration

synfuels-corporation-cutback

education-spending

superfund-right-to-sue

crime

duty-free-exports

export-administration-act-south-africa

Class

Test mode: 10-fold cross-validation

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

J48 pruned tree

------------------

physician-fee-freeze = n: democrat (253.41/3.75)

physician-fee-freeze = y

| synfuels-corporation-cutback = n: republican (145.71/4.0)

| synfuels-corporation-cutback = y

| | mx-missile = n

| | | adoption-of-the-budget-resolution = n: republican (22.61/3.32)

| | | adoption-of-the-budget-resolution = y

| | | | anti-satellite-test-ban = n: democrat (5.04/0.02)

| | | | anti-satellite-test-ban = y: republican (2.21)

| | mx-missile = y: democrat (6.03/1.03)

Number of Leaves : 6

Size of the tree : 11

Time taken to build model: 0.01 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 419 96.3218 %

Incorrectly Classified Instances 16 3.6782 %

Kappa statistic 0.9224

Mean absolute error 0.0611

Root mean squared error 0.1748

Relative absolute error 12.887 %

Root relative squared error 35.9085 %

Total Number of Instances 435

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.970 0.048 0.970 0.970 0.970 0.922 0.971 0.965 democrat

0.952 0.030 0.952 0.952 0.952 0.922 0.971 0.947 republican

Weighted Avg. 0.963 0.041 0.963 0.963 0.963 0.922 0.971 0.958

=== Confusion Matrix ===

a b <-- classified as

259 8 | a = democrat

8 160 | b = republican

**IBK:**

=== Run information ===

Scheme: weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Relation: vote

Instances: 435

Attributes: 17

handicapped-infants

water-project-cost-sharing

adoption-of-the-budget-resolution

physician-fee-freeze

el-salvador-aid

religious-groups-in-schools

anti-satellite-test-ban

aid-to-nicaraguan-contras

mx-missile

immigration

synfuels-corporation-cutback

education-spending

superfund-right-to-sue

crime

duty-free-exports

export-administration-act-south-africa

Class

Test mode: 10-fold cross-validation

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

IB1 instance-based classifier

using 1 nearest neighbour(s) for classification

Time taken to build model: 0 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 402 92.4138 %

Incorrectly Classified Instances 33 7.5862 %

Kappa statistic 0.8423

Mean absolute error 0.073

Root mean squared error 0.242

Relative absolute error 15.3852 %

Root relative squared error 49.708 %

Total Number of Instances 435

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.914 0.060 0.961 0.914 0.937 0.844 0.965 0.973 democrat

0.940 0.086 0.873 0.940 0.905 0.844 0.965 0.934 republican

Weighted Avg. 0.924 0.070 0.927 0.924 0.925 0.844 0.965 0.958

=== Confusion Matrix ===

a b <-- classified as

244 23 | a = democrat

10 158 | b = republican

**NB:**

=== Run information ===

Scheme: weka.classifiers.bayes.NaiveBayes

Relation: vote

Instances: 435

Attributes: 17

handicapped-infants

water-project-cost-sharing

adoption-of-the-budget-resolution

physician-fee-freeze

el-salvador-aid

religious-groups-in-schools

anti-satellite-test-ban

aid-to-nicaraguan-contras

mx-missile

immigration

synfuels-corporation-cutback

education-spending

superfund-right-to-sue

crime

duty-free-exports

export-administration-act-south-africa

Class

Test mode: 10-fold cross-validation

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

Naive Bayes Classifier

Class

Attribute democrat republican

(0.61) (0.39)

===============================================================

handicapped-infants

n 103.0 135.0

y 157.0 32.0

[total] 260.0 167.0

water-project-cost-sharing

n 120.0 74.0

y 121.0 76.0

[total] 241.0 150.0

adoption-of-the-budget-resolution

n 30.0 143.0

y 232.0 23.0

[total] 262.0 166.0

physician-fee-freeze

n 246.0 3.0

y 15.0 164.0

[total] 261.0 167.0

el-salvador-aid

n 201.0 9.0

y 56.0 158.0

[total] 257.0 167.0

religious-groups-in-schools

n 136.0 18.0

y 124.0 150.0

[total] 260.0 168.0

anti-satellite-test-ban

n 60.0 124.0

y 201.0 40.0

[total] 261.0 164.0

aid-to-nicaraguan-contras

n 46.0 134.0

y 219.0 25.0

[total] 265.0 159.0

mx-missile

n 61.0 147.0

y 189.0 20.0

[total] 250.0 167.0

immigration

n 140.0 74.0

y 125.0 93.0

[total] 265.0 167.0

synfuels-corporation-cutback

n 127.0 139.0

y 130.0 22.0

[total] 257.0 161.0

education-spending

n 214.0 21.0

y 37.0 136.0

[total] 251.0 157.0

superfund-right-to-sue

n 180.0 23.0

y 74.0 137.0

[total] 254.0 160.0

crime

n 168.0 4.0

y 91.0 159.0

[total] 259.0 163.0

duty-free-exports

n 92.0 143.0

y 161.0 15.0

[total] 253.0 158.0

export-administration-act-south-africa

n 13.0 51.0

y 174.0 97.0

[total] 187.0 148.0

Time taken to build model: 0 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 392 90.1149 %

Incorrectly Classified Instances 43 9.8851 %

Kappa statistic 0.7949

Mean absolute error 0.0995

Root mean squared error 0.2977

Relative absolute error 20.9815 %

Root relative squared error 61.1406 %

Total Number of Instances 435

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.891 0.083 0.944 0.891 0.917 0.797 0.973 0.984 democrat

0.917 0.109 0.842 0.917 0.877 0.797 0.973 0.957 republican

Weighted Avg. 0.901 0.093 0.905 0.901 0.902 0.797 0.973 0.973

=== Confusion Matrix ===

a b <-- classified as

238 29 | a = democrat

14 154 | b = republican

**Apply CfsSubsetEval and greedyStepwise:**

=== Run information ===

Evaluator: weka.attributeSelection.CfsSubsetEval -P 1 -E 1

Search: weka.attributeSelection.GreedyStepwise -T -1.7976931348623157E308 -N -1 -num-slots 1

Relation: vote

Instances: 435

Attributes: 17

handicapped-infants

water-project-cost-sharing

adoption-of-the-budget-resolution

physician-fee-freeze

el-salvador-aid

religious-groups-in-schools

anti-satellite-test-ban

aid-to-nicaraguan-contras

mx-missile

immigration

synfuels-corporation-cutback

education-spending

superfund-right-to-sue

crime

duty-free-exports

export-administration-act-south-africa

Class

Evaluation mode: 10-fold cross-validation

**Apply feature selection using WrapperSubsetEval:**

**J48:**

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: vote

Instances: 435

Attributes: 17

handicapped-infants

water-project-cost-sharing

adoption-of-the-budget-resolution

physician-fee-freeze

el-salvador-aid

religious-groups-in-schools

anti-satellite-test-ban

aid-to-nicaraguan-contras

mx-missile

immigration

synfuels-corporation-cutback

education-spending

superfund-right-to-sue

crime

duty-free-exports

export-administration-act-south-africa

Class

Test mode: 10-fold cross-validation

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

J48 pruned tree

------------------

physician-fee-freeze = n: democrat (253.41/3.75)

physician-fee-freeze = y

| synfuels-corporation-cutback = n: republican (145.71/4.0)

| synfuels-corporation-cutback = y

| | mx-missile = n

| | | adoption-of-the-budget-resolution = n: republican (22.61/3.32)

| | | adoption-of-the-budget-resolution = y

| | | | anti-satellite-test-ban = n: democrat (5.04/0.02)

| | | | anti-satellite-test-ban = y: republican (2.21)

| | mx-missile = y: democrat (6.03/1.03)

Number of Leaves : 6

Size of the tree : 11

Time taken to build model: 0 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 419 96.3218 %

Incorrectly Classified Instances 16 3.6782 %

Kappa statistic 0.9224

Mean absolute error 0.0611

Root mean squared error 0.1748

Relative absolute error 12.887 %

Root relative squared error 35.9085 %

Total Number of Instances 435

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.970 0.048 0.970 0.970 0.970 0.922 0.971 0.965 democrat

0.952 0.030 0.952 0.952 0.952 0.922 0.971 0.947 republican

Weighted Avg. 0.963 0.041 0.963 0.963 0.963 0.922 0.971 0.958

=== Confusion Matrix ===

a b <-- classified as

259 8 | a = democrat

8 160 | b = republican

**IBK:**

=== Run information ===

Scheme: weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Relation: vote

Instances: 435

Attributes: 17

handicapped-infants

water-project-cost-sharing

adoption-of-the-budget-resolution

physician-fee-freeze

el-salvador-aid

religious-groups-in-schools

anti-satellite-test-ban

aid-to-nicaraguan-contras

mx-missile

immigration

synfuels-corporation-cutback

education-spending

superfund-right-to-sue

crime

duty-free-exports

export-administration-act-south-africa

Class

Test mode: 10-fold cross-validation

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

IB1 instance-based classifier

using 1 nearest neighbour(s) for classification

Time taken to build model: 0 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 402 92.4138 %

Incorrectly Classified Instances 33 7.5862 %

Kappa statistic 0.8423

Mean absolute error 0.073

Root mean squared error 0.242

Relative absolute error 15.3852 %

Root relative squared error 49.708 %

Total Number of Instances 435

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.914 0.060 0.961 0.914 0.937 0.844 0.965 0.973 democrat

0.940 0.086 0.873 0.940 0.905 0.844 0.965 0.934 republican

Weighted Avg. 0.924 0.070 0.927 0.924 0.925 0.844 0.965 0.958

=== Confusion Matrix ===

a b <-- classified as

244 23 | a = democrat

10 158 | b = republican

**NB:**

=== Run information ===

Scheme: weka.classifiers.bayes.NaiveBayes

Relation: vote

Instances: 435

Attributes: 17

handicapped-infants

water-project-cost-sharing

adoption-of-the-budget-resolution

physician-fee-freeze

el-salvador-aid

religious-groups-in-schools

anti-satellite-test-ban

aid-to-nicaraguan-contras

mx-missile

immigration

synfuels-corporation-cutback

education-spending

superfund-right-to-sue

crime

duty-free-exports

export-administration-act-south-africa

Class

Test mode: 10-fold cross-validation

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

Naive Bayes Classifier

Class

Attribute democrat republican

(0.61) (0.39)

