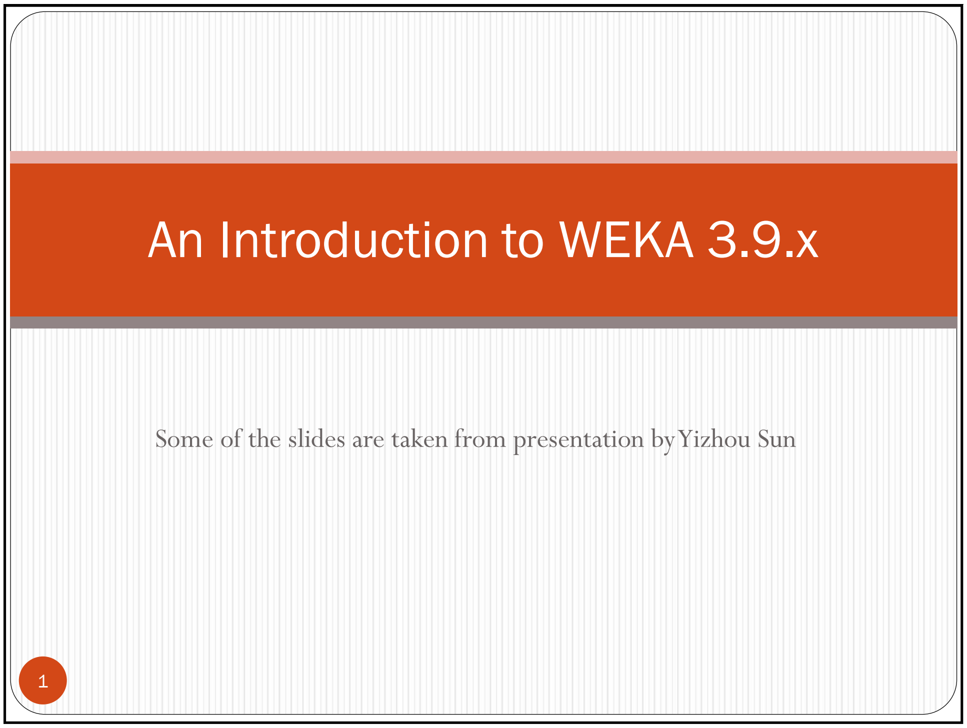
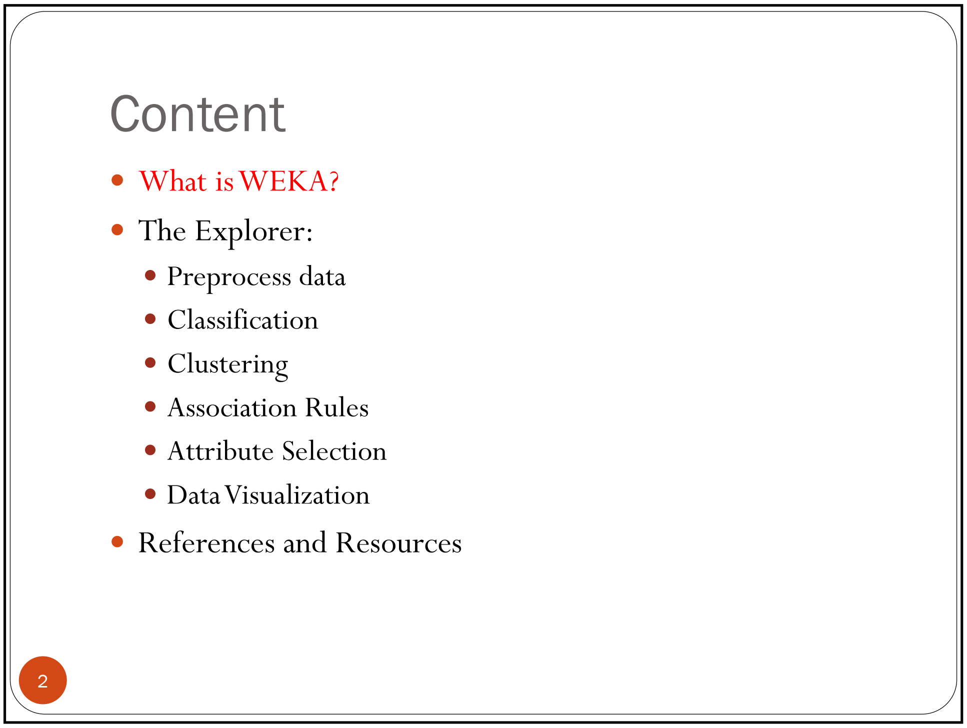


An Introduction to WEKA 3.9.x

Some of the slides are taken from presentation by Yizhou Sun

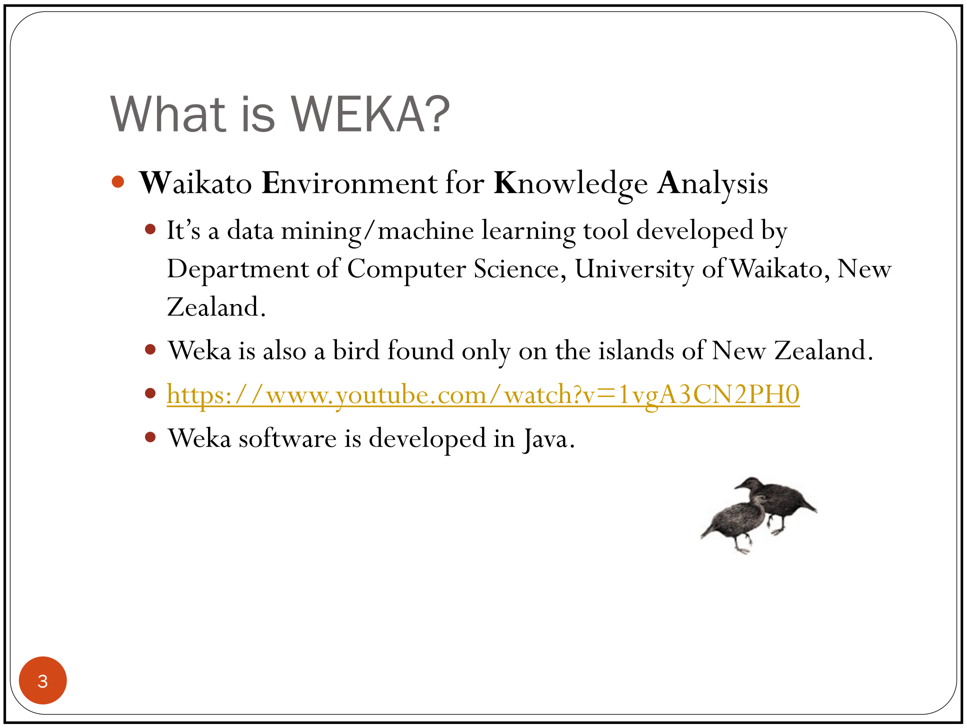
1



Content

- What is WEKA?
- The Explorer:
  - Preprocess data
  - Classification
  - Clustering
  - Association Rules
  - Attribute Selection
  - Data Visualization
- References and Resources

2

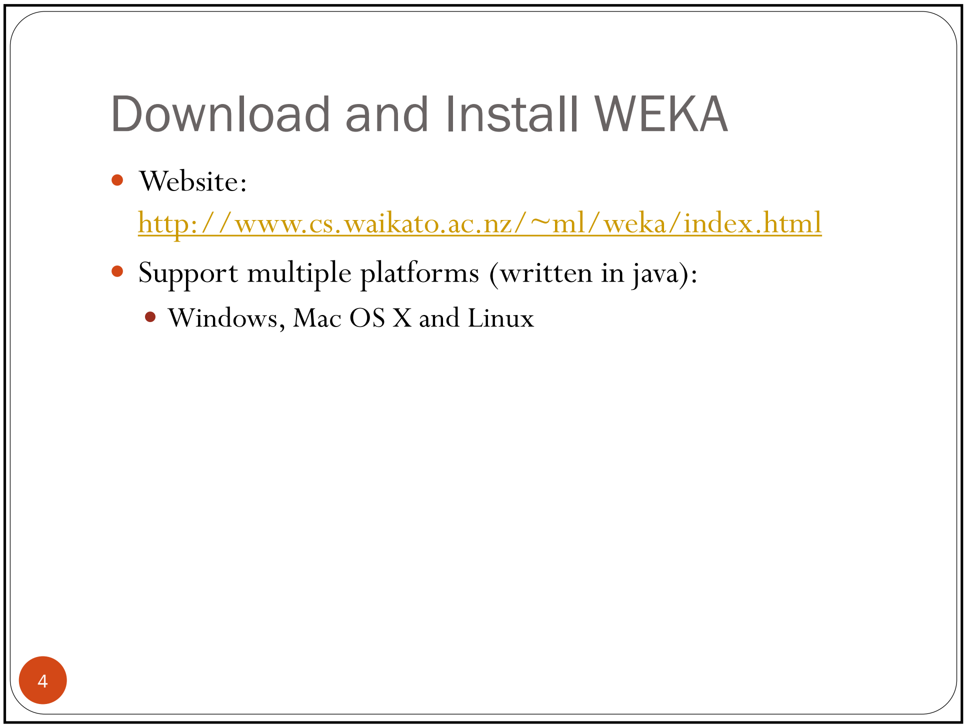


What is WEKA?

- **W**aikato **E**nvironment for **K**nowledge **A**nalysis
  - It's a data mining/machine learning tool developed by Department of Computer Science, University of Waikato, New Zealand.
  - Weka is also a bird found only on the islands of New Zealand.
  - https://www.youtube.com/watch?v=1vgA3CN2PH0
  - Weka software is developed in Java.

3

3



Download and Install WEKA

- Website: http://www.cs.waikato.ac.nz/~ml/weka/index.html
- Support multiple platforms (written in java):
  - Windows, Mac OS X and Linux

4

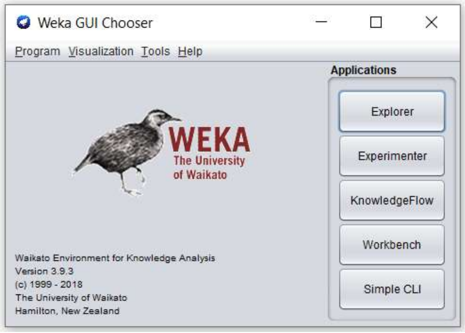
4

Main Features

- 49+ data preprocessing tools
- 76+ classification/regression algorithms
- 8+ clustering algorithms
- 3+ algorithms for finding association rules
- 15+ attribute/subset evaluators + 10+ search algorithms for feature selection

Main GUI

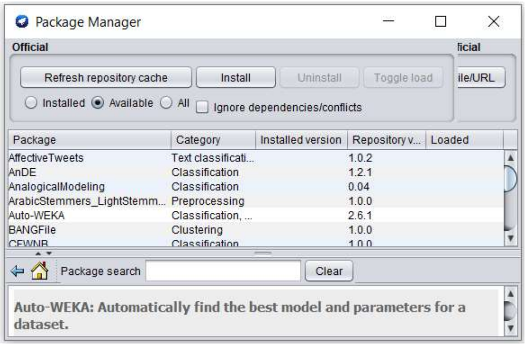
- Four graphical user interfaces
  - “The Explorer” (exploratory data analysis)
  - “The Experimenter” (experimental environment)
  - “The KnowledgeFlow” (new process model inspired interface)
  - “Workbench” (unified GUI that combines above three)
- One old fashioned Command Line Interface (CLI)



Waikato Environment for Knowledge Analysis
Version 3.9.3
(c) 1999 - 2018
The University of Waikato
Hamilton, New Zealand

The package management system

- Weka community keeps adding new algorithms and features.
- These are placed into plugin packages.
- A package management system allows the user to browse and install packages of interest.



Package	Category	Installed version	Repository v...	Loaded
AffectiveTweets	Text classificati...		1.0.2	
AnDE	Classification		1.2.1	
AnalogicalModeling	Classification		0.04	
ArabicStemmers_LightStemm...	Preprocessing		1.0.0	
Auto-WEKA	Classification, ...		2.6.1	
BANGFile	Clustering		1.0.0	
CFWNB	Classification		1.0.0	

Auto-WEKA: Automatically find the best model and parameters for a dataset.

7

7

Content

- What is WEKA?
- The Explorer:
  - Preprocess data
  - Classification
  - Clustering
  - Association Rules
  - Attribute Selection
  - Data Visualization
- References and Resources

8

8

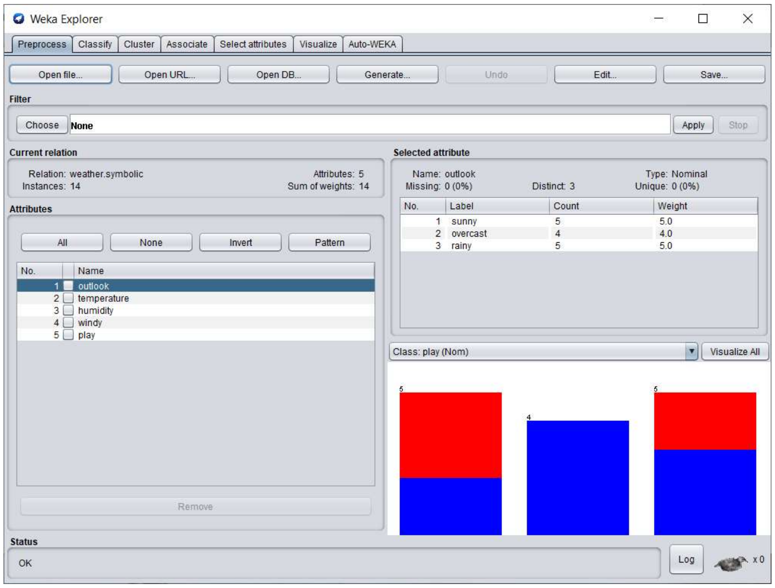
Explorer: pre-processing the data

- Data can be imported from a file in various formats: ARFF, CSV, C4.5, binary
- Data can also be read from a URL or from an SQL database (using JDBC)
- Pre-processing tools in WEKA are called "filters"
- WEKA contains filters for:
  - Discretization, normalization, resampling, attribute selection, transforming and combining attributes, …

9

9

Weka explorer interface



Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize | Auto-WEKA

Filter

None

Current relation

Relation: weather.symbolic
Instances: 14
Attributes: 5
Sum of weights: 14

Selected attribute

Name: outlook
Missing: 0 (0%)
Distinct: 3
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	sunny	5	5.0
2	overcast	4	4.0
3	rainy	5	5.0

Attributes

No.		Name
1	□	outlook
2	□	temperature
3	□	humidity
4	□	windy
5	□	play

Class: play (Nom)

Status

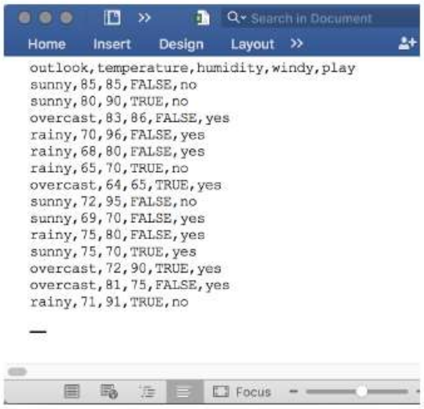
OK

10

10

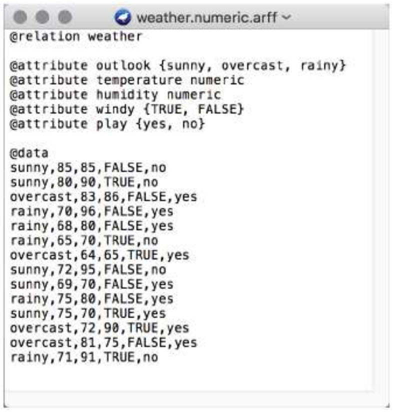
WEKA: Attribute Relation File Format (ARFF)

CSV format



```
outlook,temperature,humidity,windy,play
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
sunny,72,95,FALSE,no
sunny,69,70,FALSE,yes
rainy,75,80,FALSE,yes
sunny,75,70,TRUE,yes
overcast,72,90,TRUE,yes
overcast,81,75,FALSE,yes
rainy,71,91,TRUE,no
```

