Muhammad Johar Shah

ID: 14124 SE-07

Data Mining

Submitted To: Sir Zain Shukat

Q1:

Ans:

Classification	Regression
• The discovery of model or functions where the mapping of objects is done into predefined classes.	• A devised model in which the mapping of objects is done into values.
• Discrete values	• Continuous values
• Decision tree, logistic regression, etc.	• Regression tree (Random forest), Linear regression, etc.
• Unordered	• Ordered
• Measuring accuracy	• Measurement of root mean square error

Key Differences Between Classification and Regression

1. The Classification process models a function through which the data is predicted in discrete class labels. On the other hand, regression is the process of creating a model which predict continuous quantity.
2. The classification algorithms involve decision tree, logistic regression, etc. In contrast, regression tree (e.g. Random forest) and linear regression are the examples of regression algorithms.
3. Classification predicts unordered data while regression predicts ordered data.
4. Regression can be evaluated using root mean square error. On the contrary, classification is evaluated by measuring accuracy.

Classification example:

Suppose from your past data (train data) you come to know that your best friend likes the above movies. Now one new movie (test data) released. Hopefully, you want to know your best friend like it or not. If you strongly confirmed about the chances of your friend like the move. You can take your friend to a movie this weekend.

If you clearly observe the problem it is just whether your friend like or not. Finding a solution to this type of problem is called as classification. This is because we are classifying the things to their belongings (yes or no, like or dislike). Keep in mind here we are forecasting target class (classification) and the other thing this classification belongs to supervised learning. This is because you are learning this from your train data.

In this case, the problem is a binary classification in which we have to predict whether output belongs to class 1 or class 2 (class 1 : yes, class 2: no). As we have discussed earlier we can use classification for predicting more classes too. Like (color prediction: red,green,blue,yellow,orange)

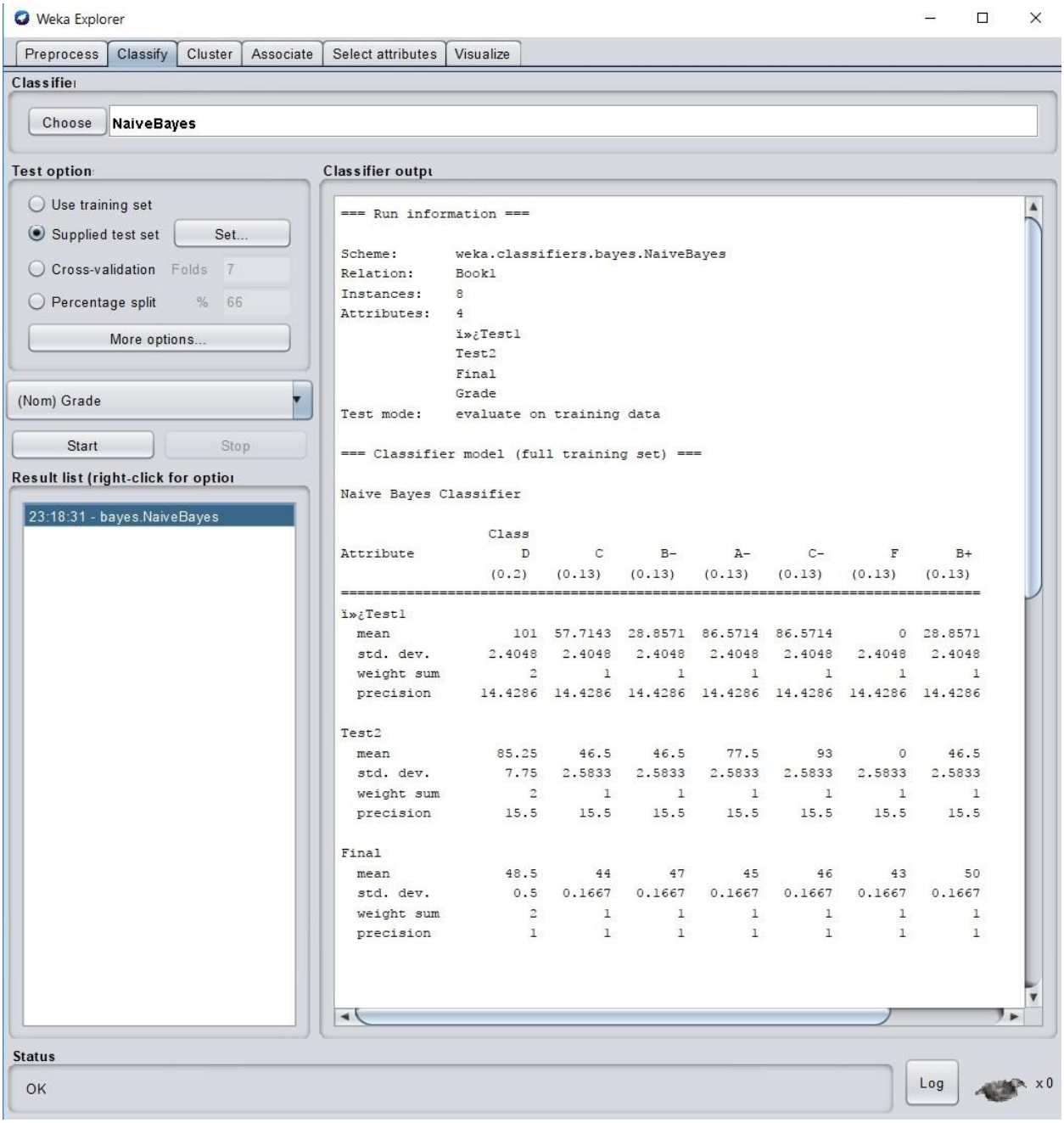
Regression example:

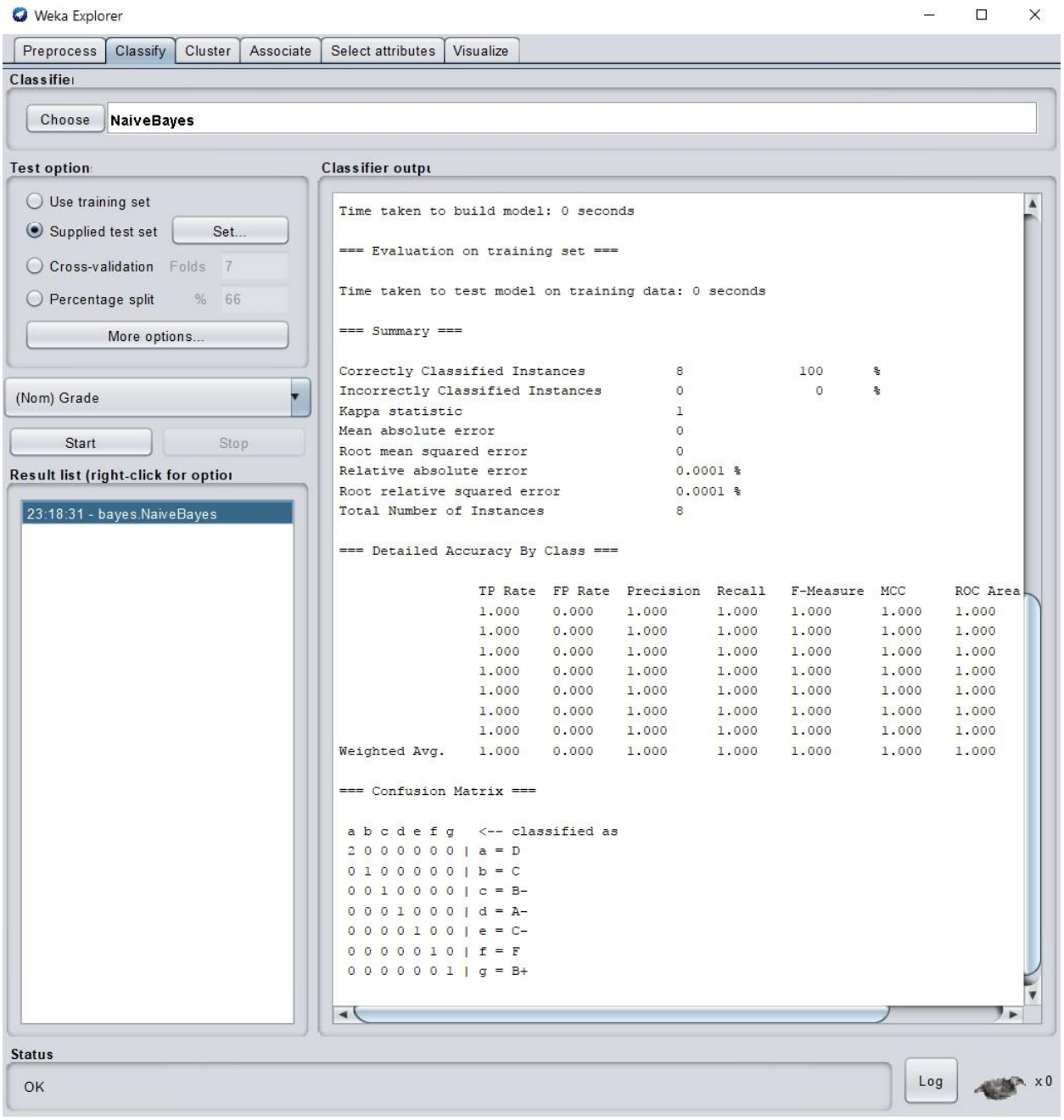
Suppose from your past data (train data) you come to know that your best friend likes the above movies. You also know how many times each particular movie seen by your friend. Now one new movie (test data) released. Now you are going to find how many times this newly released movie will your friend watch. It could be 5 times, 6 times,10 times etc…