===============================================================

handicapped-infants

n 103.0 135.0

y 157.0 32.0

[total] 260.0 167.0

water-project-cost-sharing

n 120.0 74.0

y 121.0 76.0

[total] 241.0 150.0

adoption-of-the-budget-resolution

n 30.0 143.0

y 232.0 23.0

[total] 262.0 166.0

physician-fee-freeze

n 246.0 3.0

y 15.0 164.0

[total] 261.0 167.0

el-salvador-aid

n 201.0 9.0

y 56.0 158.0

[total] 257.0 167.0

religious-groups-in-schools

n 136.0 18.0

y 124.0 150.0

[total] 260.0 168.0

anti-satellite-test-ban

n 60.0 124.0

y 201.0 40.0

[total] 261.0 164.0

aid-to-nicaraguan-contras

n 46.0 134.0

y 219.0 25.0

[total] 265.0 159.0

mx-missile

n 61.0 147.0

y 189.0 20.0

[total] 250.0 167.0

immigration

n 140.0 74.0

y 125.0 93.0

[total] 265.0 167.0

synfuels-corporation-cutback

n 127.0 139.0

y 130.0 22.0

[total] 257.0 161.0

education-spending

n 214.0 21.0

y 37.0 136.0

[total] 251.0 157.0

superfund-right-to-sue

n 180.0 23.0

y 74.0 137.0

[total] 254.0 160.0

crime

n 168.0 4.0

y 91.0 159.0

[total] 259.0 163.0

duty-free-exports

n 92.0 143.0

y 161.0 15.0

[total] 253.0 158.0

export-administration-act-south-africa

n 13.0 51.0

y 174.0 97.0

[total] 187.0 148.0

Time taken to build model: 0 seconds

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 392 90.1149 %

Incorrectly Classified Instances 43 9.8851 %

Kappa statistic 0.7949

Mean absolute error 0.0995

Root mean squared error 0.2977

Relative absolute error 20.9815 %

Root relative squared error 61.1406 %

Total Number of Instances 435

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.891 0.083 0.944 0.891 0.917 0.797 0.973 0.984 democrat

0.917 0.109 0.842 0.917 0.877 0.797 0.973 0.957 republican

Weighted Avg. 0.901 0.093 0.905 0.901 0.902 0.797 0.973 0.973

=== Confusion Matrix ===

a b <-- classified as

238 29 | a = democrat

14 154 | b = republican

**InfoGainAttributeEval and Ranker Search:**

=== Run information ===

Evaluator: weka.attributeSelection.InfoGainAttributeEval

Search: weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1

Relation: vote

Instances: 435

Attributes: 17

handicapped-infants

water-project-cost-sharing

adoption-of-the-budget-resolution

physician-fee-freeze

el-salvador-aid

religious-groups-in-schools

anti-satellite-test-ban

aid-to-nicaraguan-contras

mx-missile

immigration

synfuels-corporation-cutback

education-spending

superfund-right-to-sue

crime

duty-free-exports

export-administration-act-south-africa

Class

Evaluation mode: 10-fold cross-validation

**AttributeSelectedClassifier (with WrappedSubsetEval ,InfoGainAttributeEval and RankSearch) with classifiers J48,NB and IBK:**

=== Run information ===

Evaluator: weka.attributeSelection.InfoGainAttributeEval

Search: weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1

Relation: weka.datagenerators.classifiers.classification.RDG1-S\_1\_-n\_100\_-a\_10\_-c\_2\_-N\_0\_-I\_0\_-M\_1\_-R\_10

Instances: 100

Attributes: 11

a0

a1

a2

a3

a4

a5

a6

a7

a8

a9

class

Evaluation mode: evaluate on all training data

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

Search Method:

Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 11 class):

Information Gain Ranking Filter

Ranked attributes:

0.255714 6 a5

0.038926 4 a3

0.024319 9 a8

0.009714 3 a2

0.005152 8 a7

0.003551 10 a9

0.003551 2 a1

0.002202 7 a6

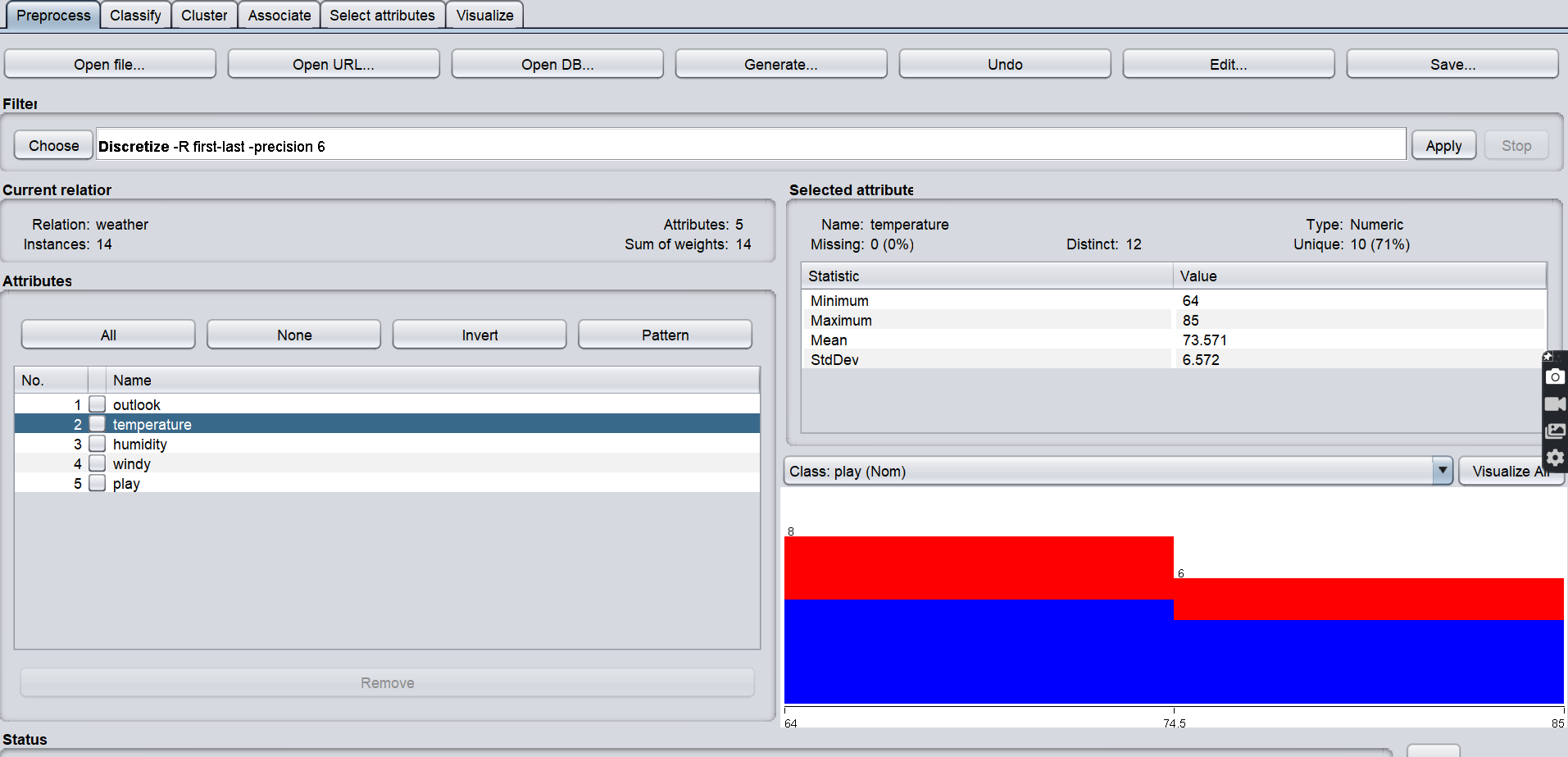
0.000531 5 a4

0.000168 1 a0

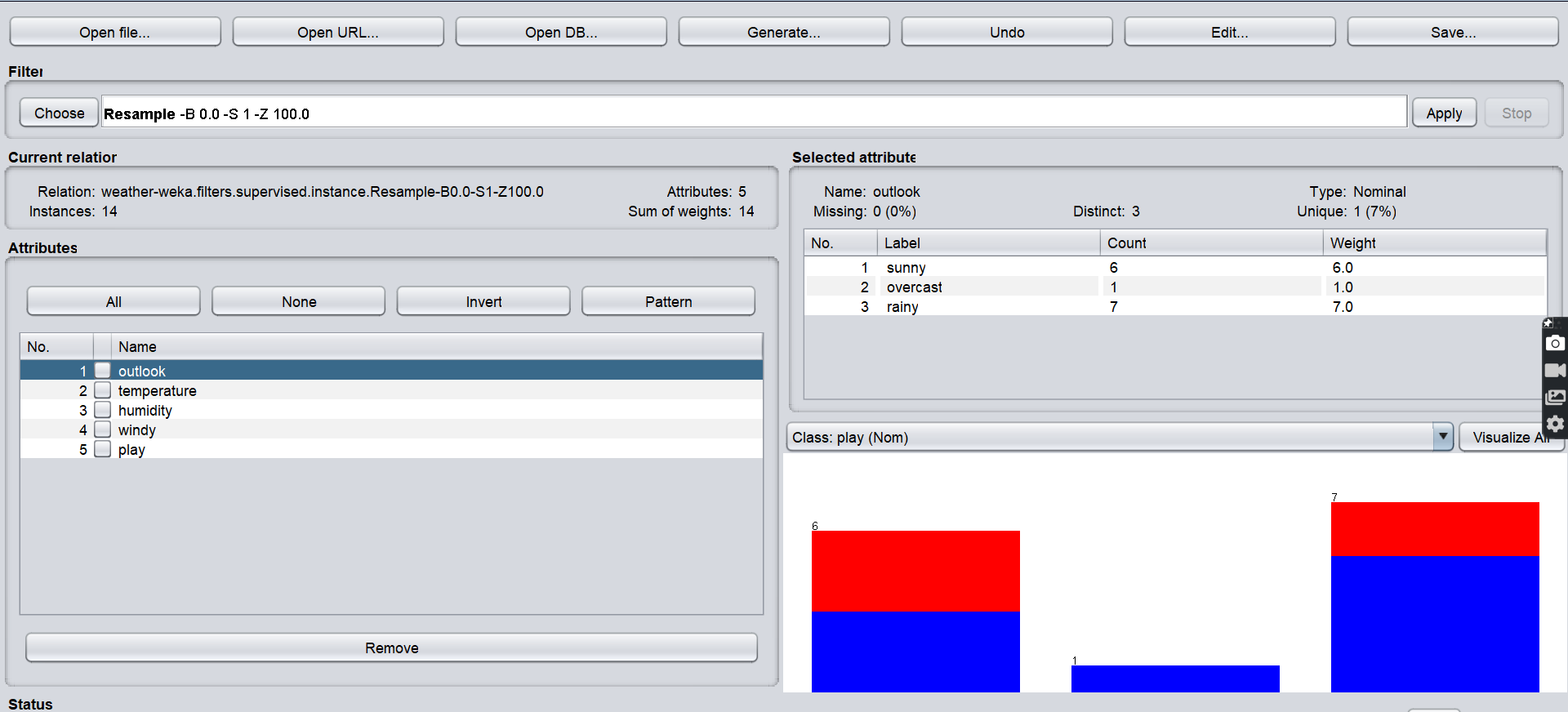
Selected attributes: 6,4,9,3,8,10,2,7,5,1 : 10

**Sampling Dataset:**

**Numeric Attribute:**



**Apply Resample Filter:**



**Result:**

The given data processing has been successfully modelled and executed.