ARFF format

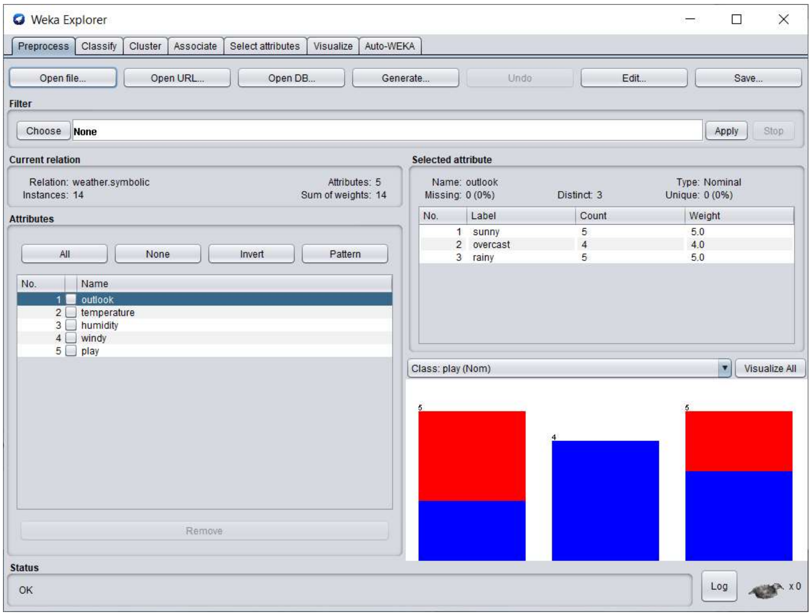


```
@relation weather

@attribute outlook {sunny, overcast, rainy}
@attribute temperature numeric
@attribute humidity numeric
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
sunny,72,95,FALSE,no
sunny,69,70,FALSE,yes
rainy,75,80,FALSE,yes
sunny,75,70,TRUE,yes
overcast,72,90,TRUE,yes
overcast,81,75,FALSE,yes
rainy,71,91,TRUE,no
```

11

11

Load file weather.numeric.arff



Filter: None

Current relation

Relation: weather.symbolic Attributes: 5
Instances: 14 Sum of weights: 14

Attributes

No.	Name
1	outlook
2	temperature
3	humidity
4	windy
5	play

Selected attribute

Name: outlook Type: Nominal
Missing: 0 (0%) Distinct: 3 Unique: 0 (0%)

No.	Label	Count	Weight
1	sunny	5	5.0
2	overcast	4	4.0
3	rainy	5	5.0

Class: play (Nom)

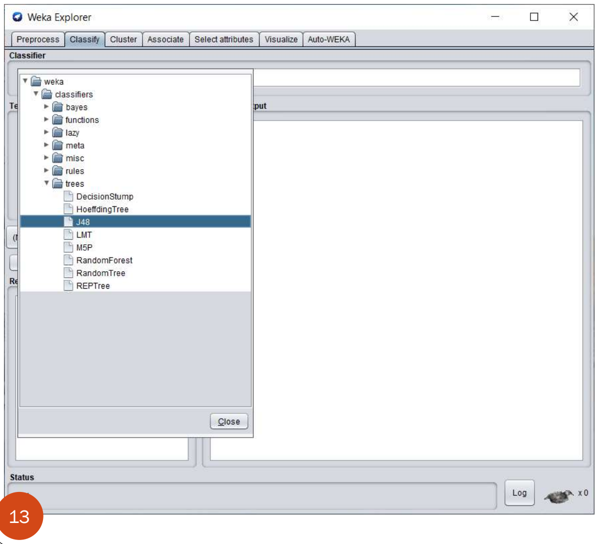
Status

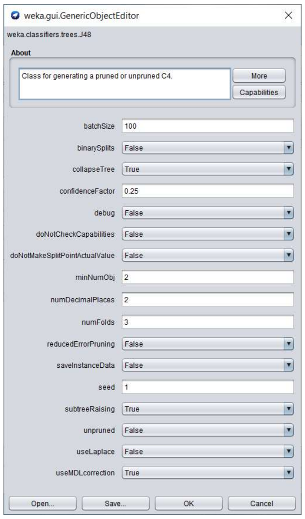
OK

12

12

Building a decision tree – Select J48 (implementation of C4.5 algo)





weka.classifiers.trees.J48

About

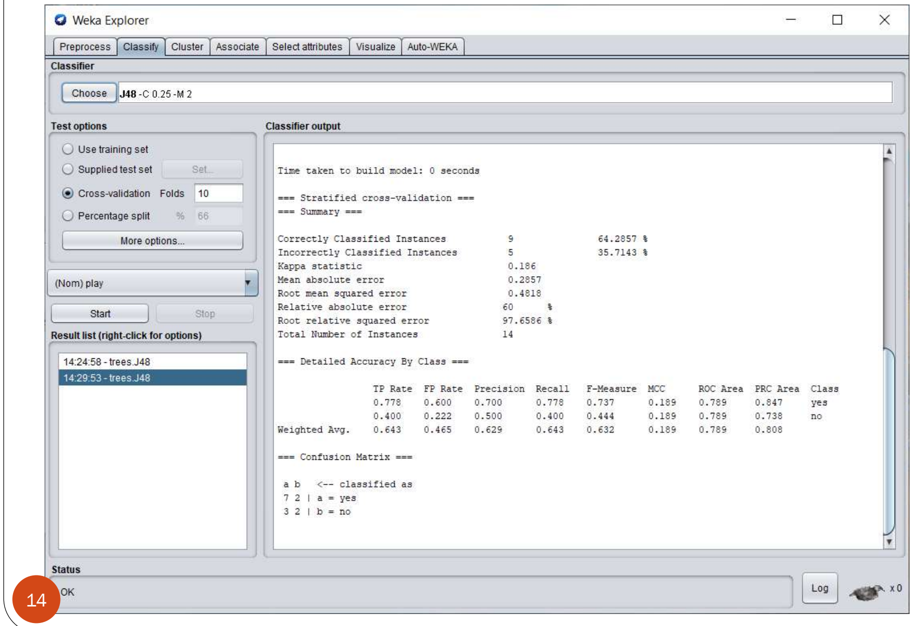
Class for generating a pruned or unpruned C4.

```
batchSize                       100
binarySplits                    False
collapseTree                    True
confidenceFactor                0.25
debug                           False
doNotCheckCapabilities          False
doNotMakeSplitPointActualValue  False
minNumObj                       2
numDecimalPlaces                2
numFolds                        3
reducedErrorPruning             False
saveInstanceData                False
seed                            1
subtreeRaising                  True
unpruned                        False
useLaplace                      False
useMDLcorrection                True
```

13

13

Using C4.5 classifier



Classifier: J48 -C 0.25 -M 2

Test options:

- ○ Use training set
- ○ Supplied test set Set...
- ● Cross-validation Folds 10
- ○ Percentage split % 66

(Nom) play

Result list:

- 14:24:58 - trees.J48
- 14:29:53 - trees.J48

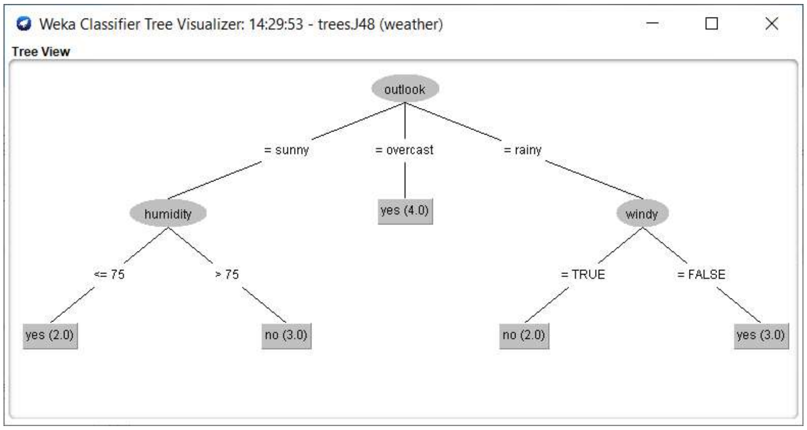
```
Time taken to build model: 0 seconds

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

Correctly Classified Instances           9               64.2857 %
Incorrectly Classified Instances         5               35.7143 %
Kappa statistic                          0.186
Mean absolute error                      0.2857
Root mean squared error                  0.4818
Relative absolute error                 60      %
Root relative squared error             97.6586 %
Total Number of Instances               14

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.778    0.600    0.700      0.778    0.737      0.189    0.789     0.847     yes
                 0.400    0.222    0.500      0.400    0.444      0.189    0.789     0.738     no
Weighted Avg.    0.643    0.465    0.629      0.643    0.632      0.189    0.789     0.808

=== Confusion Matrix ===

 a b   <-- classified as
 7 2 | a = yes
 3 2 | b = no
```

Status: OK

14

14

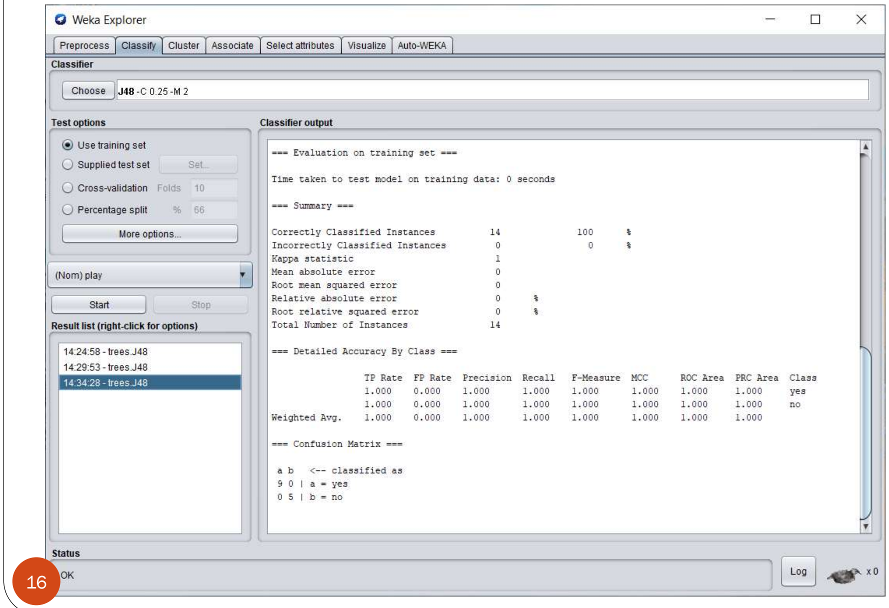
C4.5 Decision tree visualizer



15

15

Using training set for testing



Classifier

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

Test options

- ● Use training set
- ○ Supplied test set
- ○ Cross-validation Folds 10
- ○ Percentage split % 66

(Nom) play

Result list (right-click for options)

- 14:24:58 - trees.J48
- 14:29:53 - trees.J48
- 14:34:28 - trees.J48

Classifier output

```
=== 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

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     yes
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     no
Weighted Avg.    1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000

=== Confusion Matrix ===

 a b   <-- classified as
 9 0 | a = yes
 0 5 | b = no
```

Status

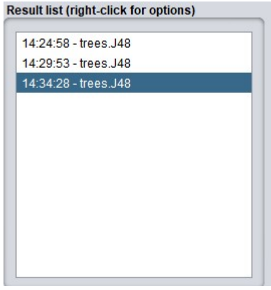
OK

16

16

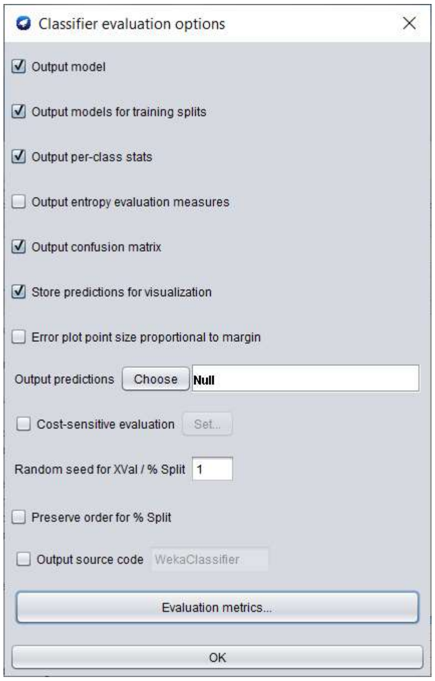
More options

History line



Result list (right-click for options)

14:24:58 - trees.J48
14:29:53 - trees.J48
14:34:28 - trees.J48



Classifier evaluation options

☑ Output model

☑ Output models for training splits

☑ Output per-class stats

□ Output entropy evaluation measures

☑ Output confusion matrix

☑ Store predictions for visualization

□ Error plot point size proportional to margin

Output predictions [Choose] Null

□ Cost-sensitive evaluation [Set...]

Random seed for XVal / % Split 1

□ Preserve order for % Split

□ Output source code WekaClassifier

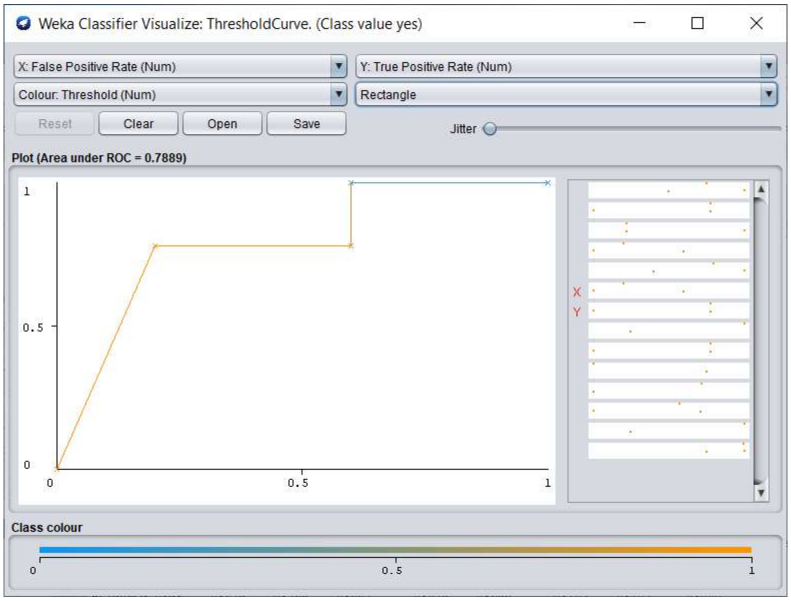
[Evaluation metrics...]

