Weka Exercises

KNN classifier is implemented with the name IBk
The tree classifier to use is the J48
The SVM are implemented with the name SMO (package functions)

DATASETS:

iris.arff:
https://gist.githubusercontent.com/myui/143fa9d05bd6e7db0114/raw/500f178316b802f1cade6e3bf8dc814a96e84b1e/iris.arff

glass.arff: https://raw.githubusercontent.com/renatopp/arff-datasets/master/classification/glass.arff

diabetes.arff: https://github.com/renatopp/arff-datasets/blob/master/classification/diabetes.arff

vehicle.arff: https://raw.githubusercontent.com/renatopp/arff-datasets/master/classification/vehicle.arff

ionoshpere.arff: https://raw.githubusercontent.com/renatopp/arff-datasets/master/classification/ionosphere.arff

1) In Preprocess:

   a) Load a dataset (iris.arff) and look at it
   b) Use the Data Set Editor
   c) Apply a filter (to remove attributes and instances).

2) Load a dataset (iris.arff) and classify it with the J48 decision tree learner (test on training set) :

   a) Examine the tree in the Classifier output panel
   b) Visualize the tree (by right-clicking the entry in the result list) c) interpret classification accuracy and confusion matrix.

3) Experiment with the IBk classifier for nearest neighbour learning:

   a) Load glass data (glass.arff); list attribute names and identify the class attribute
   b) Classify using IBk, testing with cross-validation
   c) Repeat using 10, 20 and 30 nearest neighbours
   d) Interpret the results and draw conclusions about IBk.

4) Experiment with the IBk classifier for nearest neighbour Learning :

   a) Load diabetes.arff data ; list attribute names and identify the class attribute
   b) Classify using IBk (3NN), testing with Hold-out (Training 70% - Test 30%)

c) Classify using IBk (3NN), testing with 10 fold cross-validation

d) Note the difference in classification between hold-out and cross validation

5) Experiment with the IBk classifier for nearest neighbour Learning :

a) load diabetes.arff data ; list attribute names and identify the class attribute

b) classify using IBk, testing with 10 fold cross-validation

c) Using different value of KNN (3,5,7,9,11,13,15)

d) Produces a plot of the accuracy regarding the number K

6) Use Naive Bayes, Decision Tree and KNN over the dataset iris.arff and diabetes.arff datasets and evaluate each of the different algorithm with 5 fold cross validation, 10 fold cross validation, 20 fold cross validation and hold out (Training 70% - Test 30%)

7) Apply Naïve Bayes (NB) and J48 on iris and diabetes datasets:

a) apply NB to vehicle.arff, glass.arff, diabetes.arff and ionoshpere.arff, using 10-fold cross validation.

b) apply J48 to the same datasets.

c) summarize the results

d) draw some conclusions about the datasets where NB outperformed J48

8) Investigate linear and non-linear support vector machines:

a) Apply SMO to iris.arff dataset, again evaluating on the training set
b) Apply the classification boundary visualizer, and visualize the classifier errors
c) Change the "exponent" option of the kernel "PolyKernel" from 1 to 2 and repeat
d) try to explain the differences in the test results

9) Apply discretization:

a) Open the iris.arff dataset and apply discretization
c) Classify using NB, evaluating with cross-validation
d) Apply the supervised discretization filter and look at the effect (in the Preprocess panel)
e) Apply unsupervised discretization with different numbers of bins and look at the effect
f) Use the FilteredClassifier with NB and supervised discretization, evaluating with cross-validation
g) Repeat using unsupervised discretization with different numbers of bins h) compare and interpret the results.

10) Create an "arff"-file containing the datapoints

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 data file using EM with k=2 and k=3 clusters.

11) Create an “arff” file containing the datapoints (sparse arff file)

t1 = (0,2,0,0,2,0,0,0) t2 = (3,2,0,0,0,0,1,0) t3 = (1,0,0,0,2,3,0,0) t4 = (4,0,0,0,2,0,2,0) t5 = (0,0,3,0,3,0,0,4) t6 = (0,5,0,5,2,0,0,0) t7 = (0,1,0,0,0,1,0,1) t8 = (0,0,5,0,0,2,1,4) t9 = (0,5,0,5,0,5,0,0)

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

12) Create an “arff”-file containing the following document-word representation (sparse arff file)

t1 = {machine, learning, classifier}
t2 = {data, mining, associative, classifier} t3 = {mining, decision, tree}
t4 = {association, mining, data}
t5 = {decision, tree, classifier}