**Experiment-5**

**Aim:**

To perform Clustering using Weka mining tool

**Tools:** Weka mining tool.

**Procedure:**

1. Go to the WEKA Explorer environment and load the training file “iris.arff”. Remove the class attribute using the preprocessing dialog. Go to the clustering dialog.

* Cluster the iris dataset using the k-Means Clustering algorithm with k=5. Hand in the result given by WEKA (Cluster mean and standard deviation).
* Cluster the iris dataset using the k-Means Clustering algorithm with k=3 and k=4 and with ten different values of the seed parameter, using the option: Classes to cluster evaluation and and store the results on an excel file. Compute the mean of the two different k values for the k-Means
* Visualize the cluster mean values and standard deviation for - sepallength versus sepalwidth – petallength versus petalwidth. Don’t erase the result of the k-Means algorithm.

2. Using the EM algorithm with the iris dataset, try to run the EM algorithm, with the automatic estimation of the number of clusters. Then, try to repeat the experiments of the previous section with the EM algorithm. Compare the results.

3. Create an “arff”-file containing the data points as given below.

t1 = (4, 2, 3, 5, 2, 2, 2, 1)

t2 = (3, 2, 5, 4, 3, 2, 1, 4)

t3 = (1, 3, 3, 5, 2, 3, 2, 1)

t4 = (4, 2, 0, 5, 2, 2, 2, 1)

t5 = (3, 2, 3, 4, 3, 2, 1, 4)

t6 = (2, 5, 3, 5, 2, 2, 2, 1)

t7 = (4, 1, 3, 7, 2, 1, 2, 1)

t8 = (3, 1, 5, 4, 3, 2, 1, 4)

t9 = (2, 5, 2, 5, 2, 5, 2, 1)

Cluster the above data file using K-means and EM with k=2 and k=3 clusters.

4. Using Cobweb algorithm, run the algorithm on the iris dataset. First you must use the option: Using training set and then using the option: **Classes to cluster evaluation**. For both the options visualize the tree obtained. What is the difference? How is the behaviour of the algorithm? In your opinion, what is the factor that influences the algorithm?

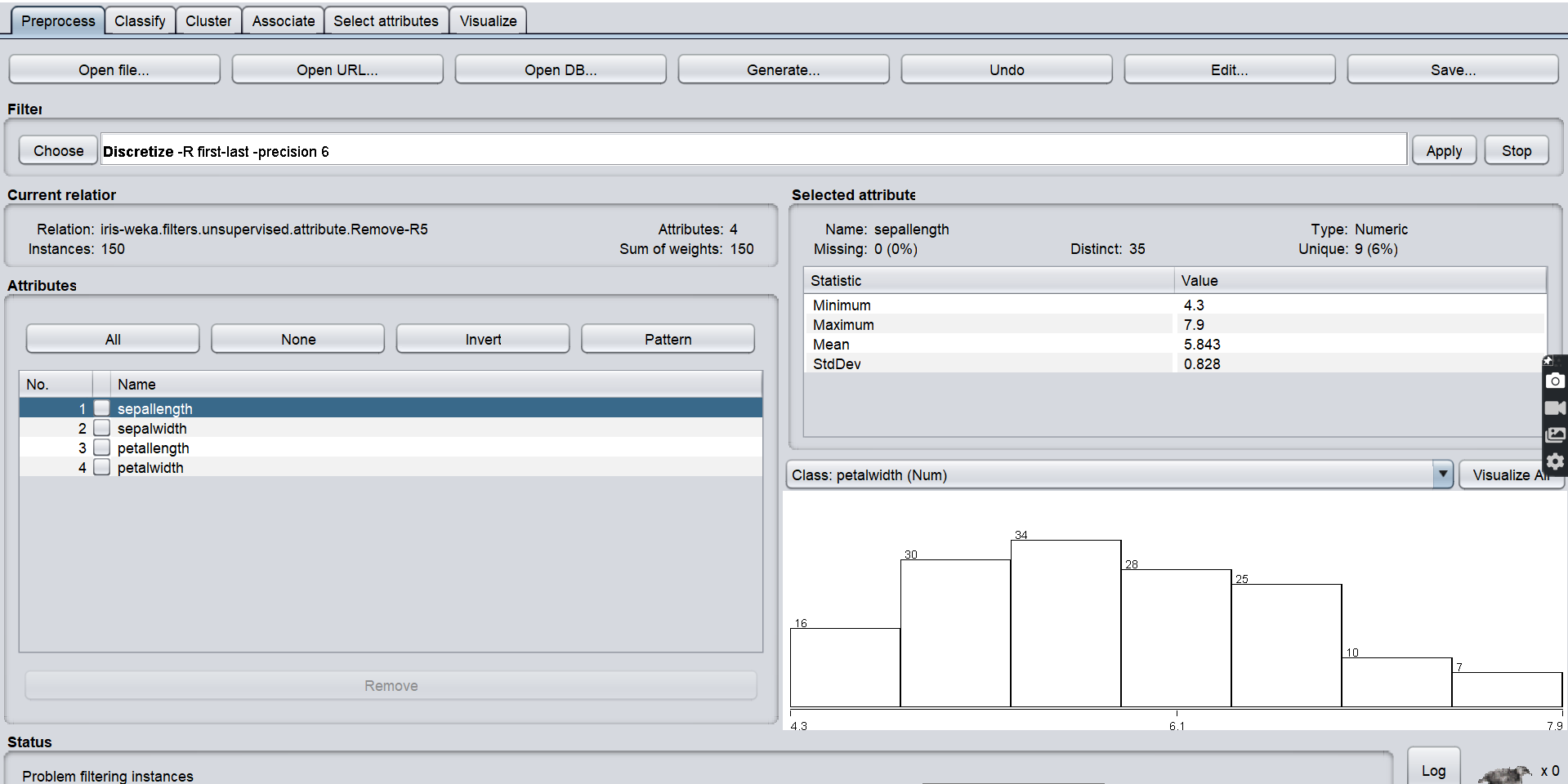
5. Download the dataset http://www.upo.es/eps/bigs/dataSet/Lymphoma96x4026+9classes.arff.

k-means and EM (with Classes to cluster evaluation option) change the number of cluster from

10 to 20, step equal to 1, and store the Incorrectly clustered instances for both the algorithms.

**Output:**

**Load the training file “iris.arff and Remove the class attribute :**

****

**Cluster the iris dataset using the k-Means Clustering algorithm with k=5**

=== Run information ===

Scheme: weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 5 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10

Relation: iris-weka.filters.unsupervised.attribute.Remove-R5

Instances: 150

Attributes: 4

sepallength

sepalwidth

petallength

petalwidth

Test mode: evaluate on training data

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

kMeans

======

Number of iterations: 9

Within cluster sum of squared errors: 5.130784647061167

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4

Cluster 1: 6.2,2.9,4.3,1.3

Cluster 2: 6.9,3.1,5.1,2.3

Cluster 3: 5.5,4.2,1.4,0.2

Cluster 4: 6.9,3.1,4.9,1.5

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1 2 3 4

(150.0) (27.0) (26.0) (27.0) (50.0) (20.0)

===============================================================================

sepallength 5.8433 6.0296 5.55 6.9667 5.006 6.55

sepalwidth 3.054 2.7556 2.5808 3.137 3.418 3.05

petallength 3.7587 4.9444 3.9269 5.8852 1.464 4.805

petalwidth 1.1987 1.7037 1.2 2.2 0.244 1.55

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

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

Clustered Instances

0 27 ( 18%)

1 26 ( 17%)

2 27 ( 18%)

3 50 ( 33%)

4 20 ( 13%)

**Cluster the iris dataset using the k-Means Clustering algorithm with k=3 and k=4**

**K=3**

=== Run information ===

Scheme: weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10

Relation: iris-weka.filters.unsupervised.attribute.Remove-R5

Instances: 150

Attributes: 4

sepallength

sepalwidth

petallength

petalwidth

Test mode: evaluate on training data

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

kMeans

======

Number of iterations: 6

Within cluster sum of squared errors: 6.998114004826762

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4

Cluster 1: 6.2,2.9,4.3,1.3

Cluster 2: 6.9,3.1,5.1,2.3

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1 2

(150.0) (61.0) (50.0) (39.0)

=========================================================

sepallength 5.8433 5.8885 5.006 6.8462

sepalwidth 3.054 2.7377 3.418 3.0821

petallength 3.7587 4.3967 1.464 5.7026

petalwidth 1.1987 1.418 0.244 2.0795

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

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

Clustered Instances

0 61 ( 41%)

1 50 ( 33%)

2 39 ( 26%)

**K=4**

=== Run information ===

Scheme: weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 4 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10

Relation: iris-weka.filters.unsupervised.attribute.Remove-R5

Instances: 150

Attributes: 4

sepallength

sepalwidth

petallength

petalwidth

Test mode: evaluate on training data

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

kMeans

======

Number of iterations: 4

Within cluster sum of squared errors: 5.532831003081898

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4

Cluster 1: 6.2,2.9,4.3,1.3

Cluster 2: 6.9,3.1,5.1,2.3

Cluster 3: 5.5,4.2,1.4,0.2

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1 2 3

(150.0) (42.0) (29.0) (29.0) (50.0)

====================================================================

sepallength 5.8433 6.25 5.5828 6.9586 5.006

sepalwidth 3.054 2.9 2.569 3.1345 3.418

petallength 3.7587 4.8738 4.0034 5.8552 1.464

petalwidth 1.1987 1.6405 1.231 2.1724 0.244

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

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

Clustered Instances

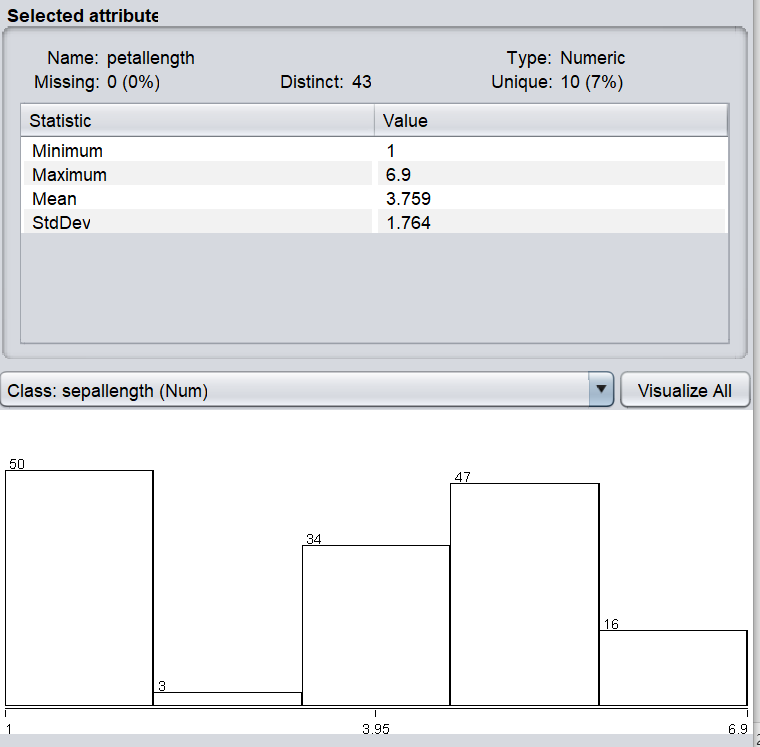
0 42 ( 28%)