[OK]

17

17

ROC curve – Visualize threshold curve



Weka Classifier Visualize: ThresholdCurve. (Class value yes)

X: False Positive Rate (Num)

Y: True Positive Rate (Num)

Colour: Threshold (Num)

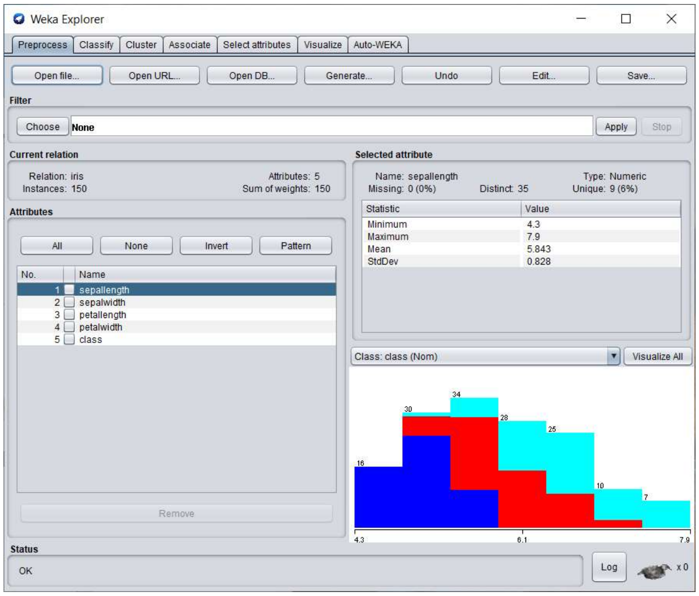
Rectangle

Plot (Area under ROC = 0.7889)

18

18

Working with iris data – iris.arff



Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize | Auto-WEKA

Filter: None

Current relation

Relation: iris Attributes: 5
Instances: 150 Sum of weights: 150

Selected attribute

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

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

Attributes

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

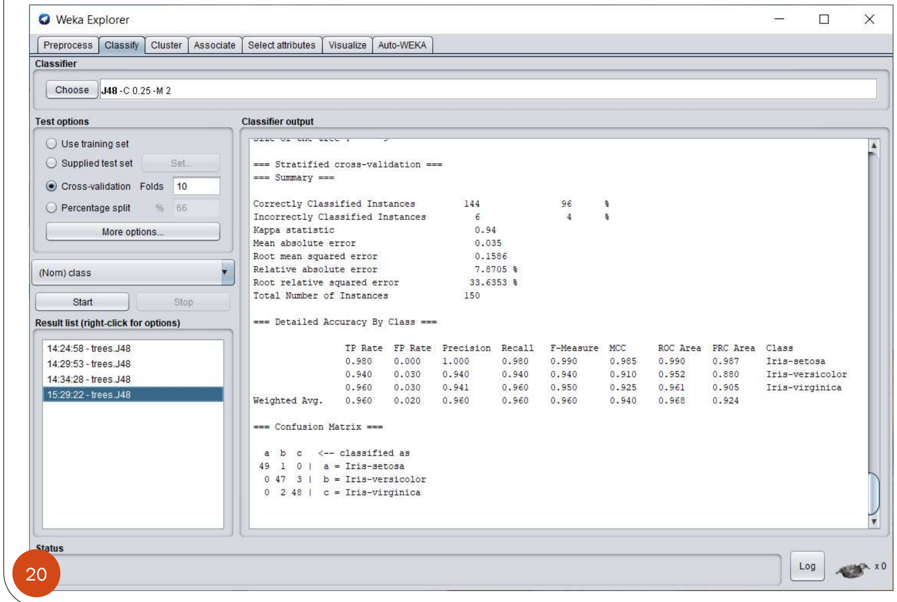
Class: class (Nom)

Status: OK

19

19

Use J48 with cross validation



Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize | Auto-WEKA

Classifier: J48 -C 0.25 -M 2

Test options

○ Use training set
○ Supplied test set
● Cross-validation Folds 10
○ Percentage split % 66

(Nom) class

Result list (right-click for options)

14:24:58 - trees.J48
14:29:53 - trees.J48
14:34:28 - trees.J48
15:29:22 - trees.J48

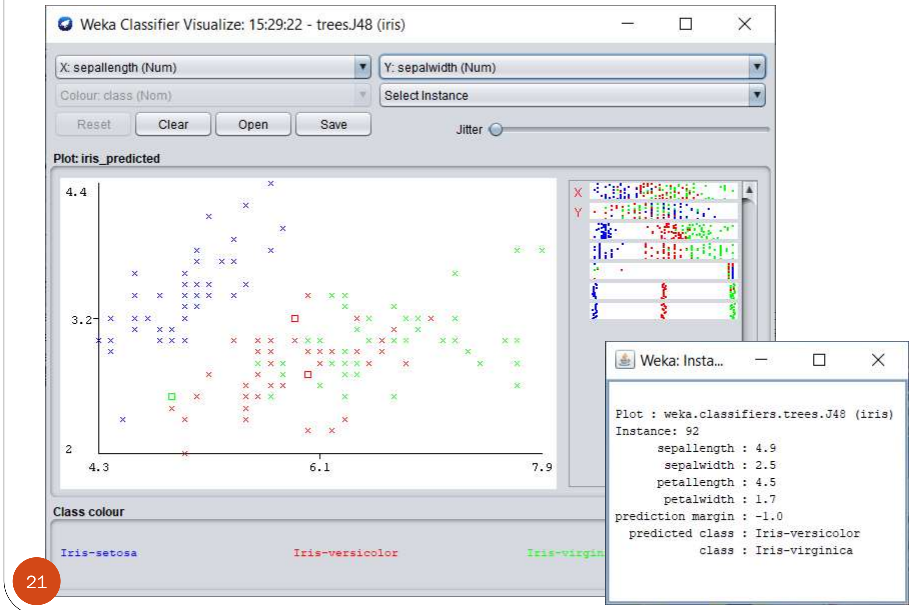
Classifier output

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

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

=== Detailed Accuracy By Class ===

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

=== Confusion Matrix ===

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

20

20

Iris classifier visualization

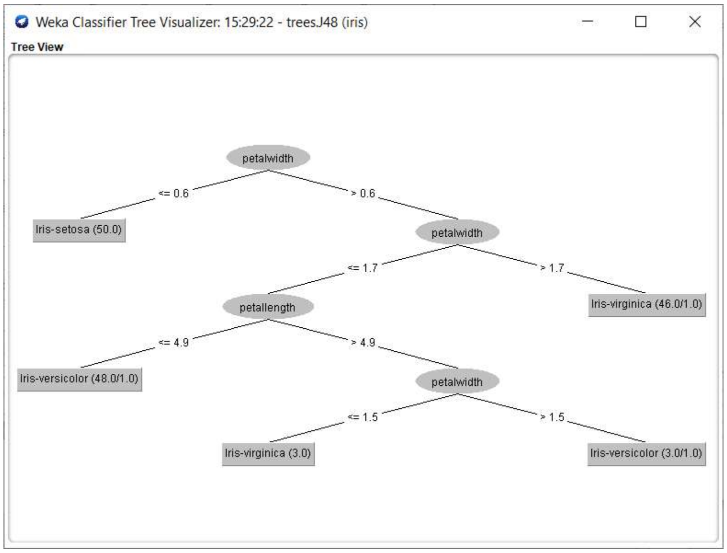


```
Plot : weka.classifiers.trees.J48 (iris)
Instance: 92
      sepallength : 4.9
       sepalwidth : 2.5
      petallength : 4.5
       petalwidth : 1.7
prediction margin : -1.0
  predicted class : Iris-versicolor
            class : Iris-virginica
```

21

21

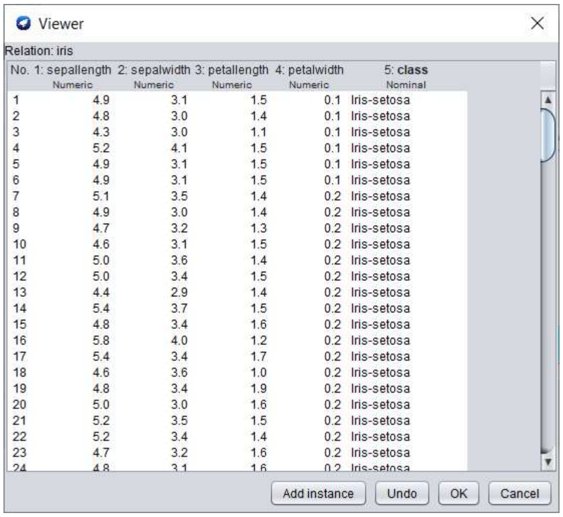
Iris data – J48 tree


Weka Classifier Tree Visualizer: 15:29:22 - trees.J48 (iris)
Tree View
petalwidth
<= 0.6
> 0.6
Iris-setosa (50.0)
petalwidth
<= 1.7
> 1.7
petallength
Iris-virginica (46.0/1.0)
<= 4.9
> 4.9
Iris-versicolor (48.0/1.0)
petalwidth
<= 1.5
> 1.5
Iris-virginica (3.0)
Iris-versicolor (3.0/1.0)

22

22

Open CSV file in Weka



Viewer

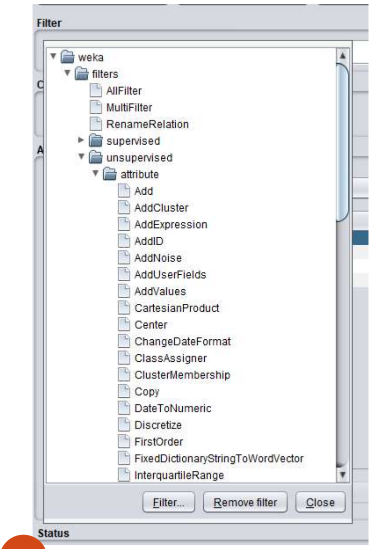
Relation: iris

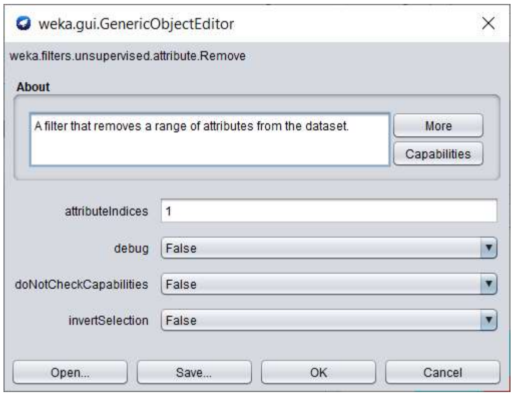
No.	1: sepallength Numeric	2: sepalwidth Numeric	3: petallength Numeric	4: petalwidth Numeric	5: **class** Nominal
1	4.9	3.1	1.5	0.1	Iris-setosa
2	4.8	3.0	1.4	0.1	Iris-setosa
3	4.3	3.0	1.1	0.1	Iris-setosa
4	5.2	4.1	1.5	0.1	Iris-setosa
5	4.9	3.1	1.5	0.1	Iris-setosa
6	4.9	3.1	1.5	0.1	Iris-setosa
7	5.1	3.5	1.4	0.2	Iris-setosa
8	4.9	3.0	1.4	0.2	Iris-setosa
9	4.7	3.2	1.3	0.2	Iris-setosa
10	4.6	3.1	1.5	0.2	Iris-setosa
11	5.0	3.6	1.4	0.2	Iris-setosa
12	5.0	3.4	1.5	0.2	Iris-setosa
13	4.4	2.9	1.4	0.2	Iris-setosa
14	5.4	3.7	1.5	0.2	Iris-setosa
15	4.8	3.4	1.6	0.2	Iris-setosa
16	5.8	4.0	1.2	0.2	Iris-setosa
17	5.4	3.4	1.7	0.2	Iris-setosa
18	4.6	3.6	1.0	0.2	Iris-setosa
19	4.8	3.4	1.9	0.2	Iris-setosa
20	5.0	3.0	1.6	0.2	Iris-setosa
21	5.2	3.5	1.5	0.2	Iris-setosa
22	5.2	3.4	1.4	0.2	Iris-setosa
23	4.7	3.2	1.6	0.2	Iris-setosa
24	4.8	3.1	1.6	0.2	Iris-setosa

23

23

Filter example – Remove attribute





weka.gui.GenericObjectEditor

weka.filters.unsupervised.attribute.Remove

About

A filter that removes a range of attributes from the dataset.

attributeIndices: 1

debug: False

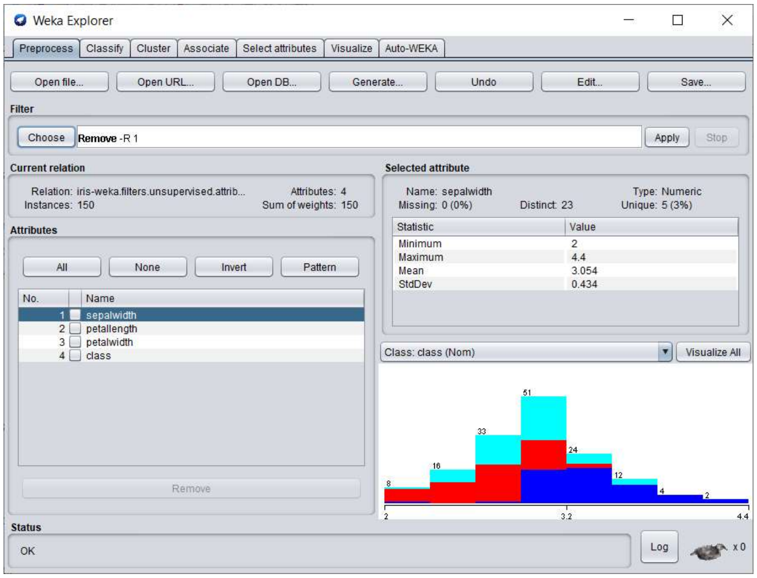
doNotCheckCapabilities: False

invertSelection: False

24

24

Iris data with first attribute removed



Filter

Remove -R 1

Current relation

Relation: iris-weka.filters.unsupervised.attrib... Attributes: 4
Instances: 150 Sum of weights: 150

Selected attribute

Name: sepalwidth Type: Numeric
Missing: 0 (0%) Distinct: 23 Unique: 5 (3%)

Statistic	Value
Minimum	2
Maximum	4.4
Mean	3.054
StdDev	0.434

Attributes

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

Class: class (Nom)

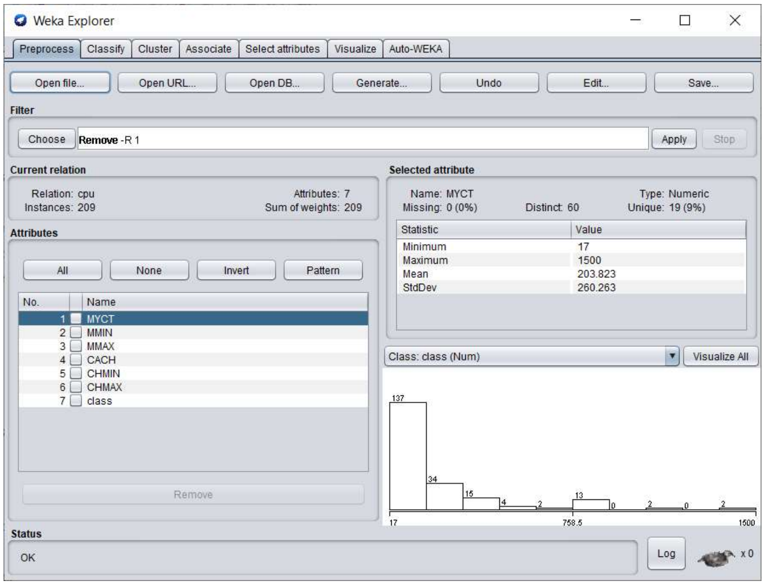
Status

OK

25

25

Load CPU data – cpu.arff



Filter

Remove -R 1

Current relation

Relation: cpu Attributes: 7
Instances: 209 Sum of weights: 209

Selected attribute

Name: MYCT Type: Numeric
Missing: 0 (0%) Distinct: 60 Unique: 19 (9%)

Statistic	Value
Minimum	17
Maximum	1500
Mean	203.823
StdDev	260.263

Attributes

No.		Name
1	□	MYCT
2	□	MMIN
3	□	MMAX
4	□	CACH
5	□	CHMIN
6	□	CHMAX
7	□	class

Class: class (Num)

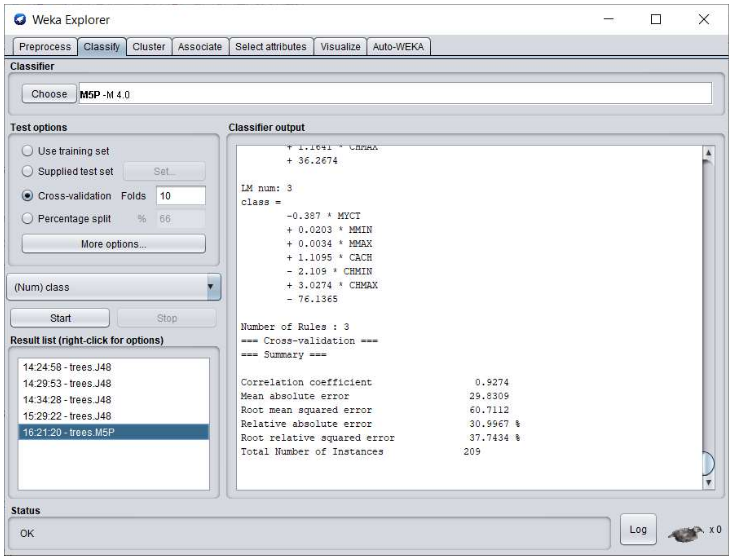
Status

OK

26

26

Choose M5P Quinlan's M5 algorithm Decision trees of regression models



M5P -M 4.0

○ Use training set
○ Supplied test set
● Cross-validation Folds 10
○ Percentage split % 66

(Num) class

Result list (right-click for options)

14:24:58 - trees.J48
14:29:53 - trees.J48
14:34:28 - trees.J48
15:29:22 - trees.J48
16:21:20 - trees.M5P