If you clearly observe the problem is about finding the count, sometimes we can say this as predicting the value. Keep in mind, here we are forecasting a value (prediction) and the other thing this prediction also belongs to supervised learning. This is because you are learning this from you train data.

Q2:

Ans:

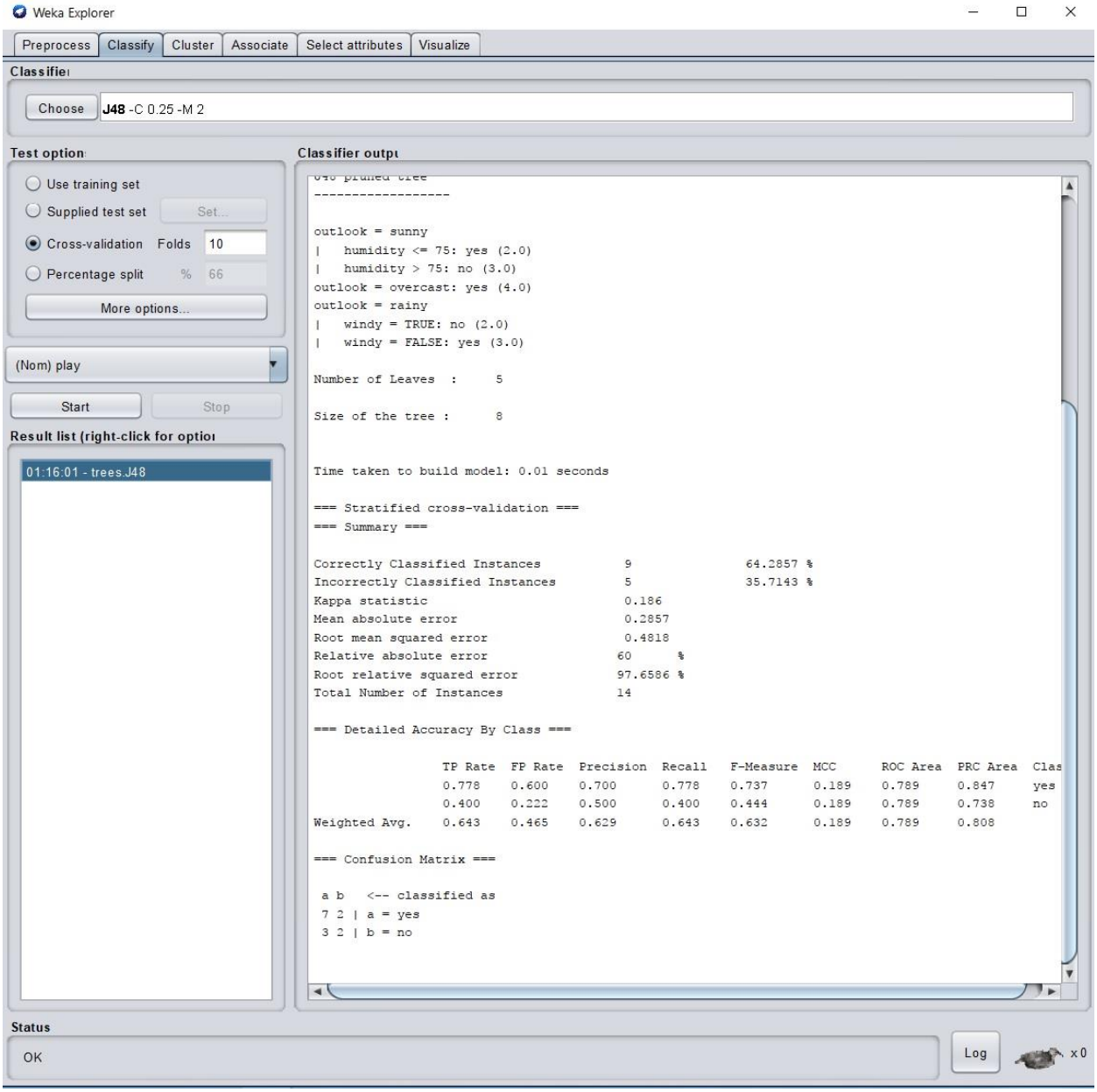

Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
NaiveBayes
Test option
Use training set
Supplied test set
Set...
Cross-validation
Folds
7
Percentage split
%
66
More options...
(Nom) Grade
Start
Stop
Result list (right-click for optio
23:18:31 - bayes.NaiveBayes
Classifier outpu
=== Run information ===
Scheme: weka.classifiers.bayes.NaiveBayes
Relation: Book1
Instances: 8
Attributes: 4
ï»¿Test1
Test2
Final
Grade
Test mode: evaluate on training data
=== Classifier model (full training set) ===
Naive Bayes Classifier
Class
Attribute D C B- A- C- F B+
(0.2) (0.13) (0.13) (0.13) (0.13) (0.13) (0.13)
ï»¿Test1
mean 101 57.7143 28.8571 86.5714 86.5714 0 28.8571
std. dev. 2.4048 2.4048 2.4048 2.4048 2.4048 2.4048 2.4048
weight sum 2 1 1 1 1 1 1
precision 14.4286 14.4286 14.4286 14.4286 14.4286 14.4286 14.4286
Test2
mean 85.25 46.5 46.5 77.5 93 0 46.5
std. dev. 7.75 2.5833 2.5833 2.5833 2.5833 2.5833 2.5833
weight sum 2 1 1 1 1 1 1
precision 15.5 15.5 15.5 15.5 15.5 15.5 15.5
Final
mean 48.5 44 47 45 46 43 50
std. dev. 0.5 0.1667 0.1667 0.1667 0.1667 0.1667 0.1667
weight sum 2 1 1 1 1 1 1
precision 1 1 1 1 1 1 1
Status
OK
Log
x 0