1 29 ( 19%)

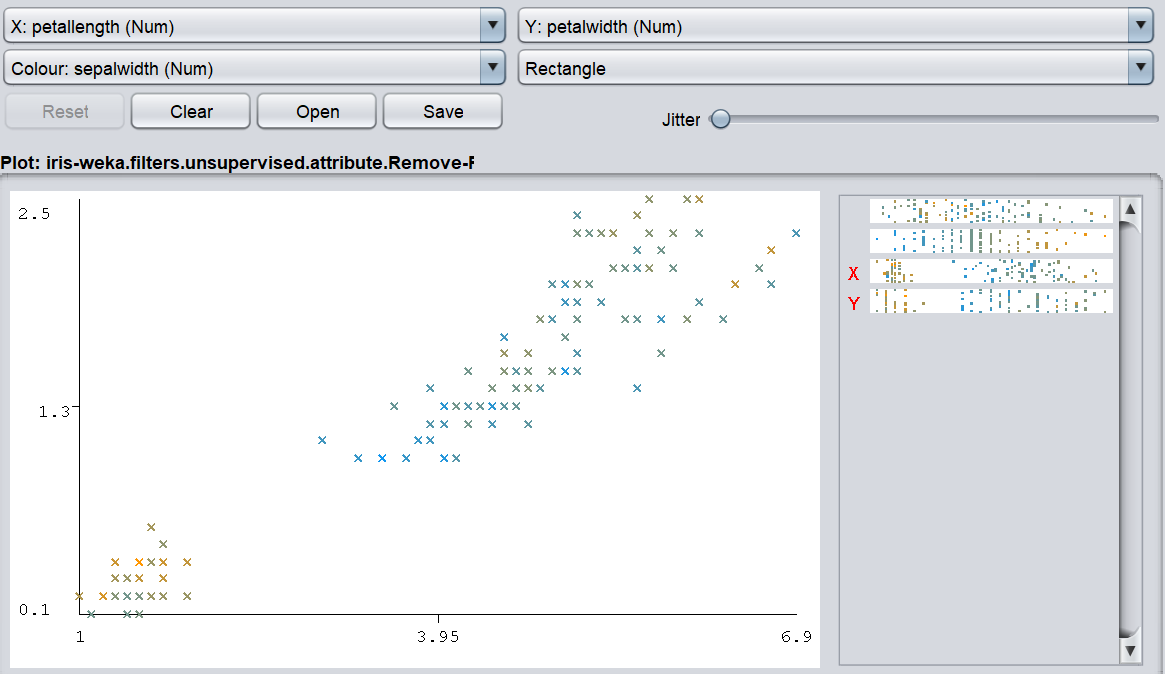
2 29 ( 19%)

3 50 ( 33%)

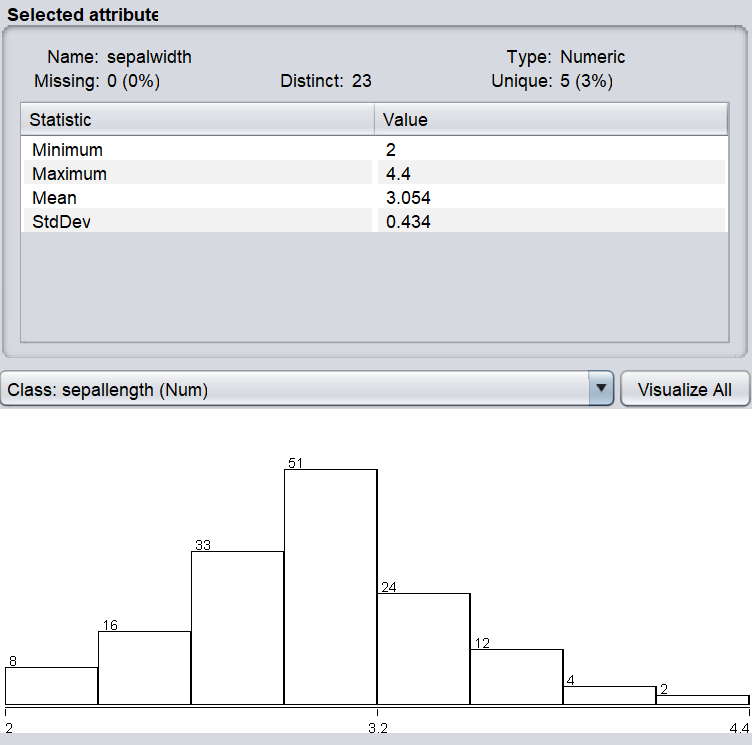
**Visualization of petallength versus petalwidth.**

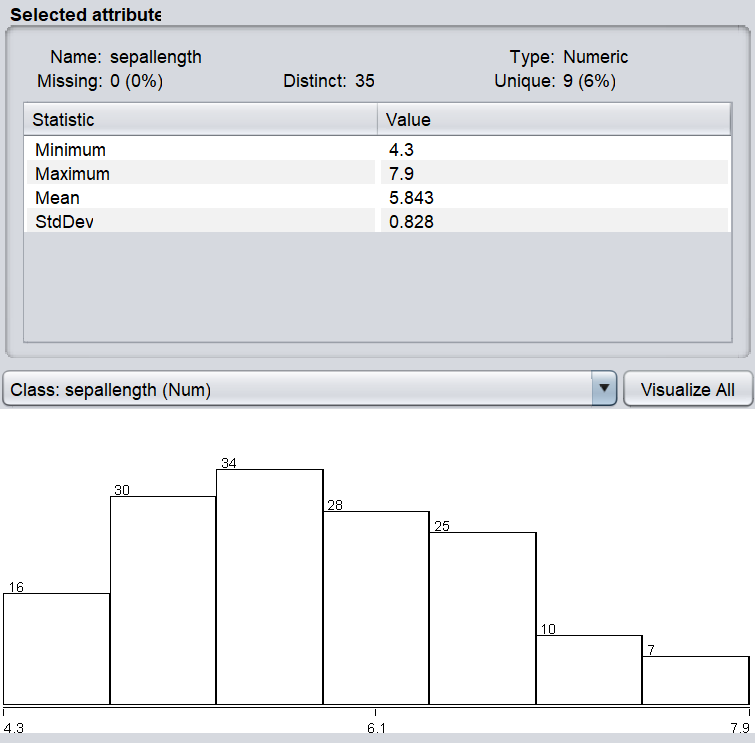
****

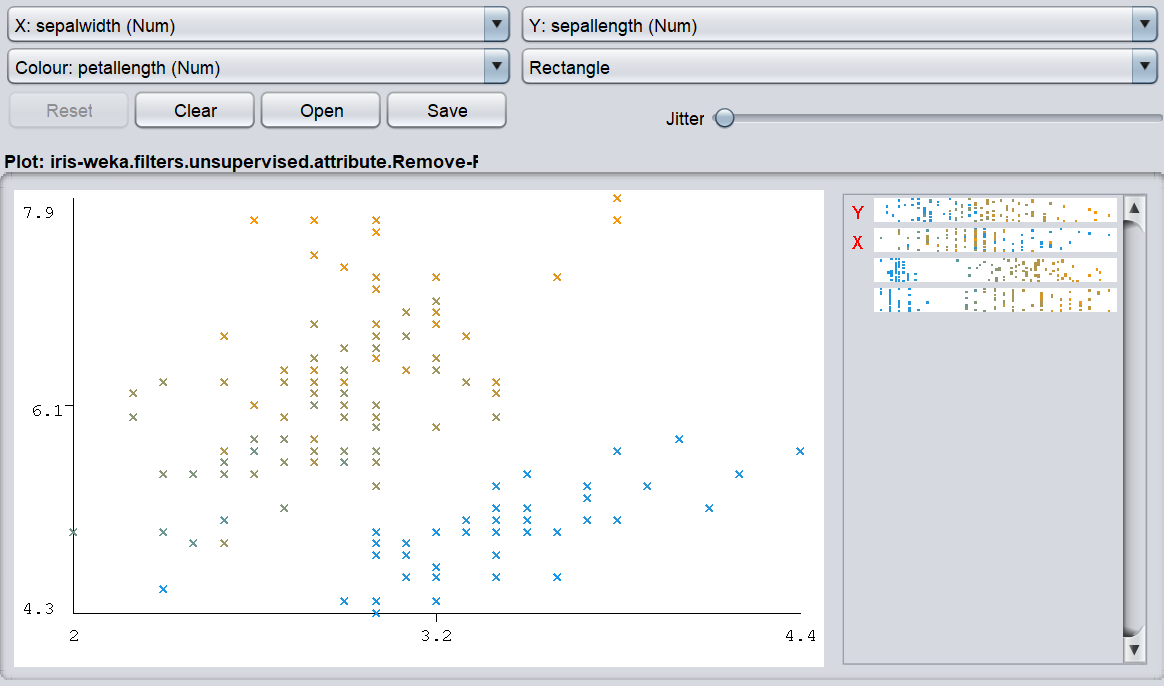
****

****

**Visualization of sepallength versus sepalwidth**

****

****

****

**Using the EM algorithm with the iris dataset, try to run the EM algorithm, with the automatic estimation of the number of clusters.**

=== Run information ===

Scheme: weka.clusterers.EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

class

Test mode: evaluate on training data

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

EM

==

Number of clusters selected by cross validation: 4

Number of iterations performed: 16

Cluster

Attribute 0 1 2 3

(0.32) (0.33) (0.2) (0.14)

====================================================

sepallength

mean 5.897 5.006 6.9426 6.1304

std. dev. 0.5279 0.3489 0.498 0.2943

sepalwidth

mean 2.7519 3.418 3.1103 2.8088

std. dev. 0.3103 0.3772 0.2952 0.2361

petallength

mean 4.2267 1.464 5.8559 5.0993

std. dev. 0.445 0.1718 0.4626 0.2462

petalwidth

mean 1.3134 0.244 2.1495 1.8254

std. dev. 0.1864 0.1061 0.232 0.2152

class

Iris-setosa 1 51 1 1

Iris-versicolor 48.1125 1 1.0182 3.8693

Iris-virginica 2.0983 1 31.0375 19.8641

[total] 51.2108 53 33.0557 24.7335

Time taken to build model (full training data) : 0.61 seconds

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

Clustered Instances

0 48 ( 32%)

1 50 ( 33%)

2 29 ( 19%)

3 23 ( 15%)

Log likelihood: -2.03504

**NUM OF CLUSTERS=10**

=== Run information ===

Scheme: weka.clusterers.EM -I 100 -N 10 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

class

Test mode: evaluate on training data

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

EM

==

Number of clusters: 10

Number of iterations performed: 0

Cluster

Attribute 0 1 2 3 4 5 6 7 8 9

(0.01) (0.13) (0.13) (0.11) (0.11) (0.08) (0.09) (0.13) (0.13) (0.08)