Classifier output

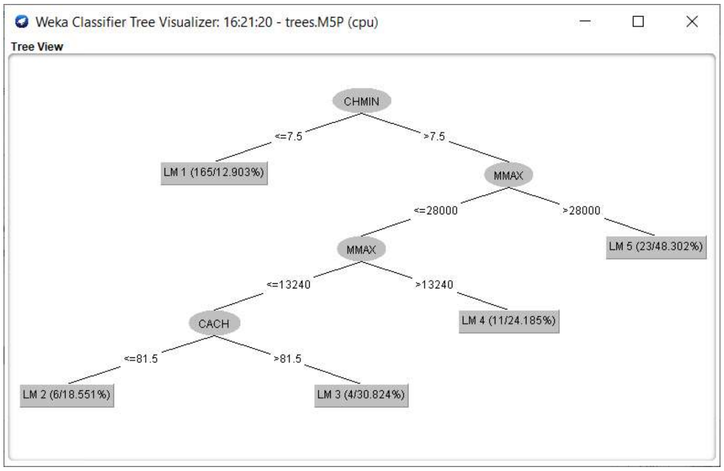
```
        + 1.1641 * CHMAX
        + 36.2674

LM num: 3
class =
        -0.387 * MYCT
        + 0.0203 * MMIN
        + 0.0034 * MMAX
        + 1.1095 * CACH
        - 2.109 * CHMIN
        + 3.0274 * CHMAX
        - 76.1365

Number of Rules : 3
=== Cross-validation ===
=== Summary ===

Correlation coefficient                  0.9274
Mean absolute error                     29.8309
Root mean squared error                 60.7112
Relative absolute error                 30.9967 %
Root relative squared error             37.7434 %
Total Number of Instances              209
```

Status
OK

27

27

M5P linear regression models tree

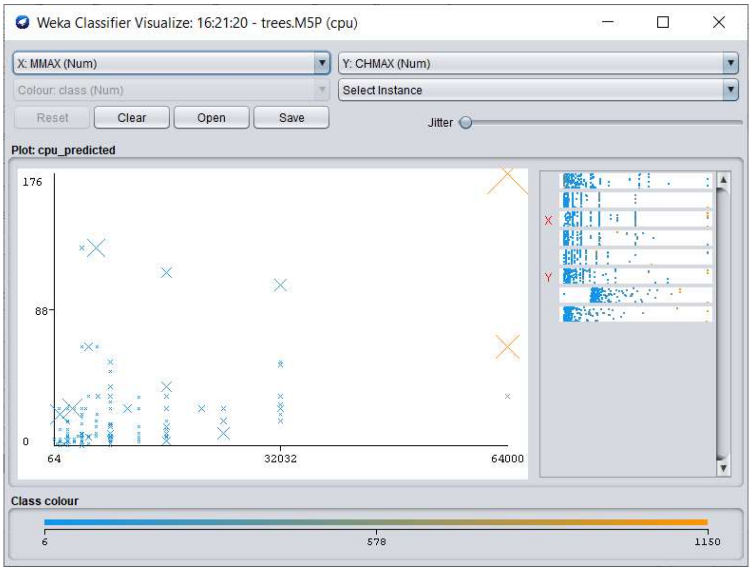


- The first number at each leaf is the number of instances that reach it
- The second is the root mean squared error of the predictions expressed as a percentage of the standard deviation

28

28

Visualization of errors – larger the cross, larger the error – M5P



X: MMAX (Num)

Y: CHMAX (Num)

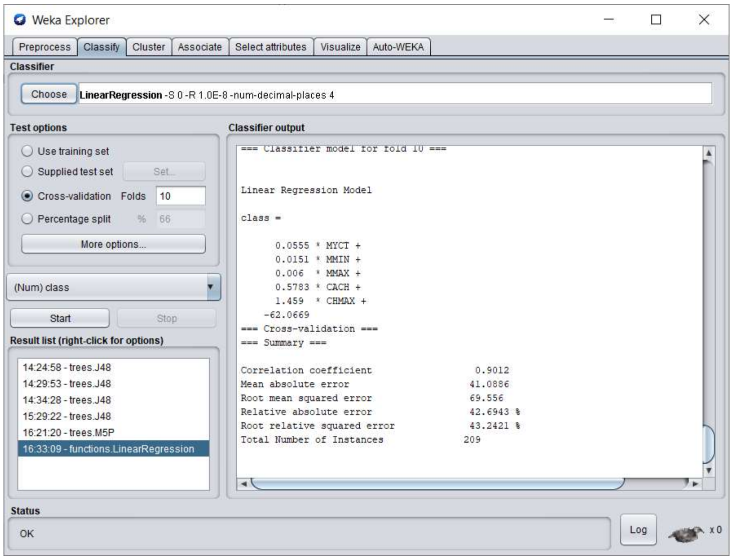
Colour: class (Num)

Select Instance

Plot: cpu_predicted

29

Choose functions -> Linear regression for CPU data



Classifier

LinearRegression -S 0 -R 1.0E-8 -num-decimal-places 4

Test options

○ Use training set

○ Supplied test set

● Cross-validation Folds 10

○ Percentage split % 66

(Num) class

Result list (right-click for options)

14:24:58 - trees.J48
14:29:53 - trees.J48
14:34:28 - trees.J48
15:29:22 - trees.J48
16:21:20 - trees.M5P
16:33:09 - functions.LinearRegression

Classifier output

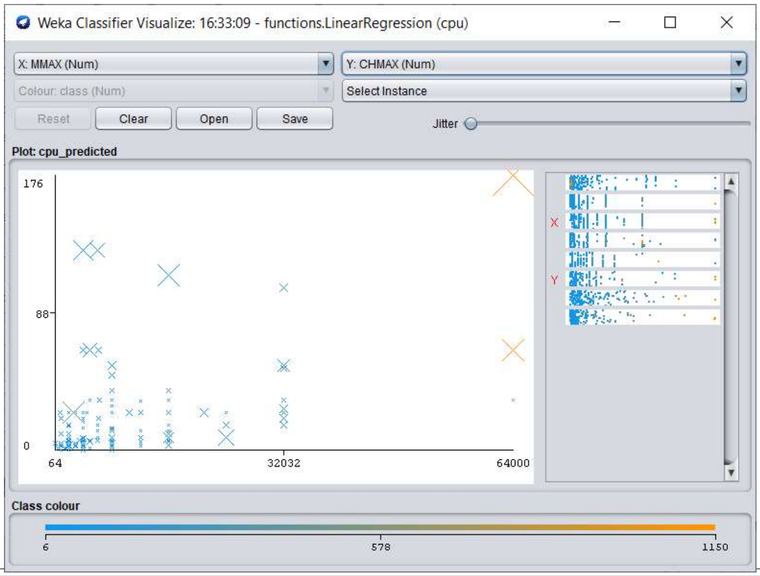
```
=== Classifier model for fold 10 ===


Linear Regression Model

class =

      0.0555 * MYCT +
      0.0151 * MMIN +
      0.006  * MMAX +
      0.5783 * CACH +
      1.459  * CHMAX +
    -62.0669
=== Cross-validation ===
=== Summary ===

Correlation coefficient                  0.9012
Mean absolute error                     41.0886
Root mean squared error                 69.556
Relative absolute error                 42.6943 %
Root relative squared error             43.2421 %
Total Number of Instances              209
```

Status

OK

30

Visualization of errors –linear regression (M5P is superior)



Weka Classifier Visualize: 16:33:09 - functions.LinearRegression (cpu)

X: MMAX (Num)

Y: CHMAX (Num)

Colour: class (Num)

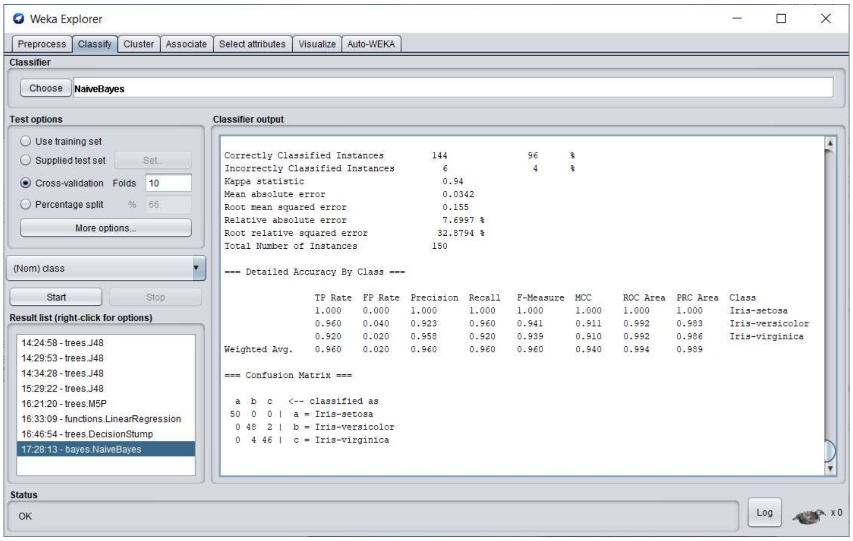
Select Instance

Plot: cpu_predicted

31

31

Naïve Bayes classifier (Iris data)



Weka Explorer

Classifier: NaiveBayes

Test options:

○ Use training set

○ Supplied test set

● Cross-validation Folds 10

○ Percentage split % 66

(Nom) class

Result list (right-click for options)

- 14:24:58 - trees.J48
- 14:29:53 - trees.J48
- 14:34:28 - trees.J48
- 15:29:22 - trees.J48
- 16:21:20 - trees.M5P
- 16:33:09 - functions.LinearRegression
- 16:46:54 - trees.DecisionStump
- 17:28:13 - bayes.NaiveBayes

Classifier output

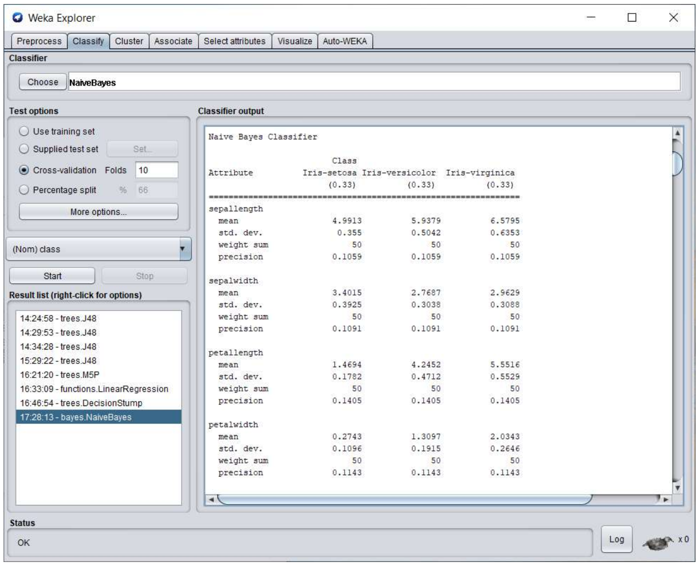
```
Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.0342
Root mean squared error                  0.155
Relative absolute error                  7.6997 %
Root relative squared error             32.8794 %
Total Number of Instances              150

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Iris-setosa
                 0.960    0.040    0.923      0.960    0.941      0.911    0.992     0.983     Iris-versicolor
                 0.920    0.020    0.958      0.920    0.939      0.910    0.992     0.986     Iris-virginica
Weighted Avg.    0.960    0.020    0.960      0.960    0.960      0.940    0.994     0.989

=== Confusion Matrix ===

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

Status: OK

32

32

Naïve Bayes classifier uses normal distribution to model numeric attributes



NaiveBayes

○ Use training set
○ Supplied test set Set...
● Cross-validation Folds 10
○ Percentage split % 66

(Nom) class

14:24:58 - trees.J48
14:29:53 - trees.J48
14:34:28 - trees.J48
15:29:22 - trees.J48
16:21:20 - trees.M5P
16:33:09 - functions.LinearRegression
16:46:54 - trees.DecisionStump
17:28:13 - bayes.NaiveBayes

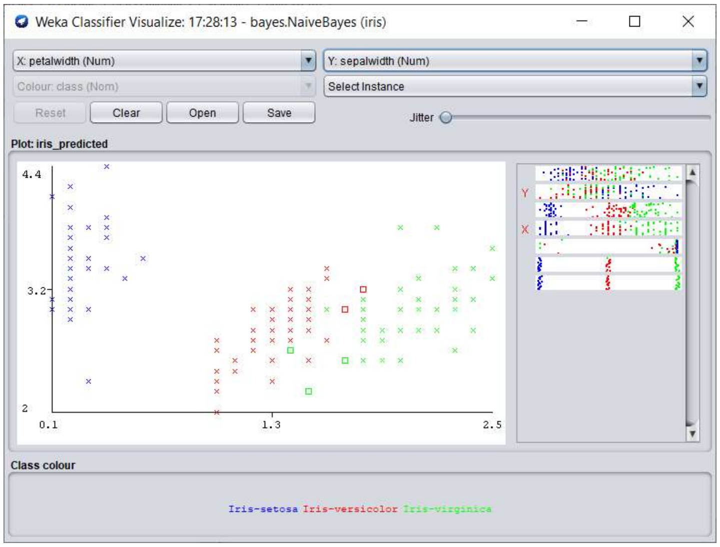
```
Naive Bayes Classifier

                          Class
Attribute           Iris-setosa Iris-versicolor  Iris-virginica
                         (0.33)          (0.33)          (0.33)
===============================================================
sepallength
  mean                   4.9913          5.9379          6.5795
  std. dev.               0.355          0.5042          0.6353
  weight sum                 50              50              50
  precision              0.1059          0.1059          0.1059

sepalwidth
  mean                   3.4015          2.7687          2.9629
  std. dev.              0.3925          0.3038          0.3088
  weight sum                 50              50              50
  precision              0.1091          0.1091          0.1091

petallength
  mean                   1.4694          4.2452          5.5516
  std. dev.              0.1782          0.4712          0.5529
  weight sum                 50              50              50
  precision              0.1405          0.1405          0.1405