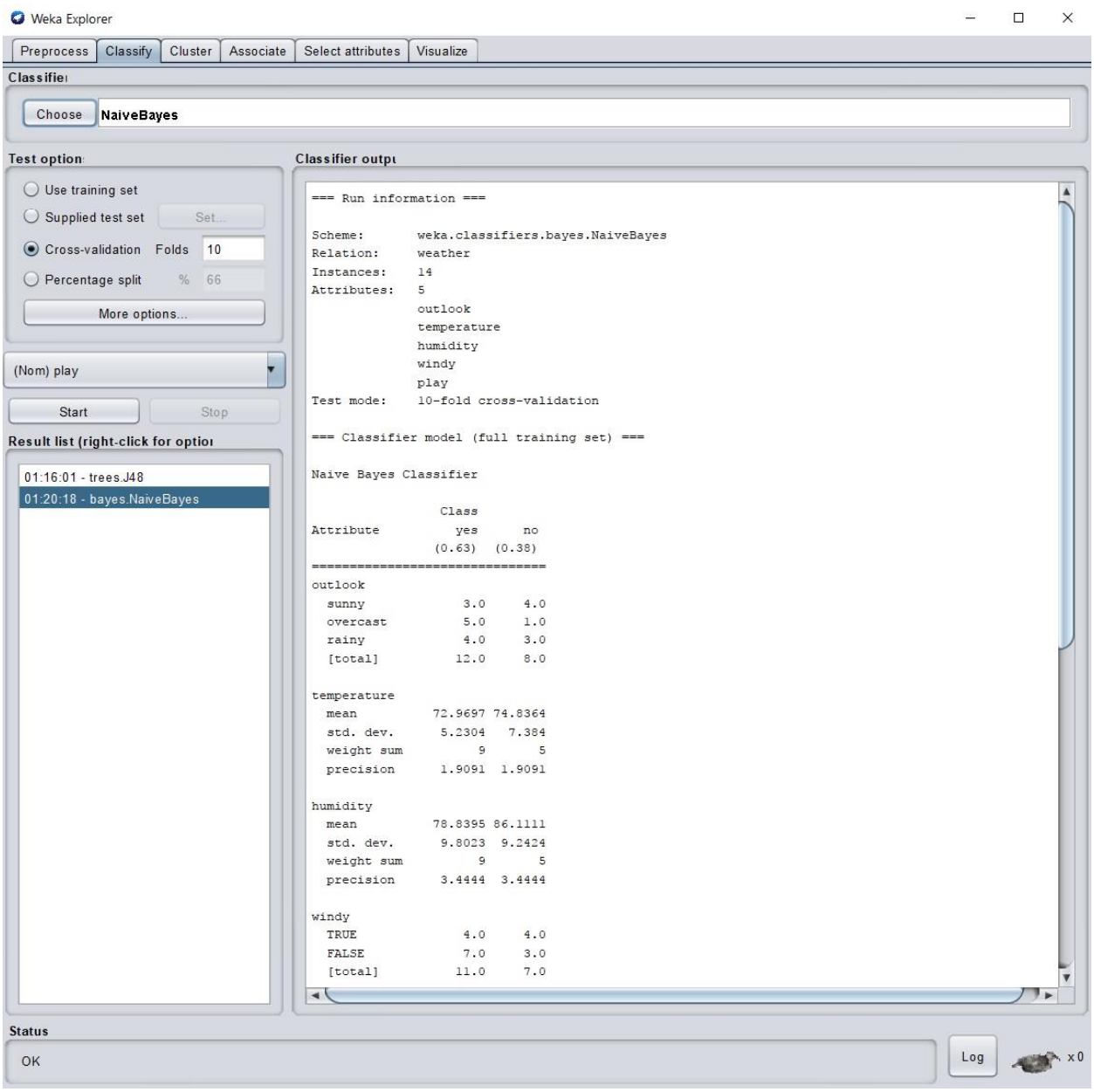

Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
NaiveBayes
Test option
Use training set
Supplied test set
Set...
Cross-validation
Folds
7
Percentage split
%
66
More options...
Classifier outpu
Time taken to build model: 0 seconds
=== Evaluation on training set ===
Time taken to test model on training data: 0 seconds
=== Summary ===
Correctly Classified Instances 8 100 %
Incorrectly Classified Instances 0 0 %
Kappa statistic 1
Mean absolute error 0
Root mean squared error 0
Relative absolute error 0.0001 %
Root relative squared error 0.0001 %
Total Number of Instances 8
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area