====================================================================================================

sepallength

mean 4.5 6.5474 6.41 4.7118 5.6941 5.3917 5.5231 6.0684 5.05 7.475

std. dev. 0 0.2855 0.3323 0.2261 0.2249 0.2466 0.4512 0.4042 0.2351 0.2701

sepalwidth

mean 2.3 3.1158 2.995 3.0824 2.8235 3.925 2.3538 2.7368 3.455 3.125

std. dev. 0 0.1893 0.2139 0.0951 0.1437 0.2137 0.1613 0.2338 0.0999 0.398

petallength

mean 1.3 5.5 4.65 1.4118 4.1824 1.525 3.7615 5.1316 1.48 6.3

std. dev. 0 0.2769 0.2236 0.1409 0.2744 0.1865 0.4369 0.3351 0.1852 0.3568

petalwidth

mean 0.3 2.2526 1.48 0.1765 1.2941 0.275 1.1308 1.7842 0.28 2.05

std. dev. 0 0.1541 0.1322 0.0562 0.1298 0.1055 0.1653 0.1608 0.1196 0.2541

class

Iris-setosa 2 1 1 18 1 13 1 1 21 1

Iris-versicolor 1 1 21 1 18 1 14 1 1 1

Iris-virginica 1 20 1 1 1 1 1 20 1 13

[total] 4 22 23 20 20 15 16 22 23 15

Time taken to build model (full training data) : 0.01 seconds

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

Clustered Instances

0 1 ( 1%)

1 20 ( 13%)

2 20 ( 13%)

3 17 ( 11%)

4 17 ( 11%)

5 13 ( 9%)

6 13 ( 9%)

7 19 ( 13%)

8 19 ( 13%)

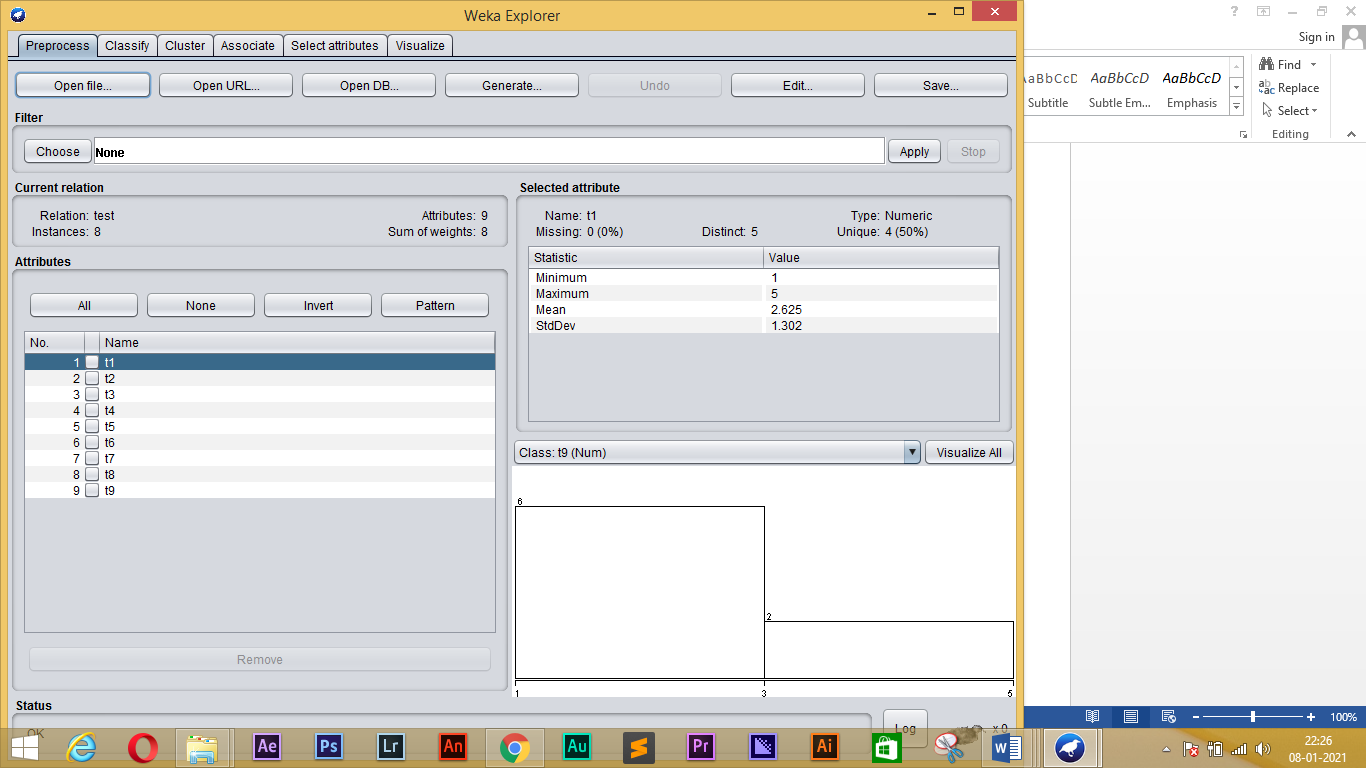
9 11 ( 7%)

Log likelihood: -1.03048

**Create an “arff”-file containing the data points :**

**Kmeans K =2 and K =3**

**Cluster the above data file using K-means and EM with k=2 and k=3 clusters.**



**K-MEANS:**

**K=2:**

=== Run information ===

Scheme: weka.clusterers.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

Relation: test

Instances: 8

Attributes: 9

t1

t2

t3

t4

t5

t6

t7

t8

t9

Test mode: evaluate on training data

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

kMeans

======

Number of iterations: 2

Within cluster sum of squared errors: 5.392129629629629

Initial starting points (random):

Cluster 0: 2,2,3,2,2,2,1,1,5

Cluster 1: 2,2,3,2,2,5,1,1,5

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1

(8.0) (6.0) (2.0)

============================================

t1 2.625 2.3333 3.5

t2 3 3 3

t3 2.5 2 4

t4 2.25 1.8333 3.5

t5 2.75 2.6667 3

t6 2.75 2 5

t7 2.625 2.1667 4

t8 2.875 3 2.5

t9 3 2.3333 5

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

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

Clustered Instances

0 6 ( 75%)

1 2 ( 25%)

**K=3:**

=== Run information ===

Scheme: weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 3 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10

Relation: test

Instances: 8

Attributes: 9

t1

t2

t3

t4

t5

t6

t7

t8

t9

Test mode: evaluate on training data

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

kMeans

======

Number of iterations: 3

Within cluster sum of squared errors: 2.861027777777778

Initial starting points (random):

Cluster 0: 2,2,3,2,2,2,1,1,5

Cluster 1: 2,2,3,2,2,5,1,1,5

Cluster 2: 5,4,5,5,4,5,7,4,5

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1 2

(8.0) (5.0) (2.0) (1.0)

=======================================================

t1 2.625 2.4 2 5

t2 3 3.2 2 4

t3 2.5 1.8 3 5

t4 2.25 1.8 2 5

t5 2.75 2.8 2 4

t6 2.75 2 3.5 5

t7 2.625 2.4 1 7

t8 2.875 3.4 1 4

t9 3 1.8 5 5

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

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

Clustered Instances

0 5 ( 63%)

1 2 ( 25%)

2 1 ( 13%)

**EM:**

**K=2:**

=== Run information ===

Scheme: weka.clusterers.EM -I 100 -N 2 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Relation: test

Instances: 8

Attributes: 9

t1

t2

t3

t4

t5

t6

t7

t8

t9

Test mode: evaluate on training data

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

EM

==

Number of clusters: 2

Number of iterations performed: 0

Cluster

Attribute 0 1

(0.13) (0.88)

==========================

t1

mean 5 2.2857

std. dev. 0 0.9512

t2

mean 4 2.8571

std. dev. 0 1.3452

t3

mean 5 2.1429

std. dev. 0 0.8997

t4

mean 5 1.8571

std. dev. 0 1.215

t5

mean 4 2.5714

std. dev. 0 0.9759

t6

mean 5 2.4286

std. dev. 0 1.2724

t7

mean 7 2

std. dev. 0 1.1547

t8

mean 4 2.7143

std. dev. 0 1.496

t9

mean 5 2.7143

std. dev. 0 1.6036

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

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

Clustered Instances

0 1 ( 13%)

1 7 ( 88%)

Log likelihood: 2.14563

**K=3:**

=== Run information ===

Scheme: weka.clusterers.EM -I 100 -N 3 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Relation: test

Instances: 8

Attributes: 9

t1

t2

t3

t4

t5

t6

t7

t8

t9

Test mode: evaluate on training data

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

EM

==

Number of clusters: 3

Number of iterations performed: 0

Cluster

Attribute 0 1 2

(0.13) (0.38) (0.5)

=================================

t1

mean 5 2 2.5

std. dev. 0 1.3025 1.291

t2

mean 4 1.6667 3.75

std. dev. 0 0.5774 0.9574

t3

mean 5 2.6667 1.75

std. dev. 0 0.5774 0.9574

t4

mean 5 2 1.75

std. dev. 0 1.5811 1.7078

t5

mean 4 1.6667 3.25

std. dev. 0 0.5774 0.5

t6

mean 5 3 2

std. dev. 0 1.7321 0.8165

t7

mean 7 1.3333 2.5

std. dev. 0 0.5774 1.291

t8

mean 4 1.3333 3.75

std. dev. 0 0.5774 0.9574

t9

mean 5 4 1.75

std. dev. 0 1.7321 0.5

Time taken to build model (full training data) : 0.01 seconds

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

Clustered Instances

0 1 ( 13%)

1 3 ( 38%)

2 4 ( 50%)

Log likelihood: 4.41901

**Cobweb algorithm**

**Using training set**

=== Run information ===

Scheme: weka.clusterers.Cobweb -A 1.0 -C 0.0028209479177387815 -S 42

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

class

Test mode: evaluate on training data

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

Number of merges: 2

Number of splits: 2

Number of clusters: 4

node 0 [150]

| leaf 1 [50]

node 0 [150]

| leaf 2 [50]

node 0 [150]

| leaf 3 [50]