petalwidth
  mean                   0.2743          1.3097          2.0343
  std. dev.              0.1096          0.1915          0.2646
  weight sum                 50              50              50
  precision              0.1143          0.1143          0.1143
```

OK

33

33

Naïve Bayes classifier visualization


Weka Classifier Visualize: 17:28:13 - bayes.NaiveBayes (iris)
Reset
Clear
Open
Save
Jitter
Plot: iris_predicted
4.4
3.2
2
0.1
1.3
2.5
Y
X
Class colour
Iris-setosa Iris-versicolor Iris-virginica

X: petalwidth (Num)

Y: sepalwidth (Num)

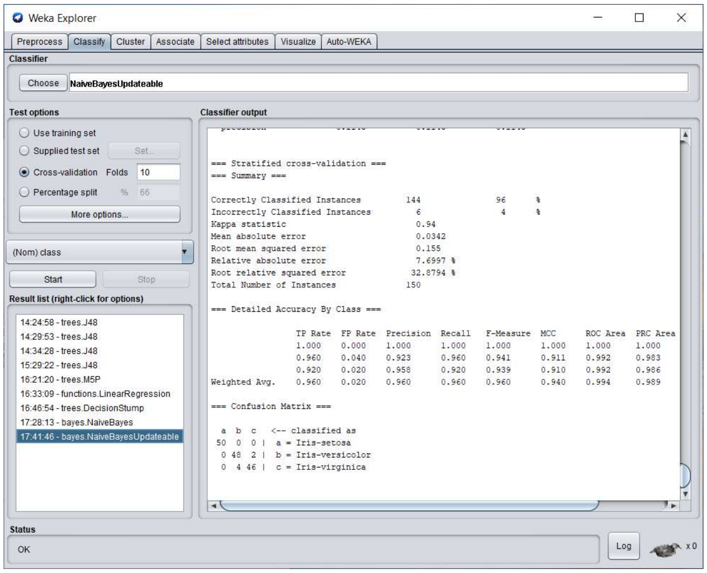
Colour: class (Nom)

Select Instance

34

34

Naïve Bayes updatable – process one instance at a time



Classifier

Choose NaiveBayesUpdateable

Test options

○ Use training set

○ Supplied test set   Set...

● Cross-validation   Folds 10

○ Percentage split   % 66

(Nom) class

Result list (right-click for options)

- 14:24:58 - trees.J48
- 14:29:53 - trees.J48
- 14:34:28 - trees.J48
- 15:29:22 - trees.J48
- 16:21:20 - trees.M5P
- 16:33:09 - functions.LinearRegression
- 16:46:54 - trees.DecisionStump
- 17:28:13 - bayes.NaiveBayes
- 17:41:46 - bayes.NaiveBayesUpdateable

Classifier output

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

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.0342
Root mean squared error                  0.155
Relative absolute error                  7.6997 %
Root relative squared error             32.8794 %
Total Number of Instances              150

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000
                 0.960    0.040    0.923      0.960    0.941      0.911    0.992     0.983
                 0.920    0.020    0.958      0.920    0.939      0.910    0.992     0.986
Weighted Avg.    0.960    0.020    0.960      0.960    0.960      0.940    0.994     0.989

=== Confusion Matrix ===

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

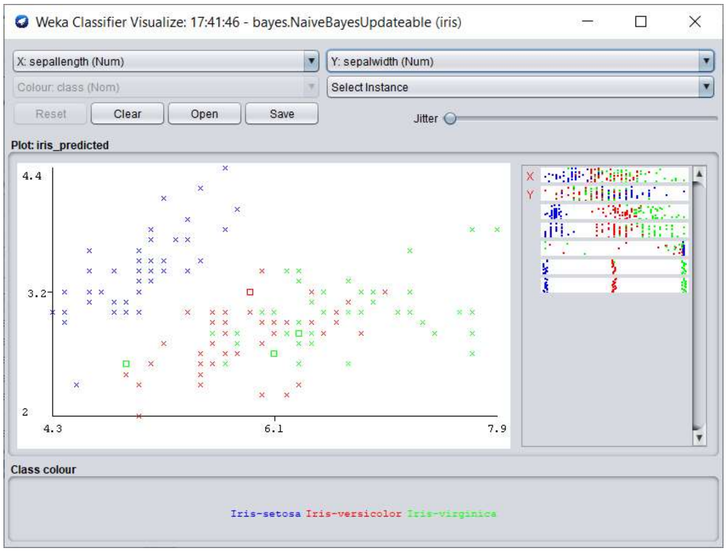
Status

OK

35

35

Naïve Bayes updatable visualization



Weka Classifier Visualize: 17:41:46 - bayes.NaiveBayesUpdateable (iris)

X: sepallength (Num)

Y: sepalwidth (Num)

Colour: class (Nom)

Select Instance

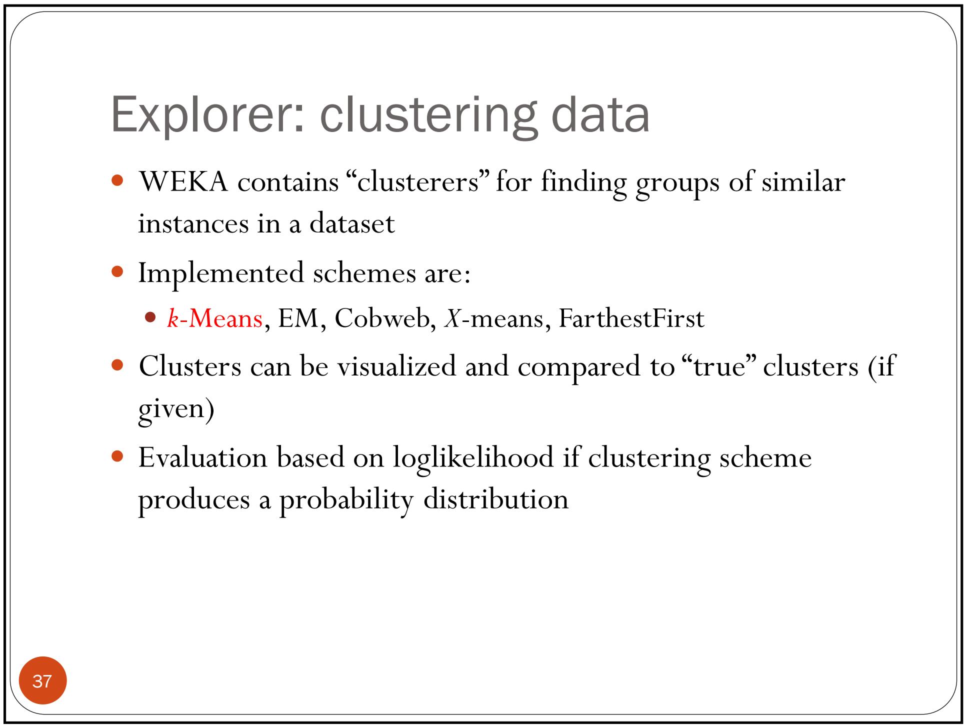
Plot: iris_predicted

Class colour

Iris-setosa Iris-versicolor Iris-virginica

36

36

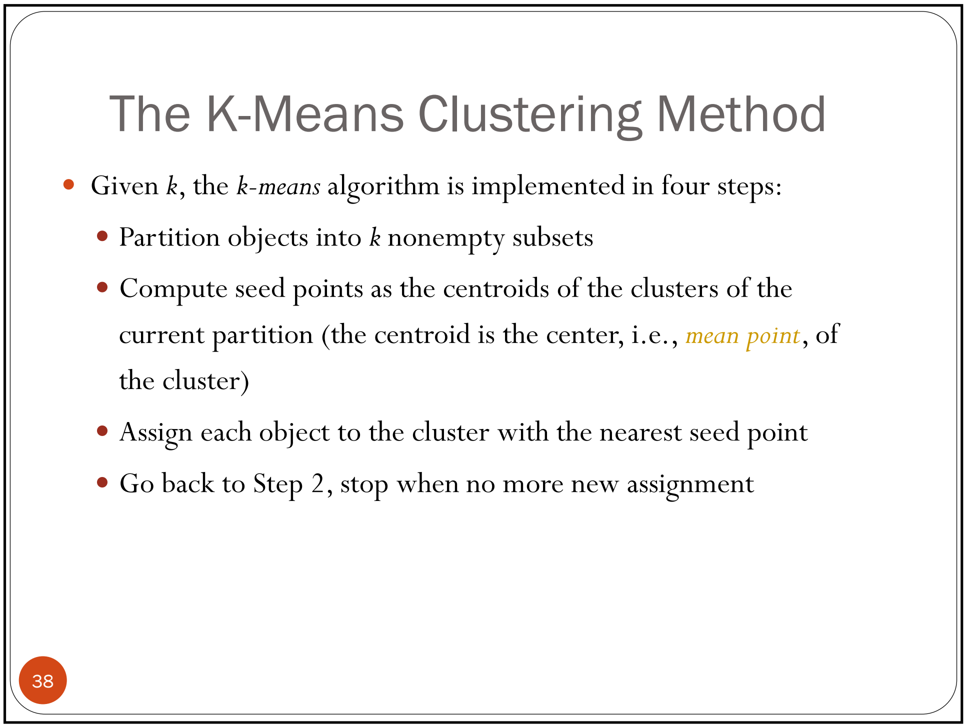


Explorer: clustering data

- WEKA contains "clusterers" for finding groups of similar instances in a dataset
- Implemented schemes are:
  - *k*-Means, EM, Cobweb, *X*-means, FarthestFirst
- Clusters can be visualized and compared to "true" clusters (if given)
- Evaluation based on loglikelihood if clustering scheme produces a probability distribution

37

37



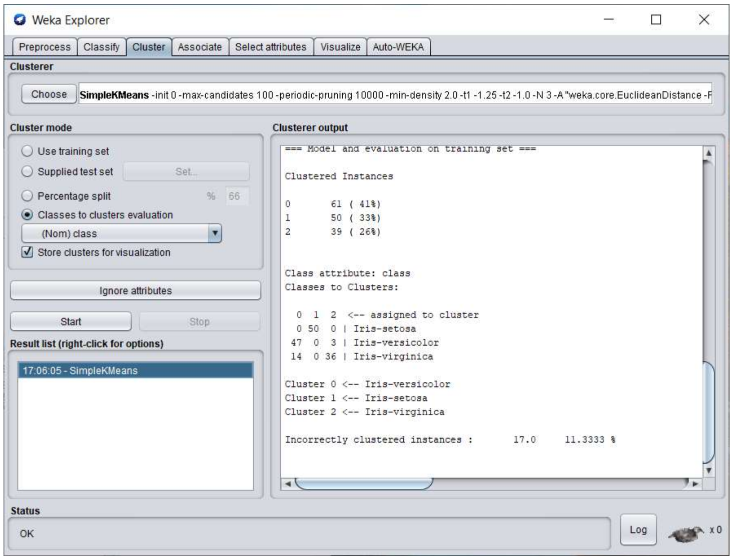
The K-Means Clustering Method

- Given *k*, the *k-means* algorithm is implemented in four steps:
  - Partition objects into *k* nonempty subsets
  - Compute seed points as the centroids of the clusters of the current partition (the centroid is the center, i.e., *mean point*, of the cluster)
  - Assign each object to the cluster with the nearest seed point
  - Go back to Step 2, stop when no more new assignment

38

38

Clustering – open iris.arff, select Simple Kmeans (numClusters = 3)



Weka Explorer

Clusterer

Choose 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 -F

Cluster mode

○ Use training set
○ Supplied test set Set...
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) class
☑ Store clusters for visualization

Result list (right-click for options)

17:06:05 - SimpleKMeans

Clusterer output

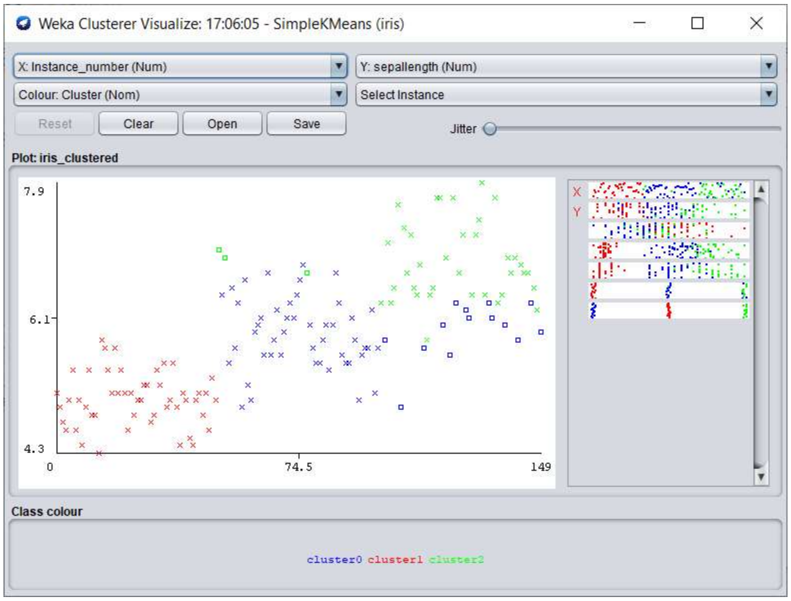
```
=== Model and evaluation on training set ===

Clustered Instances

0       61 ( 41%)
1       50 ( 33%)
2       39 ( 26%)


Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 47  0  3 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :      17.0     11.3333 %
```

Status

OK

39

K-means clustering visualization

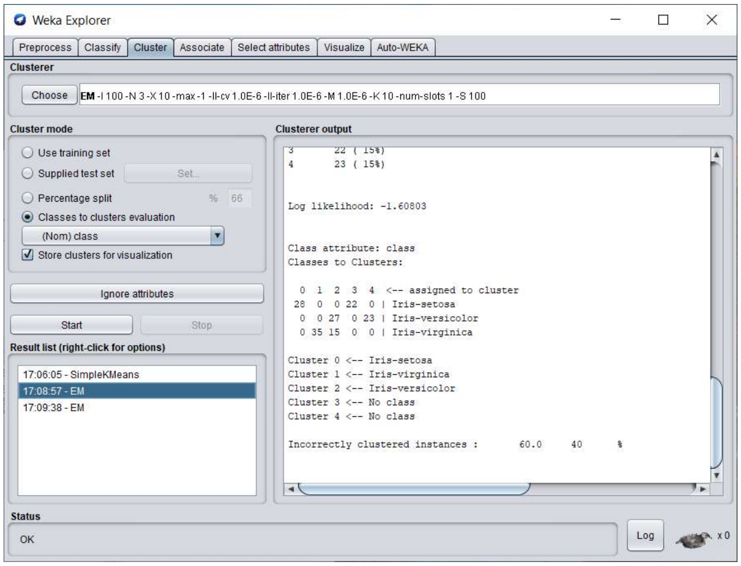


Weka Clusterer Visualize: 17:06:05 - SimpleKMeans (iris)

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

40

EM (Expectation-Maximization) clustering – out of the box



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

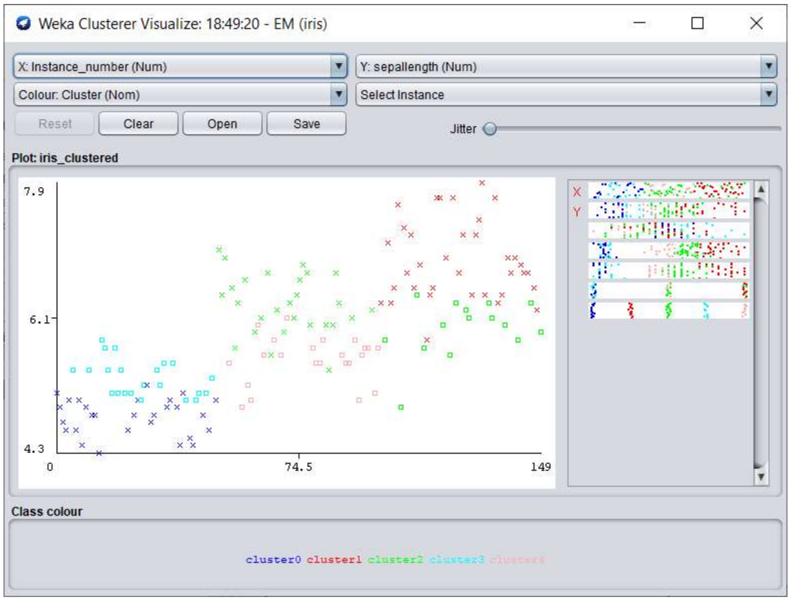
○ Use training set
○ Supplied test set
○ Percentage split % 66
● Classes to clusters evaluation
(Nom) class
☑ Store clusters for visualization

17:06:05 - SimpleKMeans
17:08:57 - EM
17:09:38 - EM

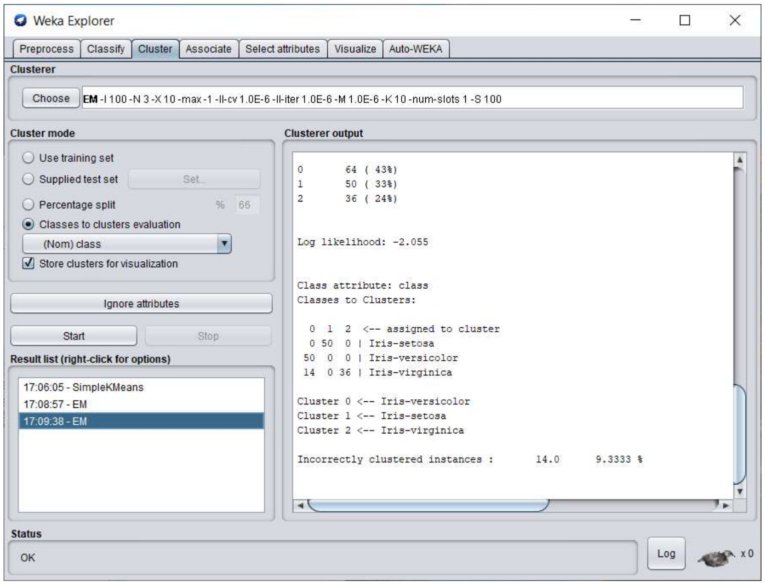
```
3       22 ( 15%)
4       23 ( 15%)


Log likelihood: -1.60803


Class attribute: class
Classes to Clusters:

  0  1  2  3  4  <-- assigned to cluster
 28  0  0 22  0 | Iris-setosa
  0  0 27  0 23 | Iris-versicolor
  0 35 15  0  0 | Iris-virginica

Cluster 0 <-- Iris-setosa
Cluster 1 <-- Iris-virginica
Cluster 2 <-- Iris-versicolor
Cluster 3 <-- No class
Cluster 4 <-- No class

Incorrectly clustered instances :      60.0     40      %
```

OK

EM clustering (out of the box) visualization



EM clustering – numClusters = 3



Clusterer: 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

Cluster mode

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

Result list (right-click for options)

17:06:05 - SimpleKMeans
17:08:57 - EM
17:09:38 - EM

Clusterer output

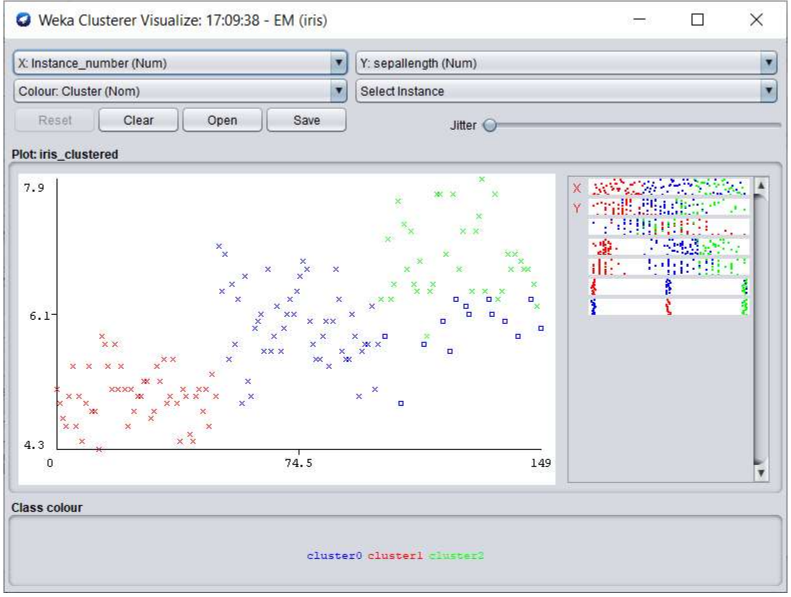
```
0       64 ( 43%)
1       50 ( 33%)
2       36 ( 24%)