1.000 0.000 1.000 1.000 1.000 1.000 1.000
1.000 0.000 1.000 1.000 1.000 1.000 1.000
1.000 0.000 1.000 1.000 1.000 1.000 1.000
1.000 0.000 1.000 1.000 1.000 1.000 1.000
1.000 0.000 1.000 1.000 1.000 1.000 1.000
1.000 0.000 1.000 1.000 1.000 1.000 1.000
1.000 0.000 1.000 1.000 1.000 1.000 1.000
Weighted Avg. 1.000 0.000 1.000 1.000 1.000 1.000 1.000
=== Confusion Matrix ===
a b c d e f g <-- classified as
2 0 0 0 0 0 0 | a = D
0 1 0 0 0 0 0 | b = C
0 0 1 0 0 0 0 | c = B-
0 0 0 1 0 0 0 | d = A-
0 0 0 0 1 0 0 | e = C-
0 0 0 0 0 1 0 | f = F
0 0 0 0 0 0 1 | g = B+
(Nom) Grade
Start
Stop
Result list (right-click for optio
23:18:31 - bayes.NaiveBayes
Status
OK
Log
x 0

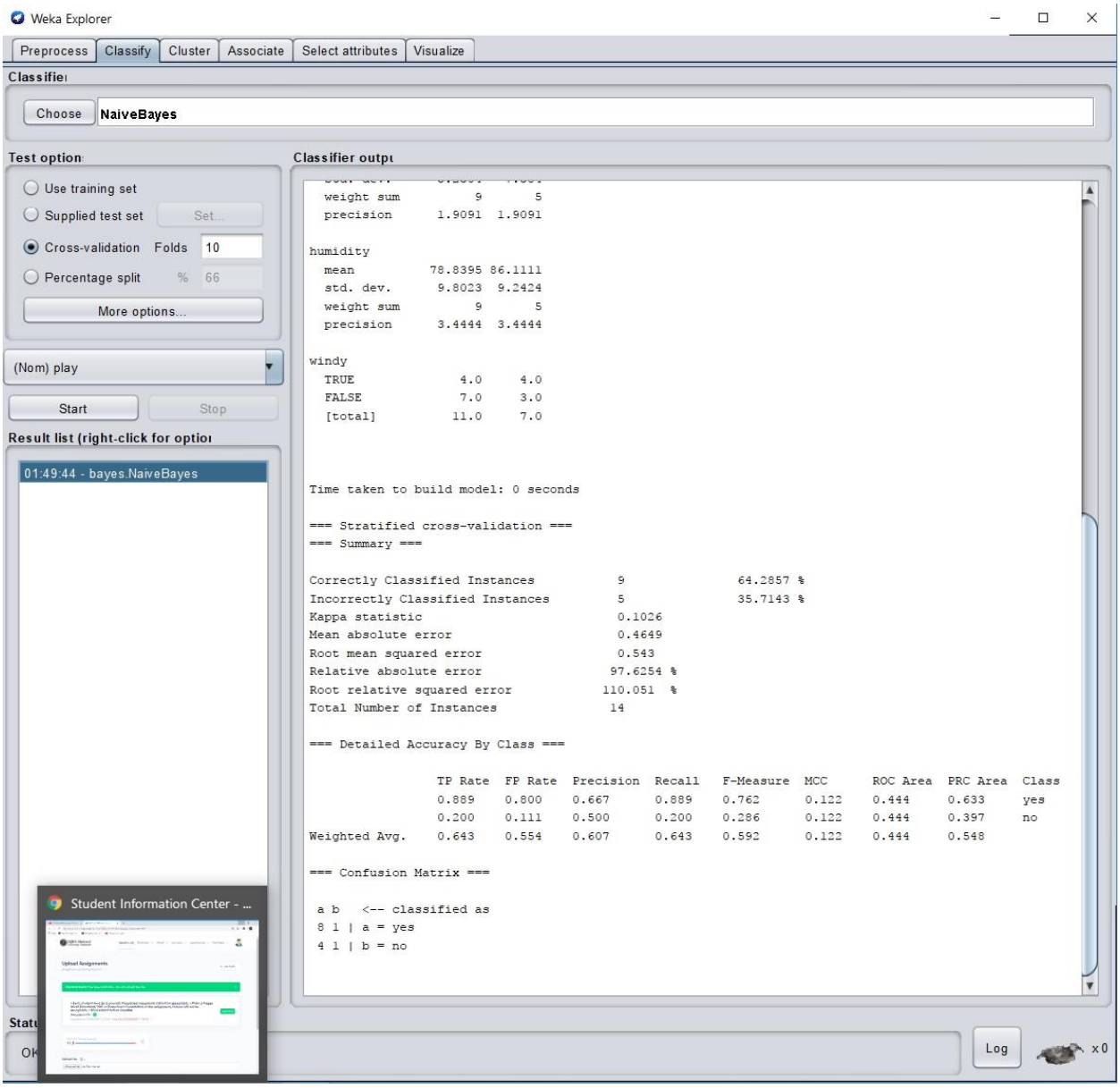
Q3:

Ans:

- First download any dataset from any source you like.
  (make sure it is in .arff extension)
- Open Weka software.
- Click on the explorer button.
- Then click on open file button on top left corner.
- Browse to file you've just downloaded
- Add filters if you want.
- Then click on classify button on top.
- Apply various classifier algorithms (naive Bayes, J48 etc.).




Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
NaiveBayes
Test option
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
Classifier output
=== Run information ===
Scheme: weka.classifiers.bayes.NaiveBayes
Relation: weather
Instances: 14
Attributes: 5
outlook
temperature
humidity
windy
play
Test mode: 10-fold cross-validation
=== Classifier model (full training set) ===
Naive Bayes Classifier
Class
Attribute yes no
(0.63) (0.38)
outlook
sunny 3.0 4.0
overcast 5.0 1.0
rainy 4.0 3.0
[total] 12.0 8.0
temperature
mean 72.9697 74.8364
std. dev. 5.2304 7.384
weight sum 9 5
precision 1.9091 1.9091
humidity
mean 78.8395 86.1111
std. dev. 9.8023 9.2424
weight sum 9 5
precision 3.4444 3.4444
windy
TRUE 4.0 4.0
FALSE 7.0 3.0
[total] 11.0 7.0
(Nom) play
Start
Stop
Result list (right-click for options)
01:16:01 - trees.J48
01:20:18 - bayes.NaiveBayes
Status
OK
Log
x 0


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
NaiveBayes
Test option
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
Classifier output
weight sum 9 5
precision 1.9091 1.9091
humidity
mean 78.8395 86.1111
std. dev. 9.8023 9.2424
weight sum 9 5
precision 3.4444 3.4444
windy
TRUE 4.0 4.0
FALSE 7.0 3.0
[total] 11.0 7.0
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 0.1026
Mean absolute error 0.4649
Root mean squared error 0.543
Relative absolute error 97.6254 %
Root relative squared error 110.051 %
Total Number of Instances 14
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
0.889 0.800 0.667 0.889 0.762 0.122 0.444 0.633 yes
0.200 0.111 0.500 0.200 0.286 0.122 0.444 0.397 no
Weighted Avg. 0.643 0.554 0.607 0.643 0.592 0.122 0.444 0.548
=== Confusion Matrix ===
a b <-- classified as
8 1 | a = yes
4 1 | b = no
(Nom) play
Start
Stop
Result list (right-click for options)
01:49:44 - bayes.NaiveBayes
Student Information Center - ...
Log
x 0