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

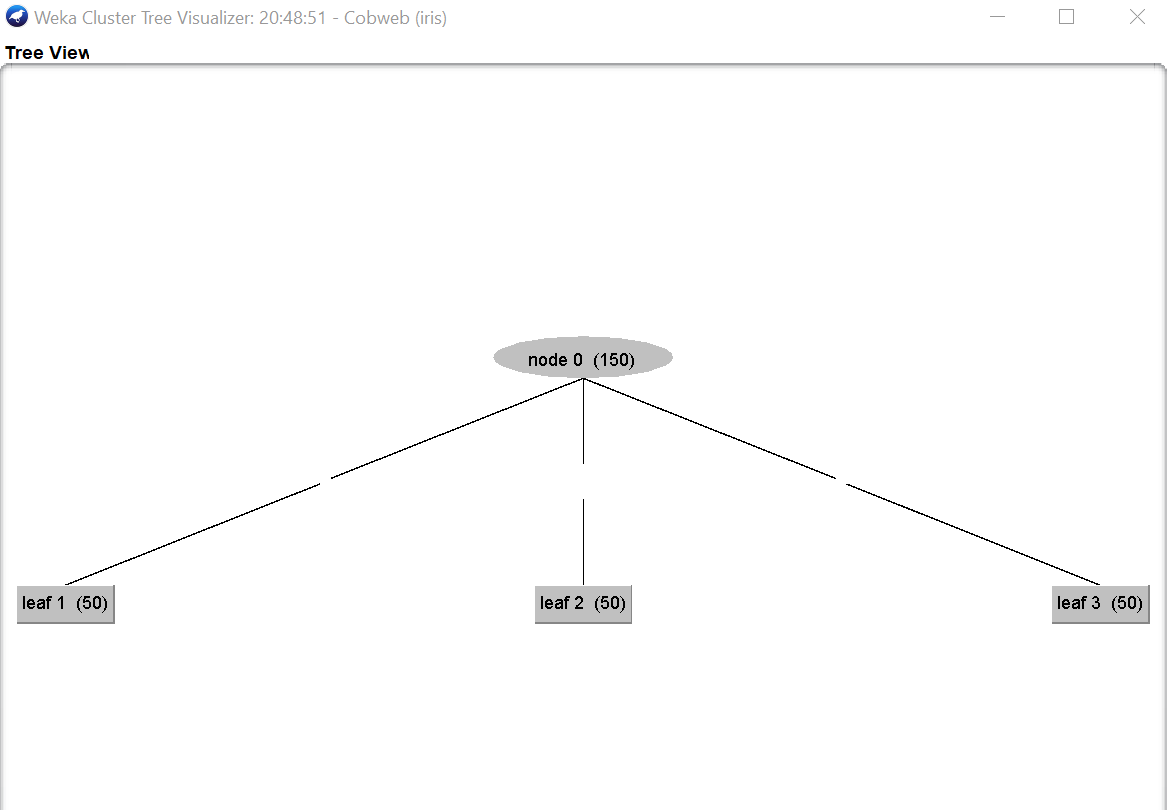
=== Model and evaluation on training set ===

Clustered Instances

1 50 ( 33%)

2 50 ( 33%)

3 50 ( 33%)



**Using Classes to cluster evaluation**

=== Run information ===

Scheme: weka.clusterers.Cobweb -A 1.0 -C 0.0028209479177387815 -S 42

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

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

Number of merges: 0

Number of splits: 0

Number of clusters: 3

node 0 [150]

| leaf 1 [97]

node 0 [150]

| leaf 2 [53]

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

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

Clustered Instances

1 100 ( 67%)

2 50 ( 33%)

Class attribute: class

Classes to Clusters:

1 2 <-- assigned to cluster

0 50 | Iris-setosa

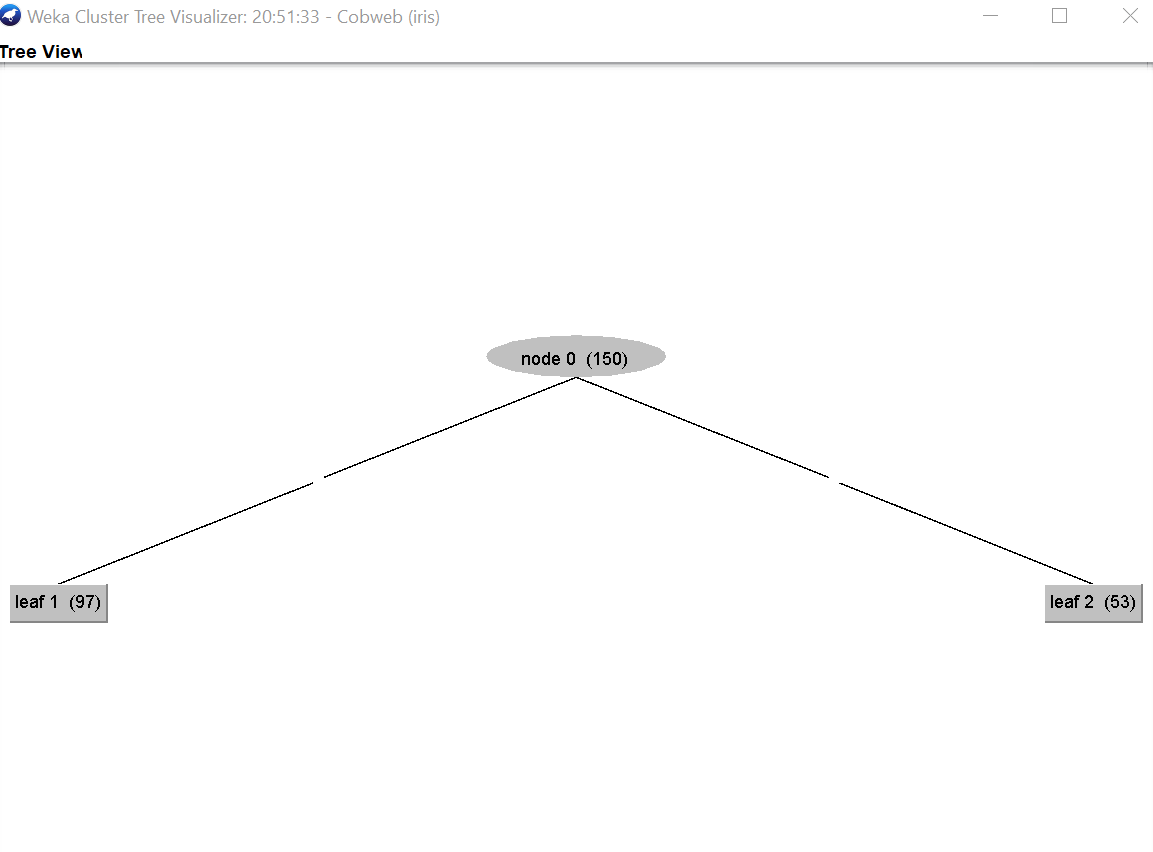
50 0 | Iris-versicolor

50 0 | Iris-virginica

Cluster 1 <-- Iris-versicolor

Cluster 2 <-- Iris-setosa

Incorrectly clustered instances : 50.0 33.3333 %



**K-means and EM (with Classes to cluster evaluation option) for the different number of cluster**

**Kmeans numclusters = 10**

=== Run information ===

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

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

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

kMeans

======

Number of iterations: 7

Within cluster sum of squared errors: 3.192318466613457

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4

Cluster 1: 6.2,2.9,4.3,1.3

Cluster 2: 6.9,3.1,5.1,2.3

Cluster 3: 5.5,4.2,1.4,0.2

Cluster 4: 6.9,3.1,4.9,1.5

Cluster 5: 6.1,3,4.6,1.4

Cluster 6: 4.9,3.1,1.5,0.1

Cluster 7: 4.4,3,1.3,0.2

Cluster 8: 5.5,2.4,3.7,1

Cluster 9: 4.3,3,1.1,0.1

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1 2 3 4 5 6 7 8 9

(150.0) (18.0) (16.0) (25.0) (13.0) (16.0) (12.0) (20.0) (9.0) (13.0) (8.0)

======================================================================================================================================

sepallength 5.8433 6.1278 5.85 7.012 5.3692 6.6438 5.9417 5.045 4.8333 5.3385 4.5125

sepalwidth 3.054 2.6333 2.775 3.164 3.9077 3.0188 3.0583 3.43 2.9667 2.4077 3.1

petallength 3.7587 5.1611 4.1875 5.908 1.5231 4.925 4.65 1.465 1.4667 3.7231 1.3625

petalwidth 1.1987 1.8333 1.25 2.204 0.2846 1.5813 1.6 0.27 0.1778 1.1538 0.1875

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

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

Clustered Instances

0 18 ( 12%)

1 16 ( 11%)

2 25 ( 17%)

3 13 ( 9%)

4 16 ( 11%)

5 12 ( 8%)

6 20 ( 13%)

7 9 ( 6%)

8 13 ( 9%)

9 8 ( 5%)

Class attribute: class

Classes to Clusters:

0 1 2 3 4 5 6 7 8 9 <-- assigned to cluster

0 0 0 13 0 0 20 9 0 8 | Iris-setosa

3 16 0 0 10 9 0 0 12 0 | Iris-versicolor

15 0 25 0 6 3 0 0 1 0 | Iris-virginica

Cluster 0 <-- No class

Cluster 1 <-- Iris-versicolor

Cluster 2 <-- Iris-virginica

Cluster 3 <-- No class

Cluster 4 <-- No class

Cluster 5 <-- No class

Cluster 6 <-- Iris-setosa

Cluster 7 <-- No class

Cluster 8 <-- No class

Cluster 9 <-- No class

Incorrectly clustered instances : 89.0 59.3333 %

**Kmeans numclusters = 15**

=== Run information ===

Scheme: weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 15 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

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

kMeans

======

Number of iterations: 6

Within cluster sum of squared errors: 2.0360842787232367

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4

Cluster 1: 6.2,2.9,4.3,1.3

Cluster 2: 6.9,3.1,5.1,2.3

Cluster 3: 5.5,4.2,1.4,0.2

Cluster 4: 6.9,3.1,4.9,1.5

Cluster 5: 6.1,3,4.6,1.4

Cluster 6: 4.9,3.1,1.5,0.1

Cluster 7: 4.4,3,1.3,0.2

Cluster 8: 5.5,2.4,3.7,1

Cluster 9: 4.3,3,1.1,0.1

Cluster 10: 6,2.7,5.1,1.6

Cluster 11: 5.7,2.5,5,2

Cluster 12: 4.6,3.1,1.5,0.2

Cluster 13: 7.4,2.8,6.1,1.9

Cluster 14: 5.9,3,5.1,1.8

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

(150.0) (9.0) (12.0) (15.0) (12.0) (10.0) (5.0) (20.0) (1.0) (11.0) (4.0) (8.0) (6.0) (13.0) (11.0) (13.0)

=============================================================================================================================================================================================

sepallength 5.8433 6.2 5.6417 6.6667 5.3917 6.67 5.76 5.05 4.5 5.3909 4.375 6.2375 5.6 4.8154 7.5 6.3

sepalwidth 3.054 2.8667 2.7833 3.2133 3.925 3.07 3.12 3.455 2.3 2.3727 3.025 2.4375 2.6667 3.1 3.0818 2.9231

petallength 3.7587 4.5333 4.1 5.6 1.525 4.67 4.5 1.48 1.3 3.6364 1.275 5.0375 4.95 1.4538 6.3182 5.2231

petalwidth 1.1987 1.3778 1.25 2.3067 0.275 1.47 1.58 0.28 0.3 1.0818 0.175 1.5625 1.9833 0.1769 2.0091 1.8923

Time taken to build model (full training data) : 0.01 seconds

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

Clustered Instances

0 9 ( 6%)

1 12 ( 8%)

2 15 ( 10%)

3 12 ( 8%)

4 10 ( 7%)

5 5 ( 3%)

6 20 ( 13%)

7 1 ( 1%)

8 11 ( 7%)

9 4 ( 3%)

10 8 ( 5%)

11 6 ( 4%)

12 13 ( 9%)