Log likelihood: -2.055


Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 50  0  0 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :       14.0      9.3333 %
```

Status: OK

43

43

EM (numClusters = 3) visualization



X: Instance_number (Num)

Y: sepallength (Num)

Colour: Cluster (Nom)

Select Instance

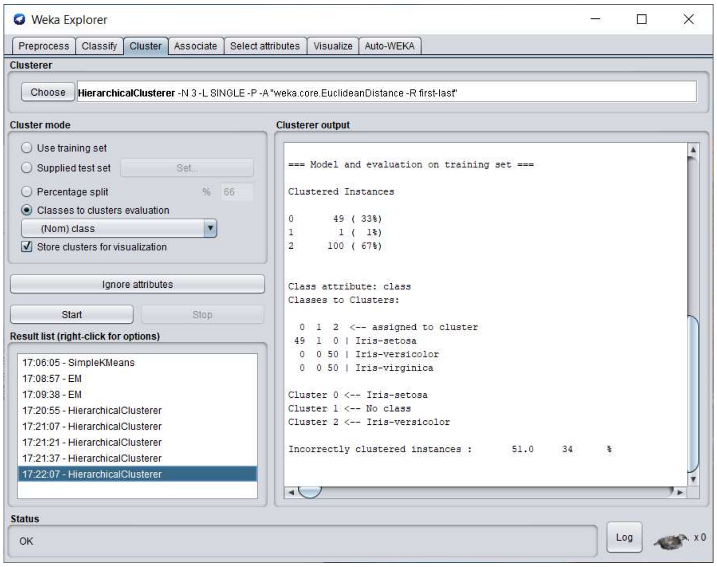
Plot: iris_clustered

Class colour: cluster0 cluster1 cluster2

44

44

Hierarchical clustering (numClusters = 3)



Clusterer: **HierarchicalClusterer** -N 3 -L SINGLE -P -A "weka.core.EuclideanDistance -R first-last"

Cluster mode:

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

Result list (right-click for options):

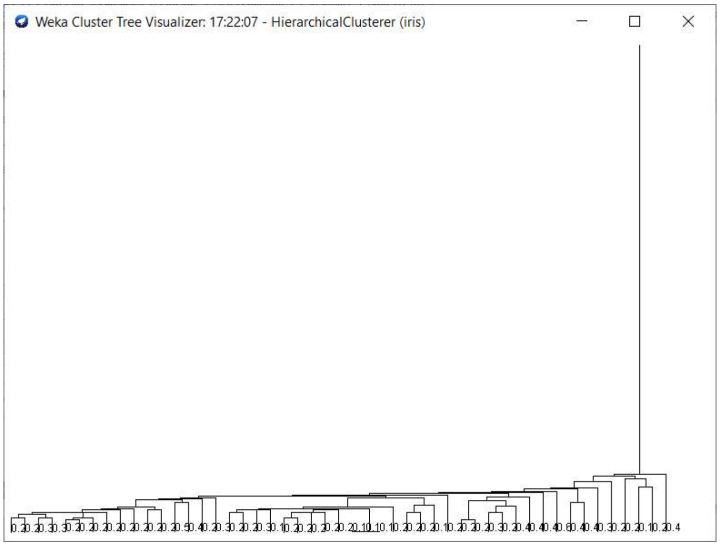
- 17:06:05 - SimpleKMeans
- 17:08:57 - EM
- 17:09:38 - EM
- 17:20:55 - HierarchicalClusterer
- 17:21:07 - HierarchicalClusterer
- 17:21:21 - HierarchicalClusterer
- 17:21:37 - HierarchicalClusterer
- 17:22:07 - HierarchicalClusterer

Clusterer output:

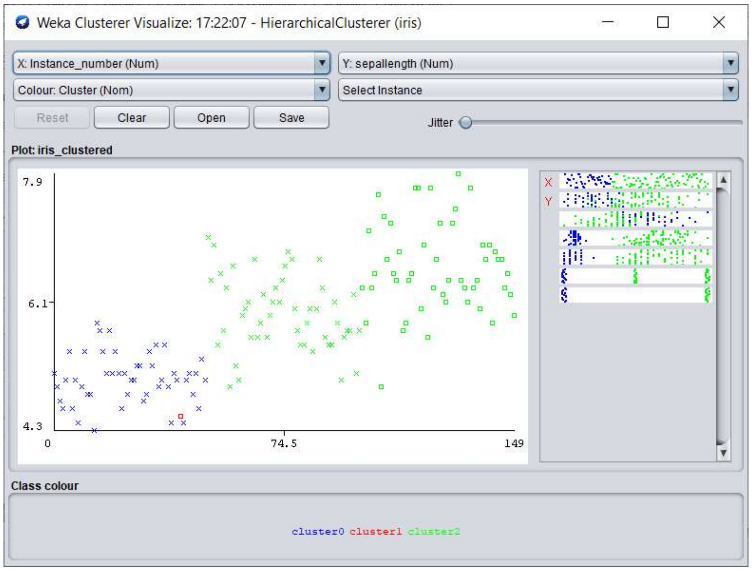
```
=== Model and evaluation on training set ===

Clustered Instances

0       49 ( 33%)
1        1 (  1%)
2      100 ( 67%)


Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
 49  1  0 | Iris-setosa
  0  0 50 | Iris-versicolor
  0  0 50 | Iris-virginica

Cluster 0 <-- Iris-setosa
Cluster 1 <-- No class
Cluster 2 <-- Iris-versicolor

Incorrectly clustered instances :      51.0     34      %
```

Status: OK

Hierarchical clustering – tree visualization



Hierarchical clustering - visualization



Weka Clusterer Visualize: 17:22:07 - HierarchicalClusterer (iris)

X: Instance_number (Num)

Y: sepallength (Num)

Colour: Cluster (Nom)

Plot: iris_clustered

47

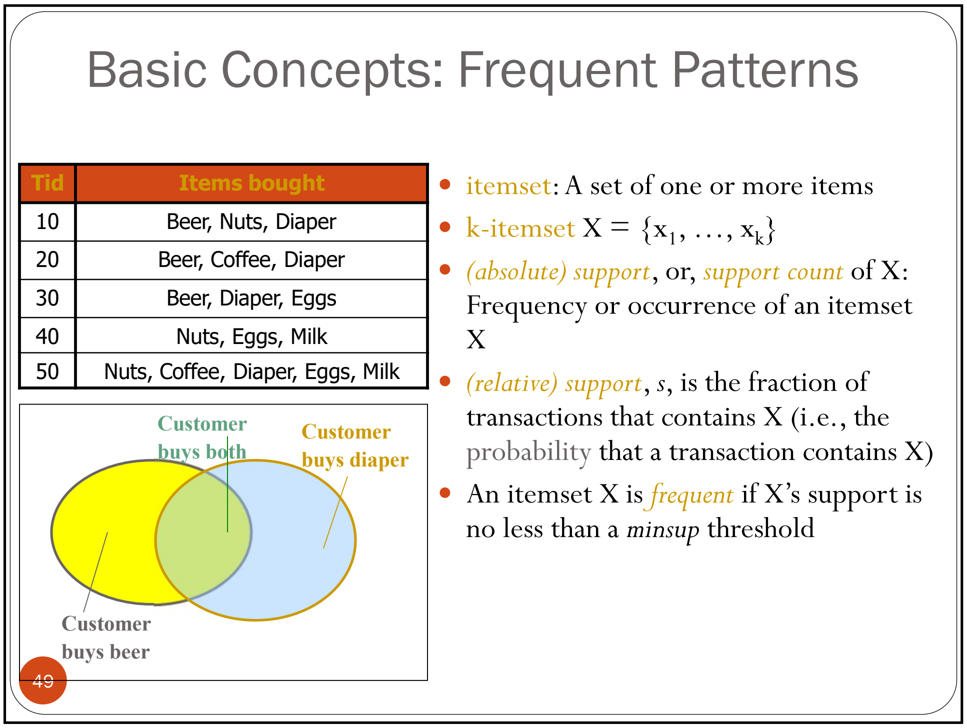
47

Explorer: finding associations

- WEKA contains an implementation of the Apriori algorithm for learning association rules
  - Works only with discrete data
- Can identify statistical dependencies between groups of attributes:
  - milk, butter ⇒ bread, eggs (with confidence 0.9 and support 2000)
- Apriori can compute all rules that have a given minimum support and exceed a given confidence

48

48



Basic Concepts: Frequent Patterns

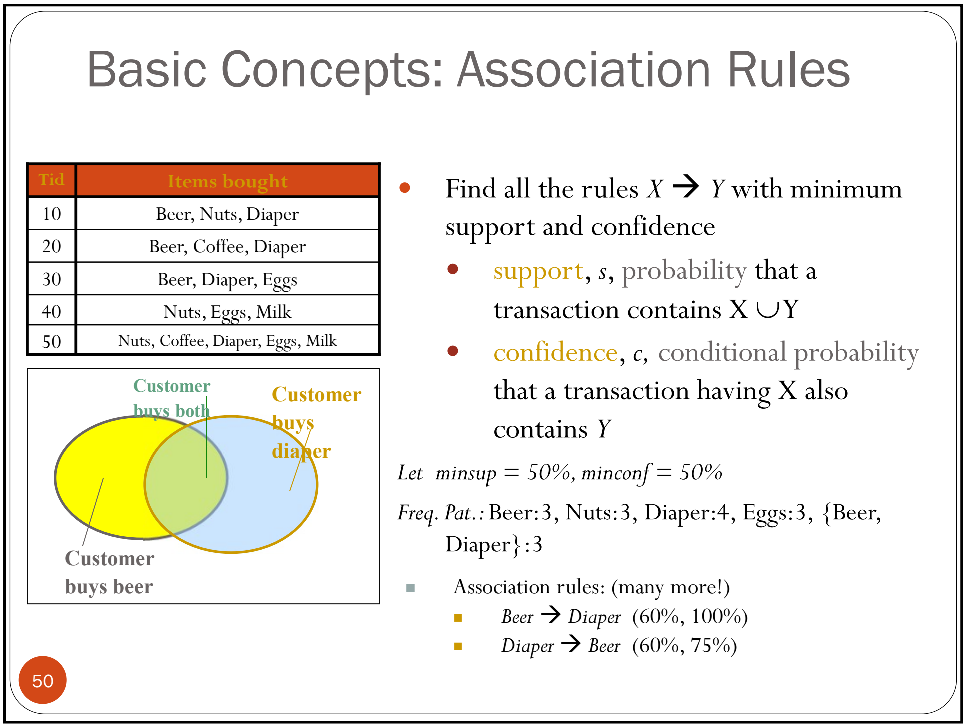
Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

Customer buys both
Customer buys diaper
Customer buys beer

- itemset: A set of one or more items
- k-itemset X = $\{x_1, \ldots, x_k\}$
- *(absolute) support*, or, *support count* of X: Frequency or occurrence of an itemset X
- *(relative) support*, *s*, is the fraction of transactions that contains X (i.e., the probability that a transaction contains X)
- An itemset X is *frequent* if X's support is no less than a *minsup* threshold

49

49



Basic Concepts: Association Rules

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

Customer buys both
Customer buys diaper
Customer buys beer

- Find all the rules *X* → *Y* with minimum support and confidence
  - support, *s*, probability that a transaction contains X ∪Y
  - confidence, *c,* conditional probability that a transaction having X also contains *Y*

Let minsup = 50%, minconf = 50%

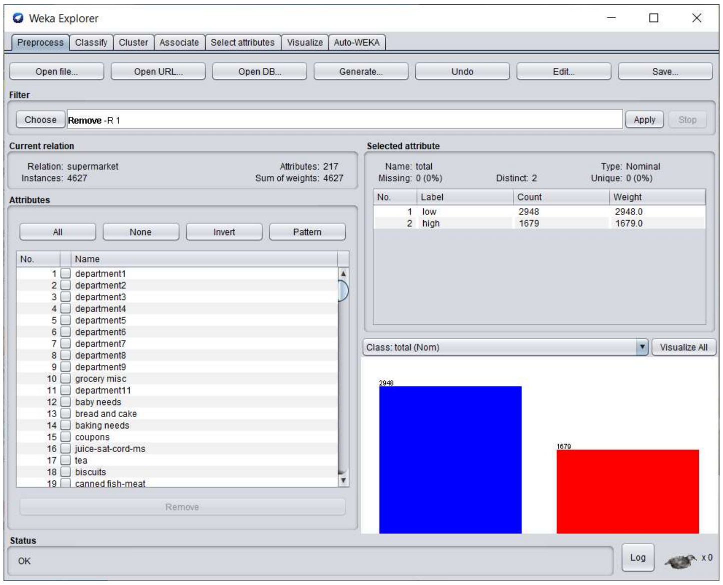
Freq. Pat.: Beer:3, Nuts:3, Diaper:4, Eggs:3, {Beer, Diaper}:3

- Association rules: (many more!)
  - *Beer* → *Diaper* (60%, 100%)
  - *Diaper* → *Beer* (60%, 75%)

50

50

Associative rules: open supermarket.arff



Filter

Remove -R 1

Current relation

Relation: supermarket
Instances: 4627
Attributes: 217
Sum of weights: 4627

Selected attribute

Name: total
Missing: 0 (0%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	low	2948	2948.0
2	high	1679	1679.0

Class: total (Nom)

Attributes

No.		Name
1	□	department1
2	□	department2
3	□	department3
4	□	department4
5	□	department5
6	□	department6
7	□	department7
8	□	department8
9	□	department9
10	□	grocery misc
11	□	department11
12	□	baby needs
13	□	bread and cake
14	□	baking needs
15	□	coupons
16	□	juice-sat-cord-ms
17	□	tea
18	□	biscuits
19	□	canned fish-meat

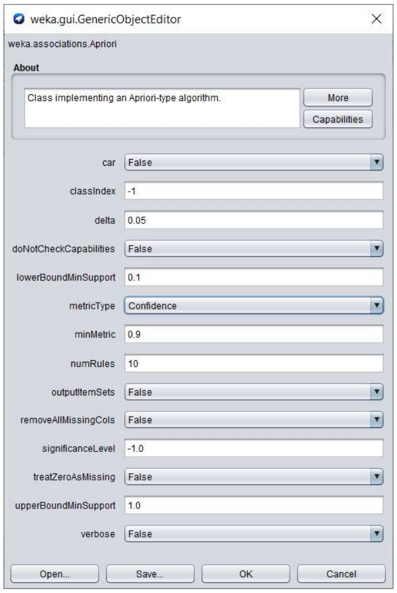
Status

OK

51

51

Associative rules: Apriori algorithm



weka.associations.Apriori

About

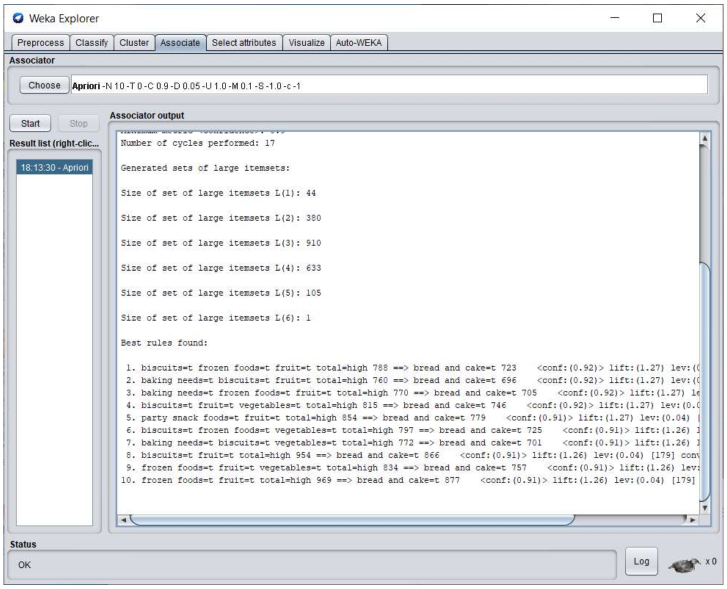
Class implementing an Apriori-type algorithm.

car	False
classIndex	-1
delta	0.05
doNotCheckCapabilities	False
lowerBoundMinSupport	0.1
metricType	Confidence
minMetric	0.9
numRules	10
outputItemSets	False
removeAllMissingCols	False
significanceLevel	-1.0
treatZeroAsMissing	False
upperBoundMinSupport	1.0
verbose	False

52

52

Associative rules – Apriori output



Associator

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

Result list (right-clic...

18:13:30 - Apriori

Associator output