13 11 ( 7%)

14 13 ( 9%)

Class attribute: class

Classes to Clusters:

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 <-- assigned to cluster

0 0 0 12 0 0 20 1 0 4 0 0 13 0 0 | Iris-setosa

8 12 0 0 10 5 0 0 11 0 4 0 0 0 0 | Iris-versicolor

1 0 15 0 0 0 0 0 0 0 4 6 0 11 13 | Iris-virginica

Cluster 0 <-- No class

Cluster 1 <-- Iris-versicolor

Cluster 2 <-- Iris-virginica

Cluster 3 <-- No class

Cluster 4 <-- No class

Cluster 5 <-- No class

Cluster 6 <-- Iris-setosa

Cluster 7 <-- No class

Cluster 8 <-- No class

Cluster 9 <-- No class

Cluster 10 <-- No class

Cluster 11 <-- No class

Cluster 12 <-- No class

Cluster 13 <-- No class

Cluster 14 <-- No class

Incorrectly clustered instances : 103.0 68.6667 %

**Kmeans numclusters = 20**

=== Run information ===

Scheme: weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 20 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

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

kMeans

======

Number of iterations: 7

Within cluster sum of squared errors: 1.487620554620973

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4

Cluster 1: 6.2,2.9,4.3,1.3

Cluster 2: 6.9,3.1,5.1,2.3

Cluster 3: 5.5,4.2,1.4,0.2

Cluster 4: 6.9,3.1,4.9,1.5

Cluster 5: 6.1,3,4.6,1.4

Cluster 6: 4.9,3.1,1.5,0.1

Cluster 7: 4.4,3,1.3,0.2

Cluster 8: 5.5,2.4,3.7,1

Cluster 9: 4.3,3,1.1,0.1

Cluster 10: 6,2.7,5.1,1.6

Cluster 11: 5.7,2.5,5,2

Cluster 12: 4.6,3.1,1.5,0.2

Cluster 13: 7.4,2.8,6.1,1.9

Cluster 14: 5.9,3,5.1,1.8

Cluster 15: 6.9,3.2,5.7,2.3

Cluster 16: 6.7,3.3,5.7,2.5

Cluster 17: 7.2,3.6,6.1,2.5

Cluster 18: 7.3,2.9,6.3,1.8

Cluster 19: 6.1,2.8,4.7,1.2

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

(150.0) (8.0) (7.0) (7.0) (12.0) (9.0) (6.0) (20.0) (1.0) (8.0) (4.0) (8.0) (6.0) (13.0) (4.0) (9.0) (7.0) (5.0) (3.0) (4.0) (9.0)

====================================================================================================================================================================================================================================================

sepallength 5.8433 6.2125 5.7286 6.5429 5.3917 6.7111 5.85 5.05 4.5 5.3125 4.375 6.2375 5.6 4.8154 7.675 6.2333 6.8571 6.38 7.6 7.275 5.6111

sepalwidth 3.054 2.875 2.8857 2.9714 3.925 3.0444 3.15 3.455 2.3 2.3125 3.025 2.4375 2.6667 3.1 2.85 2.9111 3.1429 3.32 3.7333 2.975 2.6222

petallength 3.7587 4.6 4.1 5.4286 1.525 4.6667 4.5333 1.48 1.3 3.575 1.275 5.0375 4.95 1.4538 6.575 5.1556 5.6143 5.6 6.4 6.05 3.9889

petalwidth 1.1987 1.3875 1.2857 2.1286 0.275 1.4556 1.5833 0.28 0.3 1.0625 0.175 1.5625 1.9833 0.1769 2.175 1.8111 2.2286 2.4 2.2333 1.775 1.1889

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

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

Clustered Instances

0 8 ( 5%)

1 7 ( 5%)

2 7 ( 5%)

3 12 ( 8%)

4 9 ( 6%)

5 6 ( 4%)

6 20 ( 13%)

7 1 ( 1%)

8 8 ( 5%)

9 4 ( 3%)

10 8 ( 5%)

11 6 ( 4%)

12 13 ( 9%)

13 4 ( 3%)

14 9 ( 6%)

15 7 ( 5%)

16 5 ( 3%)

17 3 ( 2%)

18 4 ( 3%)

19 9 ( 6%)

Class attribute: class

Classes to Clusters:

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 <-- assigned to cluster

0 0 0 12 0 0 20 1 0 4 0 0 13 0 0 0 0 0 0 0 | Iris-setosa

7 7 0 0 9 6 0 0 8 0 4 0 0 0 0 0 0 0 0 9 | Iris-versicolor

1 0 7 0 0 0 0 0 0 0 4 6 0 4 9 7 5 3 4 0 | Iris-virginica

Cluster 0 <-- No class

Cluster 1 <-- No class

Cluster 2 <-- No class

Cluster 3 <-- No class

Cluster 4 <-- No class

Cluster 5 <-- No class

Cluster 6 <-- Iris-setosa

Cluster 7 <-- No class

Cluster 8 <-- No class

Cluster 9 <-- No class

Cluster 10 <-- No class

Cluster 11 <-- No class

Cluster 12 <-- No class

Cluster 13 <-- No class

Cluster 14 <-- Iris-virginica

Cluster 15 <-- No class

Cluster 16 <-- No class

Cluster 17 <-- No class

Cluster 18 <-- No class

Cluster 19 <-- Iris-versicolor

Incorrectly clustered instances : 112.0 74.6667 %

**EM numclusters = 10**

=== Run information ===

Scheme: weka.clusterers.EM -I 100 -N 10 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

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

EM

==

Number of clusters: 10

Number of iterations performed: 60

Cluster

Attribute 0 1 2 3 4 5 6 7 8 9

(0.18) (0.1) (0.06) (0.08) (0.11) (0.02) (0.12) (0.12) (0.06) (0.15)

==============================================================================================

sepallength

mean 4.7754 6.6258 6.4074 5.8684 6.3236 7.6004 6.0603 5.5098 7.4426 5.2826

std. dev. 0.2408 0.2352 0.1615 0.2895 0.4461 0.2943 0.2826 0.4114 0.2277 0.2408

sepalwidth

mean 3.1794 3.1773 2.8114 2.8544 3.0421 3.7335 2.7731 2.4377 2.9199 3.7041

std. dev. 0.26 0.1371 0.1903 0.0963 0.162 0.0942 0.2426 0.1889 0.1636 0.2858

petallength

mean 1.4194 5.5371 5.5575 4.196 4.5752 6.4003 4.9827 3.8809 6.2769 1.5174

std. dev. 0.1691 0.2695 0.144 0.2701 0.1662 0.2449 0.1171 0.4187 0.3623 0.1592

petalwidth

mean 0.1949 2.2676 1.8587 1.2777 1.4669 2.2331 1.8129 1.1799 1.9859 0.3029

std. dev. 0.0556 0.1476 0.2259 0.0422 0.0747 0.2054 0.2151 0.1966 0.2258 0.1212

Time taken to build model (full training data) : 0.04 seconds

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

Clustered Instances

0 28 ( 19%)

1 16 ( 11%)

2 8 ( 5%)

3 12 ( 8%)

4 17 ( 11%)

5 3 ( 2%)

6 17 ( 11%)

7 18 ( 12%)

8 9 ( 6%)

9 22 ( 15%)

Log likelihood: -1.08508

Class attribute: class

Classes to Clusters:

0 1 2 3 4 5 6 7 8 9 <-- assigned to cluster

28 0 0 0 0 0 0 0 0 22 | Iris-setosa

0 0 0 12 17 0 4 17 0 0 | Iris-versicolor

0 16 8 0 0 3 13 1 9 0 | Iris-virginica

Cluster 0 <-- Iris-setosa

Cluster 1 <-- Iris-virginica

Cluster 2 <-- No class

Cluster 3 <-- No class

Cluster 4 <-- No class

Cluster 5 <-- No class

Cluster 6 <-- No class

Cluster 7 <-- Iris-versicolor

Cluster 8 <-- No class

Cluster 9 <-- No class

Incorrectly clustered instances : 89.0 59.3333 %

**EM numclusters = 15**

=== Run information ===

Scheme: weka.clusterers.EM -I 100 -N 15 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

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

EM

==

Number of clusters: 15

Number of iterations performed: 0

Cluster

Attribute 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

(0.01) (0.03) (0.11) (0.11) (0.13) (0.07) (0.07) (0.08) (0.1) (0.02) (0.05) (0.08) (0.05) (0.05) (0.05)

======================================================================================================================================

sepallength

mean 4.5 6.38 6.5 4.7118 5.6368 5.4 5.38 6.0667 4.96 7.6 5.3286 6.75 6.2429 6.1625 7.475

std. dev. 0 0.1924 0.3011 0.2261 0.2733 0.2708 0.3615 0.2774 0.1805 0.3606 0.138 0.1883 0.3599 0.1302 0.2252

sepalwidth

mean 2.3 3.32 3.0375 3.0824 2.7947 3.97 2.36 2.9083 3.44 3.7333 3.5571 3.1 2.6571 2.5125 2.9125

std. dev. 0 0.0837 0.1746 0.0951 0.1715 0.2058 0.1713 0.1676 0.091 0.1155 0.1397 0.1044 0.1512 0.2532 0.1808

petallength

mean 1.3 5.6 4.6063 1.4118 4.1789 1.53 3.6 5.0833 1.4733 6.4 1.5 5.5083 5.3429 4.9125 6.3125

std. dev. 0 0.2739 0.2016 0.1409 0.2551 0.2058 0.3367 0.2949 0.2017 0.3 0.1155 0.3029 0.3259 0.3834 0.3834

petalwidth

mean 0.3 2.4 1.45 0.1765 1.3316 0.29 1.06 1.8333 0.28 2.2333 0.2571 2.1833 2.0429 1.4375 1.975

std. dev. 0 0.1 0.1155 0.0562 0.1565 0.1101 0.0966 0.0651 0.1265 0.2517 0.0976 0.1337 0.207 0.1302 0.2493

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

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

Clustered Instances

0 1 ( 1%)

1 5 ( 3%)

2 16 ( 11%)

3 17 ( 11%)

4 20 ( 13%)

5 10 ( 7%)

6 9 ( 6%)

7 12 ( 8%)

8 15 ( 10%)

9 3 ( 2%)

10 7 ( 5%)

11 12 ( 8%)

12 7 ( 5%)

13 8 ( 5%)

14 8 ( 5%)

Log likelihood: -0.64309

Class attribute: class

Classes to Clusters:

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 <-- assigned to cluster

1 0 0 17 0 10 0 0 15 0 7 0 0 0 0 | Iris-setosa

0 0 16 0 19 0 9 1 0 0 0 0 0 5 0 | Iris-versicolor

0 5 0 0 1 0 0 11 0 3 0 12 7 3 8 | Iris-virginica

Cluster 0 <-- No class

Cluster 1 <-- No class

Cluster 2 <-- No class

Cluster 3 <-- Iris-setosa

Cluster 4 <-- Iris-versicolor

Cluster 5 <-- No class

Cluster 6 <-- No class

Cluster 7 <-- No class

Cluster 8 <-- No class

Cluster 9 <-- No class

Cluster 10 <-- No class

Cluster 11 <-- Iris-virginica

Cluster 12 <-- No class

Cluster 13 <-- No class

Cluster 14 <-- No class

Incorrectly clustered instances : 102.0 68 %

**EM numclusters =20**

=== Run information ===

Scheme: weka.clusterers.EM -I 100 -N 20 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Relation: iris