```
Number of cycles performed: 17

Generated sets of large itemsets:

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

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

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

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

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

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

Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    <conf:(0.92)> lift:(1.27) lev:(
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    <conf:(0.92)> lift:(1.27) lev:(
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    <conf:(0.92)> lift:(1.27) le
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    <conf:(0.92)> lift:(1.27) lev:(0.
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    <conf:(0.91)> lift:(1.27) lev:(0.04)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    <conf:(0.91)> lift:(1.26) 
 7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    <conf:(0.91)> lift:(1.26) 
 8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv
 9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    <conf:(0.91)> lift:(1.26) lev:
10. frozen foods=t fruit=t total=high 969 ==> bread and cake=t 877    <conf:(0.91)> lift:(1.26) lev:(0.04) [179]
```

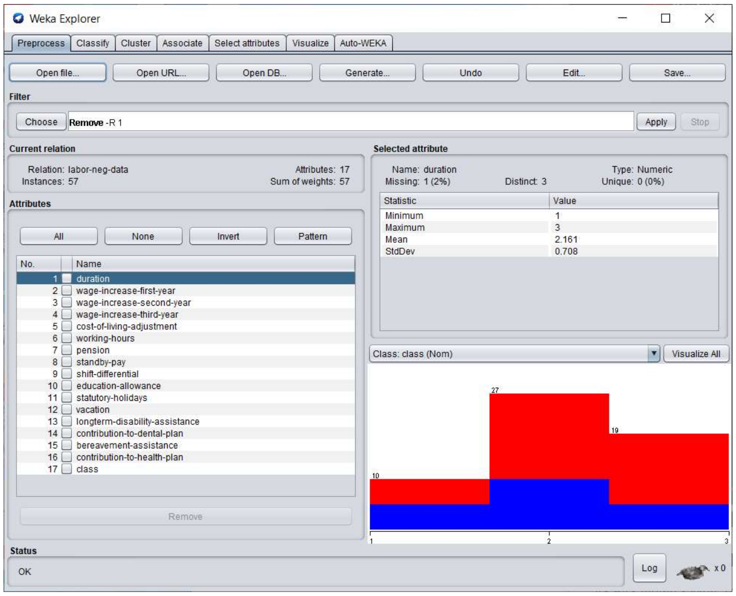
Status

OK

53

53

Weka – Automated feature selection – labor.arff



Filter

Remove -R 1

Current relation

Relation: labor-neg-data
Instances: 57
Attributes: 17
Sum of weights: 57

Attributes

No.		Name
1	□	duration
2	□	wage-increase-first-year
3	□	wage-increase-second-year
4	□	wage-increase-third-year
5	□	cost-of-living-adjustment
6	□	working-hours
7	□	pension
8	□	standby-pay
9	□	shift-differential
10	□	education-allowance
11	□	statutory-holidays
12	□	vacation
13	□	longterm-disability-assistance
14	□	contribution-to-dental-plan
15	□	bereavement-assistance
16	□	contribution-to-health-plan
17	□	class

Selected attribute

Name: duration
Type: Numeric
Missing: 1 (2%)
Distinct: 3
Unique: 0 (0%)

Statistic	Value
Minimum	1
Maximum	3
Mean	2.161
StdDev	0.708

Class: class (Nom)

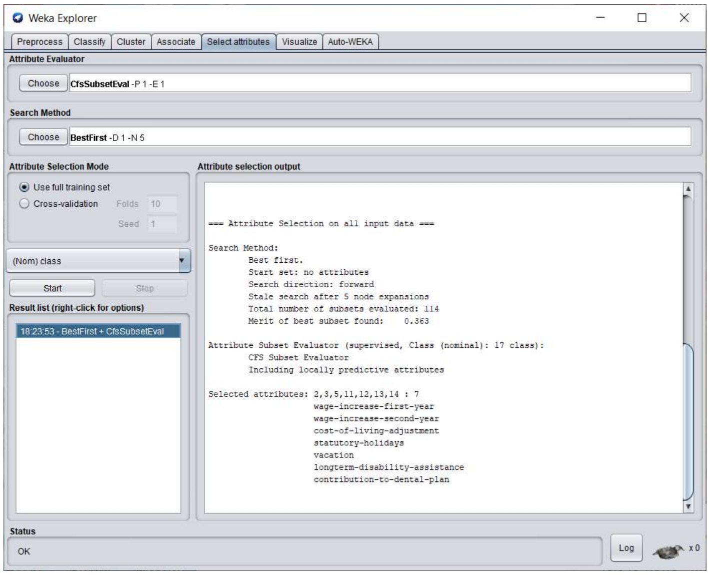
Status

OK

54

54

Weka – Automated feature selection Select attributes



Weka Explorer

Attribute Evaluator

Choose **CfsSubsetEval** -P 1 -E 1

Search Method

Choose **BestFirst** -D 1 -N 5

Attribute Selection Mode

● Use full training set

○ Cross-validation Folds 10 Seed 1

(Nom) class

Result list (right-click for options)

18:23:53 - BestFirst + CfsSubsetEval

Attribute selection output

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

Search Method:
        Best first.
        Start set: no attributes
        Search direction: forward
        Stale search after 5 node expansions
        Total number of subsets evaluated: 114
        Merit of best subset found:    0.363

Attribute Subset Evaluator (supervised, Class (nominal): 17 class):
        CFS Subset Evaluator
        Including locally predictive attributes

Selected attributes: 2,3,5,11,12,13,14 : 7
                     wage-increase-first-year
                     wage-increase-second-year
                     cost-of-living-adjustment
                     statutory-holidays
                     vacation
                     longterm-disability-assistance
                     contribution-to-dental-plan
```

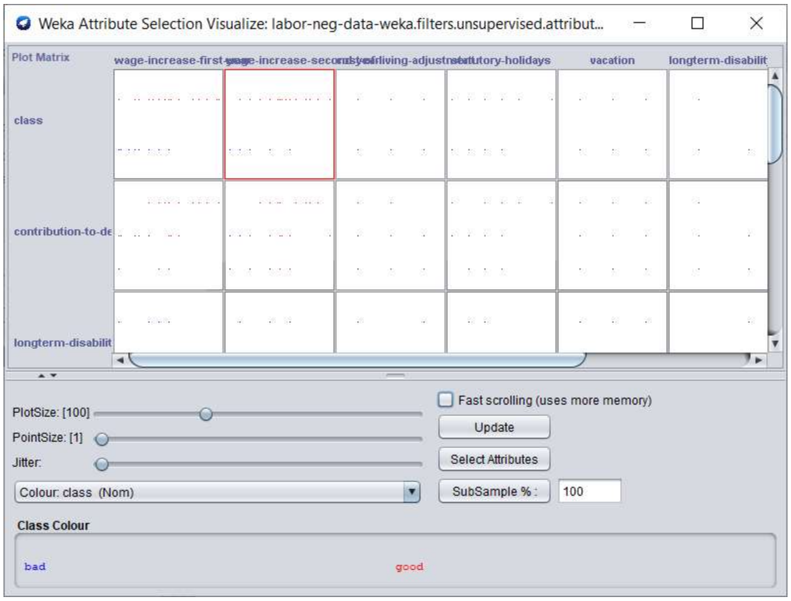
Status

OK

55

55

Attribute selection visualize



Weka Attribute Selection Visualize: labor-neg-data-weka.filters.unsupervised.attribut...

PlotSize: [100]

PointSize: [1]

Jitter:

Colour: class (Nom)

□ Fast scrolling (uses more memory)

SubSample % : 100

Class Colour

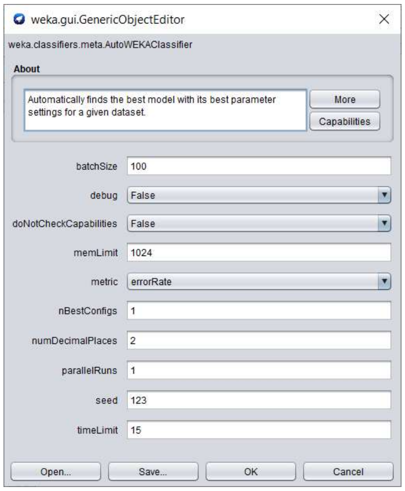
bad good

56

56

AutoWeka

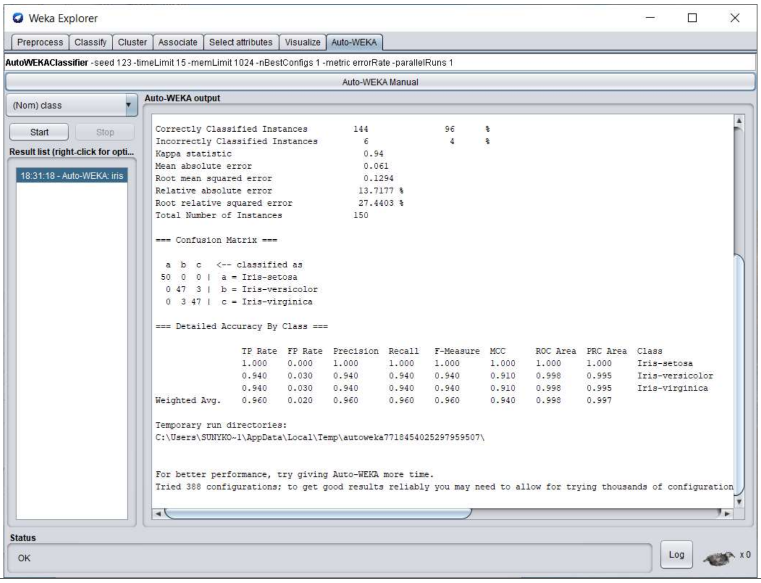
- Install AutoWeka plugin from Weka plugin manager
- Automatically searches through the joint space of WEKA's learning algorithms and their respective hyperparameter settings to maximize performance
- Load iris.arff
- Tune AutoWeka parameters
- Hit start


weka.gui.GenericObjectEditor
weka.classifiers.meta.AutoWEKAClassifier
About
More
Capabilities
Open...
Save...
OK
Cancel

Automatically finds the best model with its best parameter settings for a given dataset.

batchSize: 100
debug: False
doNotCheckCapabilities: False
memLimit: 1024
metric: errorRate
nBestConfigs: 1
numDecimalPlaces: 2
parallelRuns: 1
seed: 123
timeLimit: 15

AutoWeka result



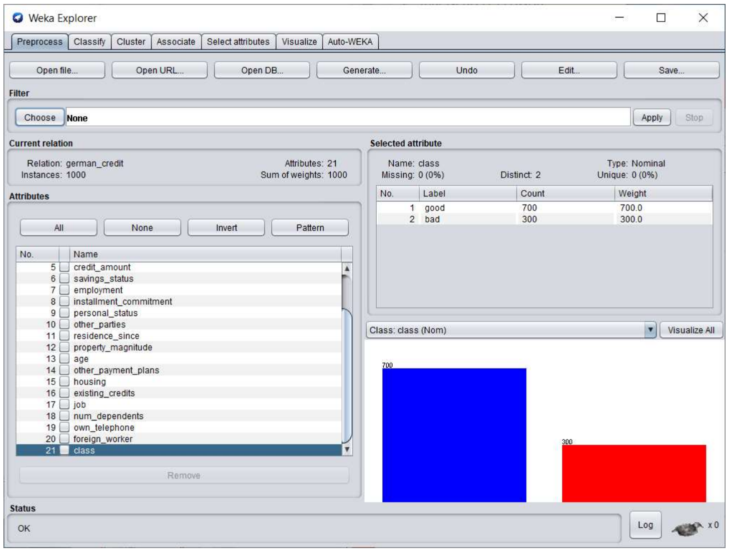
AutoWEKAClassifier -seed 123 -timeLimit 15 -memLimit 1024 -nBestConfigs 1 -metric errorRate -parallelRuns 1

```
Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.061
Root mean squared error                  0.1294
Relative absolute error                 13.7177 %
Root relative squared error             27.4403 %
Total Number of Instances              150

=== Confusion Matrix ===

  a  b  c   <-- classified as
 50  0  0 |  a = Iris-setosa
  0 47  3 |  b = Iris-versicolor
  0  3 47 |  c = Iris-virginica

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Iris-setosa
                 0.940    0.030    0.940      0.940    0.940      0.910    0.998     0.995     Iris-versicolor
                 0.940    0.030    0.940      0.940    0.940      0.910    0.998     0.995     Iris-virginica
Weighted Avg.    0.960    0.020    0.960      0.960    0.960      0.940    0.998     0.997

Temporary run directories:
C:\Users\SUNYKO~1\AppData\Local\Temp\autoweka7718454025297959507\


For better performance, try giving Auto-WEKA more time.
Tried 388 configurations; to get good results reliably you may need to allow for trying thousands of configuration
```

Imbalanced class problem

- Use Weka SMOTE filter
- Example of German Credit Data Analysis
- Two types of risks are associated with the bank's decision:
- If the applicant is a good credit risk, i.e. is likely to repay the loan, then not approving the loan to the person results in a loss of business to the bank
- If the applicant is a bad credit risk, i.e. is not likely to repay the loan, then approving the loan to the person results in a financial loss to the bank
- The German Credit data set is a publically available data set downloaded from the UCI Machine Learning Repository

Imbalanced class problem – German_credit dataset



Filter: None

Current relation

Relation: german_credit
Instances: 1000
Attributes: 21
Sum of weights: 1000

Attributes

No.	Name
5	credit_amount
6	savings_status
7	employment
8	installment_commitment
9	personal_status
10	other_parties
11	residence_since
12	property_magnitude
13	age
14	other_payment_plans
15	housing
16	existing_credits
17	job
18	num_dependents
19	own_telephone
20	foreign_worker
21	class

Selected attribute

Name: class
Missing: 0 (0%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)

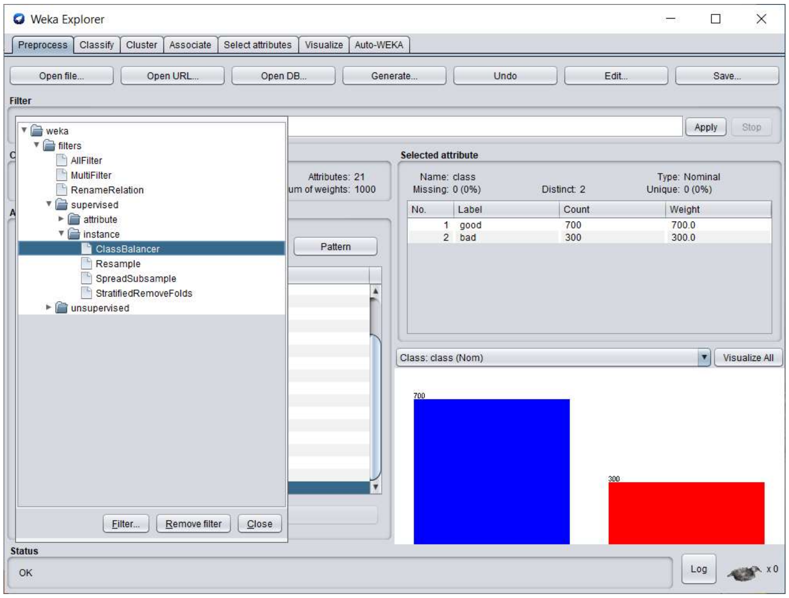
No.	Label	Count	Weight
1	good	700	700.0
2	bad	300	300.0

Class: class (Nom)

Status

OK

Use classbalancer from filter and apply



Attributes: 21
um of weights: 1000

Selected attribute

Name: class
Missing: 0 (0%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	good	700	700.0
2	bad	300	300.0

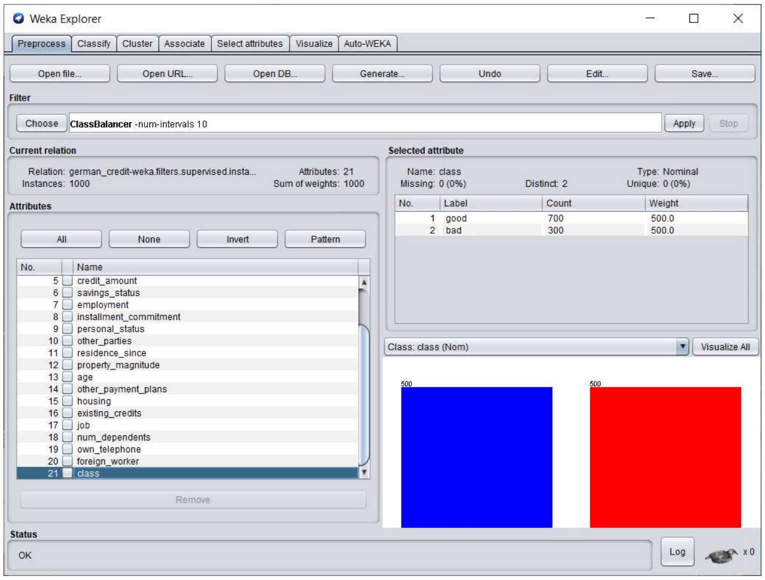
Class: class (Nom)

OK

61

61

Use classbalancer from filter and apply



ClassBalancer -num-intervals 10

Current relation

Relation: german_credit-weka.filters.supervised.insta...
Instances: 1000
Attributes: 21
Sum of weights: 1000

Selected attribute

Name: class
Missing: 0 (0%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	good	700	500.0
2	bad	300	500.0

Attributes

No.		Name
5	□	credit_amount
6	□	savings_status
7	□	employment
8	□	installment_commitment
9	□	personal_status
10	□	other_parties
11	□	residence_since
12	□	property_magnitude
13	□	age
14	□	other_payment_plans
15	□	housing
16	□	existing_credits
17	□	job
18	□	num_dependents
19	□	own_telephone
20	□	foreign_worker
21	□	class

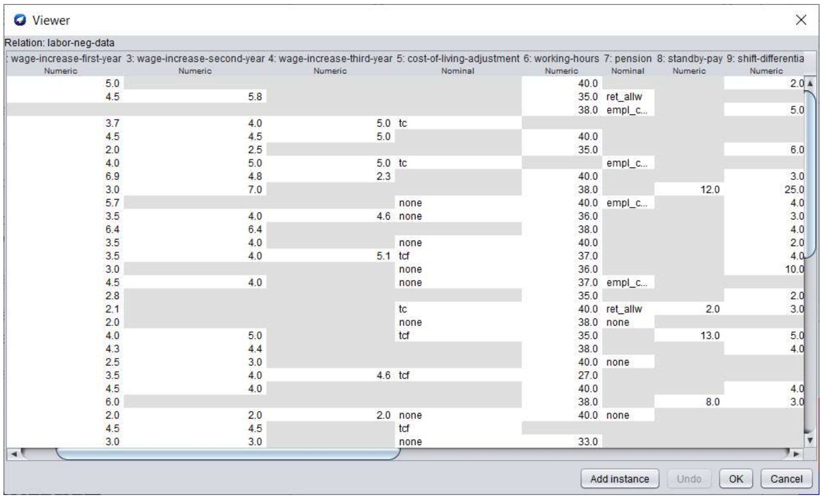
Class: class (Nom)

OK

62

62

Dealing with missing values – labor negotiation dataset



Viewer

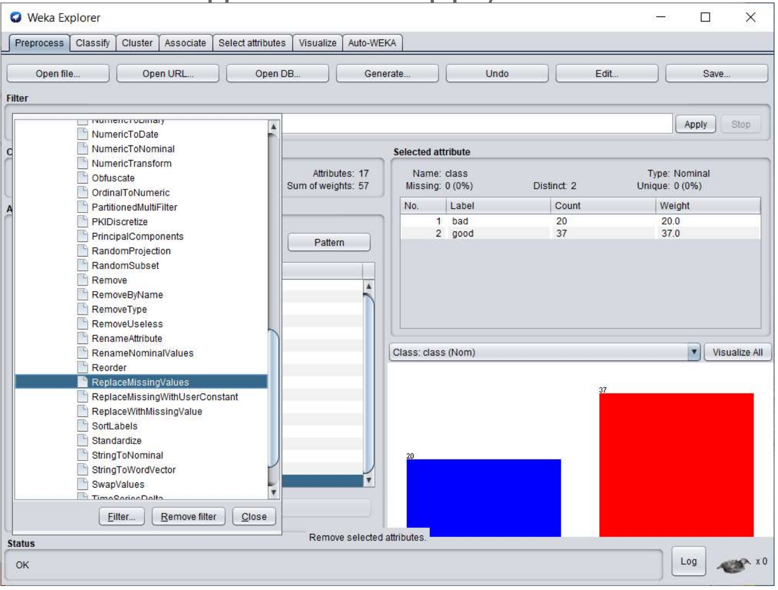
Relation: labor-neg-data

: wage-increase-first-year Numeric	3: wage-increase-second-year Numeric	4: wage-increase-third-year Numeric	5: cost-of-living-adjustment Nominal	6: working-hours Numeric	7: pension Nominal	8: standby-pay Numeric	9: shift-differentia Numeric
5.0				40.0			2.0
4.5	5.8			35.0	ret_allw		
				38.0	empl_c...		5.0
3.7	4.0	5.0	tc				
4.5	4.5	5.0		40.0			
2.0	2.5			35.0			6.0
4.0	5.0	5.0	tc		empl_c...		
6.9	4.8	2.3		40.0			3.0
3.0	7.0			38.0		12.0	25.0
5.7			none	40.0	empl_c...		4.0
3.5	4.0	4.6	none	36.0			3.0
6.4	6.4			38.0			4.0
3.5	4.0		none	40.0			2.0
3.5	4.0	5.1	tcf	37.0			4.0
3.0			none	36.0			10.0
4.5	4.0		none	37.0	empl_c...		
2.8				35.0			2.0
2.1			tc	40.0	ret_allw	2.0	3.0
2.0			none	38.0	none		
4.0	5.0		tcf	35.0		13.0	5.0
4.3	4.4			38.0			4.0
2.5	3.0			40.0	none		
3.5	4.0	4.6	tcf	27.0			
4.5	4.0			40.0			4.0
6.0				38.0		8.0	3.0
2.0	2.0	2.0	none	40.0	none		
4.5	4.5		tcf				
3.0	3.0		none	33.0			

63

63

Filters -> unsupervised -> attribute -> replaceMissingValues -> Apply



Attributes: 17
Sum of weights: 57

Selected attribute

Name: class
Missing: 0 (0%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	bad	20	20.0
2	good	37	37.0

Class: class (Nom)

Remove selected attributes.

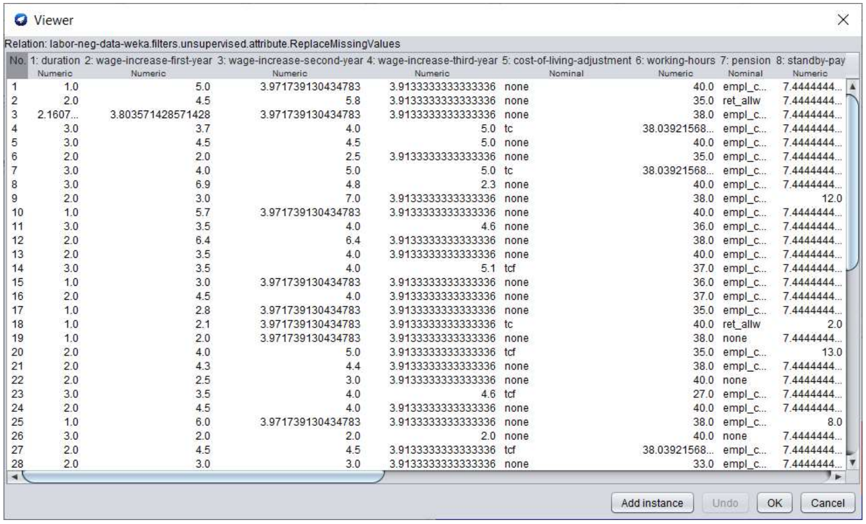
Status

OK

64

64

Replace numeric values with mean and nominal values with mode



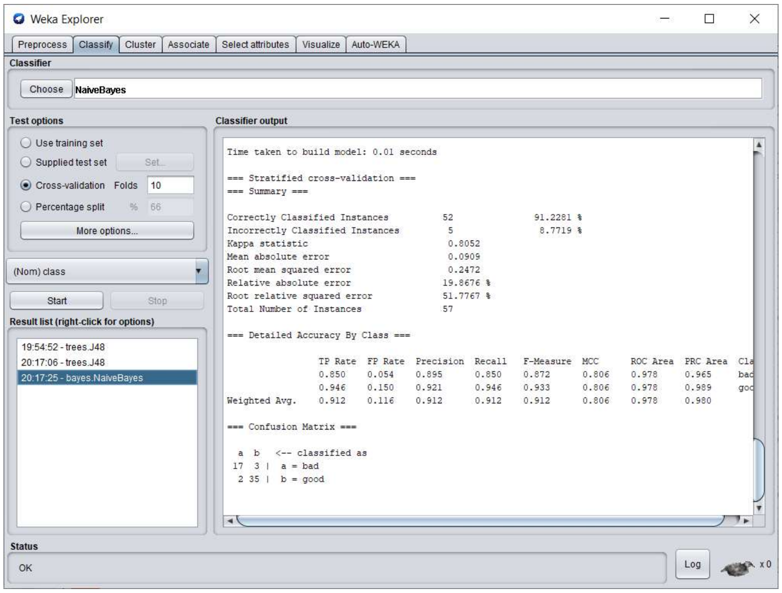
Viewer

Relation: labor-neg-data-weka.filters.unsupervised.attribute.ReplaceMissingValues

No.	1: duration	2: wage-increase-first-year	3: wage-increase-second-year	4: wage-increase-third-year	5: cost-of-living-adjustment	6: working-hours	7: pension	8: standby-pay
	Numeric	Numeric	Numeric	Numeric	Nominal	Numeric	Nominal	Numeric
1	1.0	5.0	3.971739130434783	3.9133333333333336	none	40.0	empl_c...	7.4444444...
2	2.0	4.5	5.8	3.9133333333333336	none	35.0	ret_allw	7.4444444...
3	2.1607...	3.803571428571428	3.971739130434783	3.9133333333333336	none	38.0	empl_c...	7.4444444...
4	3.0	3.7	4.0	5.0	tc	38.03921568...	empl_c...	7.4444444...
5	3.0	4.5	4.5	5.0	none	40.0	empl_c...	7.4444444...
6	2.0	2.0	2.5	3.9133333333333336	none	35.0	empl_c...	7.4444444...
7	3.0	4.0	5.0	5.0	tc	38.03921568...	empl_c...	7.4444444...
8	3.0	6.9	4.8	2.3	none	40.0	empl_c...	7.4444444...
9	2.0	3.0	7.0	3.9133333333333336	none	38.0	empl_c...	12.0
10	1.0	5.7	3.971739130434783	3.9133333333333336	none	40.0	empl_c...	7.4444444...
11	3.0	3.5	4.0	4.6	none	36.0	empl_c...	7.4444444...
12	2.0	6.4	6.4	3.9133333333333336	none	38.0	empl_c...	7.4444444...
13	2.0	3.5	4.0	3.9133333333333336	none	40.0	empl_c...	7.4444444...
14	3.0	3.5	4.0	5.1	tcf	37.0	empl_c...	7.4444444...
15	1.0	3.0	3.971739130434783	3.9133333333333336	none	36.0	empl_c...	7.4444444...
16	2.0	4.5	4.0	3.9133333333333336	none	37.0	empl_c...	7.4444444...
17	1.0	2.8	3.971739130434783	3.9133333333333336	none	35.0	empl_c...	7.4444444...
18	1.0	2.1	3.971739130434783	3.9133333333333336	tc	40.0	ret_allw	2.0
19	1.0	2.0	3.971739130434783	3.9133333333333336	none	38.0	none	7.4444444...
20	2.0	4.0	5.0	3.9133333333333336	tcf	35.0	empl_c...	13.0
21	2.0	4.3	4.4	3.9133333333333336	none	38.0	empl_c...	7.4444444...
22	2.0	2.5	3.0	3.9133333333333336	none	40.0	none	7.4444444...
23	3.0	3.5	4.0	4.6	tcf	27.0	empl_c...	7.4444444...
24	2.0	4.5	4.0	3.9133333333333336	none	40.0	empl_c...	7.4444444...
25	1.0	6.0	3.971739130434783	3.9133333333333336	none	38.0	empl_c...	8.0
26	3.0	2.0	2.0	2.0	none	40.0	none	7.4444444...
27	2.0	4.5	4.5	3.9133333333333336	tcf	38.03921568...	empl_c...	7.4444444...
28	2.0	3.0	3.0	3.9133333333333336	none	33.0	empl_c...	7.4444444...

65

Try Naïve Bayes on modified data



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)

19:54:52 - trees.J48
20:17:06 - trees.J48
20:17:25 - bayes.NaiveBayes

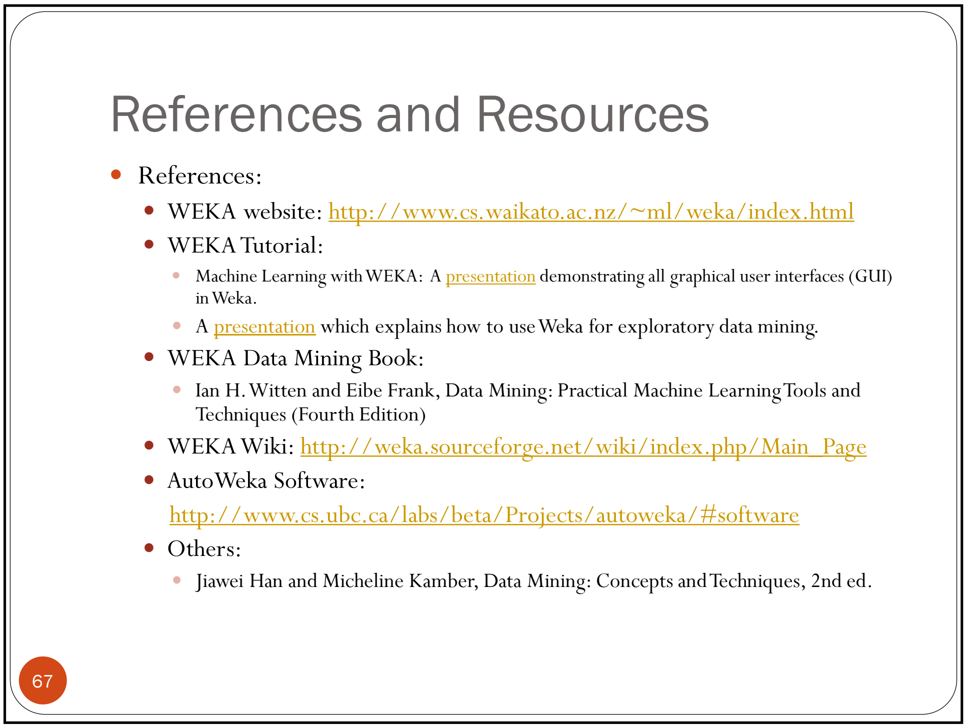
Classifier output

```
Time taken to build model: 0.01 seconds

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

Correctly Classified Instances          52               91.2281 %
Incorrectly Classified Instances         5                8.7719 %
Kappa statistic                          0.8052
Mean absolute error                      0.0909
Root mean squared error                  0.2472
Relative absolute error                 19.8676 %
Root relative squared error             51.7767 %
Total Number of Instances               57

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Cla
                 0.850    0.054    0.895      0.850    0.872      0.806    0.978     0.965     bac
                 0.946    0.150    0.921      0.946    0.933      0.806    0.978     0.989     goc
Weighted Avg.    0.912    0.116    0.912      0.912    0.912      0.806    0.978     0.980

=== Confusion Matrix ===

  a  b   <-- classified as
 17  3 |  a = bad
  2 35 |  b = good
```

Status

OK

66



References and Resources

- References:
  - WEKA website: http://www.cs.waikato.ac.nz/~ml/weka/index.html
  - WEKA Tutorial:
    - Machine Learning with WEKA: A presentation demonstrating all graphical user interfaces (GUI) in Weka.
    - A presentation which explains how to use Weka for exploratory data mining.
  - WEKA Data Mining Book:
    - Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques (Fourth Edition)
  - WEKA Wiki: http://weka.sourceforge.net/wiki/index.php/Main_Page
  - AutoWeka Software:

    http://www.cs.ubc.ca/labs/beta/Projects/autoweka/#software
  - Others:
    - Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, 2nd ed.