Instances: 150

Attributes: 5

sepallength

sepalwidth

petallength

petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

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

EM

==

Number of clusters: 20

Number of iterations performed: 3

Cluster

Attribute 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

(0.03) (0.1) (0.04) (0.09) (0.06) (0.02) (0.05) (0.07) (0.03) (0.03) (0.02) (0.09) (0.07) (0.03) (0.04) (0.09) (0.03) (0.05) (0.03) (0.03)

==============================================================================================================================================================================

sepallength

mean 4.4017 6.6209 6.4339 5.7103 6.2485 7.6004 5.736 5.6535 7.2435 5.071 5.2869 4.8187 6.6244 5 5.5237 5.0401 5.1246 6.1751 6.1194 7.675

std. dev. 0.0659 0.232 0.1372 0.2622 0.2692 0.2943 0.3809 0.168 0.1008 0.2279 0.147 0.1341 0.2429 0.0707 0.2077 0.2322 0.1639 0.1195 0.1776 0.0433

sepalwidth

mean 2.8825 3.1772 2.7108 2.8751 3.0096 3.7335 2.6713 2.5025 2.9795 3.7499 3.6837 3.1082 2.9986 2.3 4.048 3.4462 3.4001 2.5048 3.1464 2.85

std. dev. 0.3102 0.14 0.159 0.1164 0.1009 0.0942 0.1209 0.1603 0.1343 0.0904 0.0946 0.0882 0.1389 0.1871 0.1979 0.0714 0.0708 0.253 0.206 0.1658

petallength

mean 1.279 5.539 5.4182 4.2023 5.1203 6.4003 4.9556 3.9288 6.023 1.5298 1.4979 1.452 4.6004 3.275 1.4627 1.4744 1.5999 4.8789 4.5904 6.575

std. dev. 0.0999 0.2668 0.302 0.2622 0.2941 0.2449 0.1973 0.2125 0.1731 0.2722 0.0812 0.1133 0.1809 0.1785 0.1696 0.1761 0.071 0.3846 0.1155 0.2947

petalwidth

mean 0.2003 2.271 1.9495 1.3391 1.804 2.2331 1.9275 1.1297 1.8335 0.3248 0.2019 0.1783 1.4164 1.025 0.2783 0.222 0.4744 1.4376 1.5344 2.175

std. dev. 0.0636 0.1436 0.1495 0.0984 0.065 0.2054 0.22 0.1179 0.1592 0.084 0.0136 0.0571 0.0731 0.0433 0.1125 0.0415 0.0838 0.1035 0.1181 0.1299

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

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

Clustered Instances

0 5 ( 3%)

1 15 ( 10%)

2 6 ( 4%)

3 14 ( 9%)

4 9 ( 6%)

5 3 ( 2%)

6 8 ( 5%)

7 11 ( 7%)

8 5 ( 3%)

9 3 ( 2%)

10 4 ( 3%)

11 13 ( 9%)

12 10 ( 7%)

13 4 ( 3%)

14 7 ( 5%)

15 14 ( 9%)

16 4 ( 3%)

17 6 ( 4%)

18 5 ( 3%)

19 4 ( 3%)

Log likelihood: -0.52946

Class attribute: class

Classes to Clusters:

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 <-- assigned to cluster

5 0 0 0 0 0 0 0 0 3 4 13 0 0 7 14 4 0 0 0 | Iris-setosa

0 0 0 14 2 0 1 11 0 0 0 0 10 4 0 0 0 3 5 0 | Iris-versicolor

0 15 6 0 7 3 7 0 5 0 0 0 0 0 0 0 0 3 0 4 | Iris-virginica

Cluster 0 <-- No class

Cluster 1 <-- Iris-virginica

Cluster 2 <-- No class

Cluster 3 <-- Iris-versicolor

Cluster 4 <-- No class

Cluster 5 <-- No class

Cluster 6 <-- No class

Cluster 7 <-- No class

Cluster 8 <-- No class

Cluster 9 <-- No class

Cluster 10 <-- No class

Cluster 11 <-- No class

Cluster 12 <-- No class

Cluster 13 <-- No class

Cluster 14 <-- No class

Cluster 15 <-- Iris-setosa

Cluster 16 <-- No class

Cluster 17 <-- No class

Cluster 18 <-- No class

Cluster 19 <-- No class

Incorrectly clustered instances : 107.0 71.3333 %

**Result:**

Clustering using WEKA Tool is performed.

**Experiment-6**

**Aim:**

To perform Association rule analysis in Weka tool.

**Tools:** Weka mining tool.

**Procedure:**

1. Implement the Apriori algorithm. Your implementation should allow the user to specify a minimum support threshold (*minsup*), a minimum confidence threshold (*minconf*), and a maximum number of rules to display at a time (*maxrules*).
2. Use your algorithm on the Binarized Lenses problem.
   * Run Apriori for 0.1 <= *minsup* <= 0.8 and 0.1 <= *minconf* <= 0.6, using increments of 0.1 (i.e., this means you should run the algorithm 48 times).
   * Summarize your findings.
   * Given this additional information about the data, what do your (discovered) associations tell us?
3. Use your algorithm on the Mirror Symmetry problem.
   * Run Apriori for various combinations of *minsup* and *minconf* values.
   * Summarize your findings.
   * This is an artificial problem. Each attribute represents a bit position in a string of 30 bits: *Lmost, Lmost1, ..., Lmost14, Rmost14, Rmost13, ..., Rmost1, Rmost* and the attribute *Symm* is 1 if the pattern is symmetric about its center, and 0 otherwise. Given this interpretation, do any of the rules discovered by your Apriori algorithm make sense?
4. Build your own association task.
   * Design your task so that it contains some simple associations you can check your algorithm against. List these associations.
   * Run Apriori for various combinations of *minsup* and *minconf* values.
   * Verify that the associations you designed into the task are discovered by your algorithm.
   * Are there any surprises?

**Output:**

**Use your algorithm on the Binarized Lenses problem.**

**Minsup = 0.3 and minconf = 0.7**

=== Run information ===

Scheme: weka.associations.Apriori -N 10 -T 0 -C 0.7 -D 0.05 -U 1.0 -M 0.3 -S -1.0 -c –

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

=== Associator model (full training set) ===

Apriori

=======

Minimum support: 0.3 (7 instances)

Minimum metric <confidence>: 0.7

Number of cycles performed: 14

Generated sets of large itemsets:

Size of set of large itemsets L(1): 10

Size of set of large itemsets L(2): 5

Best rules found:

1. tear-prod-rate=reduced 12 ==> contact-lenses=none 12 <conf:(1)> lift:(1.6) lev:(0.19) [4] conv:(4.5)

2. contact-lenses=none 15 ==> tear-prod-rate=reduced 12 <conf:(0.8)> lift:(1.6) lev:(0.19) [4] conv:(1.88)

**Minsup = 0.7 and minconf = 0.3**

=== Run information ===

Scheme: weka.associations.Apriori -N 10 -T 0 -C 0.3 -D 0.05 -U 1.0 -M 0.7 -S -1.0 -c -1 1

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

=== Associator model (full training set) ===

No large itemsets and rules found!

**Use your algorithm on the Mirror Symmetry problem.**

**Minsup: 0.2 and Minconf: 10**

=== Run information ===

Scheme: weka.associations.Apriori -N 10 -T 0 -C 10.0 -D 0.05 -U 1.0 -M 0.2 -S -1.0 -c -1

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

=== Associator model (full training set) ===

Apriori

=======

Minimum support: 0.2 (5 instances)

Minimum metric <confidence>: 10

Number of cycles performed: 16

Generated sets of large itemsets:

Size of set of large itemsets L(1): 11

Size of set of large itemsets L(2): 21

Size of set of large itemsets L(3): 6

Best rules found:

**Minsup: 0.4 and Minconf: 10**

=== Run information ===

Scheme: weka.associations.Apriori -N 10 -T 0 -C 10.0 -D 0.05 -U 1.0 -M 0.4 -S -1.0 -c -1

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

=== Associator model (full training set) ===

Apriori

=======

Minimum support: 0.4 (10 instances)

Minimum metric <confidence>: 10

Number of cycles performed: 12

Generated sets of large itemsets:

Size of set of large itemsets L(1): 7

Size of set of large itemsets L(2): 1

Best rules found:

=== Run information ===

Scheme: weka.associations.Apriori -N 10 -T 0 -C 0.5 -D 0.05 -U 1.0 -M 0.25 -S -1.0 -c -1

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

=== Associator model (full training set) ===

Apriori

=======

Minimum support: 0.25 (6 instances)

Minimum metric <confidence>: 0.5

Number of cycles performed: 15

Generated sets of large itemsets:

Size of set of large itemsets L(1): 10

Size of set of large itemsets L(2): 18

Size of set of large itemsets L(3): 4

Best rules found:

1. tear-prod-rate=reduced 12 ==> contact-lenses=none 12 <conf:(1)> lift:(1.6) lev:(0.19) [4] conv:(4.5)

2. spectacle-prescrip=myope tear-prod-rate=reduced 6 ==> contact-lenses=none 6 <conf:(1)> lift:(1.6) lev:(0.09) [2] conv:(2.25)

3. spectacle-prescrip=hypermetrope tear-prod-rate=reduced 6 ==> contact-lenses=none 6 <conf:(1)> lift:(1.6) lev:(0.09) [2] conv:(2.25)

4. astigmatism=no tear-prod-rate=reduced 6 ==> contact-lenses=none 6 <conf:(1)> lift:(1.6) lev:(0.09) [2] conv:(2.25)

5. astigmatism=yes tear-prod-rate=reduced 6 ==> contact-lenses=none 6 <conf:(1)> lift:(1.6) lev:(0.09) [2] conv:(2.25)

6. spectacle-prescrip=myope contact-lenses=none 7 ==> tear-prod-rate=reduced 6 <conf:(0.86)> lift:(1.71) lev:(0.1) [2] conv:(1.75)

7. astigmatism=no contact-lenses=none 7 ==> tear-prod-rate=reduced 6 <conf:(0.86)> lift:(1.71) lev:(0.1) [2] conv:(1.75)

8. contact-lenses=none 15 ==> tear-prod-rate=reduced 12 <conf:(0.8)> lift:(1.6) lev:(0.19) [4] conv:(1.88)

9. age=presbyopic 8 ==> contact-lenses=none 6 <conf:(0.75)> lift:(1.2) lev:(0.04) [1] conv:(1)

10. spectacle-prescrip=hypermetrope contact-lenses=none 8 ==> tear-prod-rate=reduced 6 <conf:(0.75)> lift:(1.5) lev:(0.08) [2] conv:(1.33)

**Result:**

Association Rule Analysis is performed using WEKA tool.