Practical Data Mining

COMP-321B

Tutorial 2: Nearest Neighbor Learning and Decision Trees

Gabi Schmidberger
Mark Hall

July 20, 2006

1 Introduction

This tutorial contains exercises where you are asked to perform several tests. The tests are classification tasks, first using a nearest neighbor learning method, and second a decision tree method. In both areas several methods have been developed and the most important ones are implemented ready to be used in WEKA. The two methods used in this tutorial are 'IBk' as a nearest neighbor learning method and 'J48' as a decision tree building method.

Each classification is performed on data that has been selected and prepared for this tutorial. The data sets that will be used are explained in the following subsection 1.1.

Tests are made on noisy data or with data that has irrelevant attributes. Some tests are repeated with a different option setting. The measurement taken to evaluate the tests should be the accuracy (percentage of correctly classified instances).

1.1 Introduction to the data sets used

All data that is to be used those exercises has been prepared in WEKA's ARFF format.

The 'glass' data set used in this tutorial is taken from the so called UCI data sets. 1998 the University UC Irvine (UCI) started to collect data sets and offered them on the world wide web. Today they are used as a benchmark for comparing data mining algorithms (For more information on UCI data sets see [1]).

glass.arff The 'glass' data set contains data of 6 types of glass and comes from the USA Forensic Science Service. The glass is described by its oxide content (i.e. Na, Fe, K, etc).

The following data sets have been derived from the 'glass' data set and have been specially prepared to suit the exercises.

glass-minusatt.arff The data in this data set is the same as in the 'glass' data set but with some of the attributes removed.

glass-withnoise.arff The data in this data set is also generally the same as in the 'glass' data set but 10% of the instances have been randomly selected and their class attribute value changed. This means that the data set now contains 10% class noise.

glass-mini-normalized.arff This data set contains a reduced set of instances taken from the 'glass' data set (33 instances only) and the number of attributes is reduced as well to only two attributes. The values of these two numeric attributes are normalized. 'Normalized' means all values have been transformed to values between 0.0 and 1.0 relative to the original. This data set will be used to compute part of the nearest neighbor algorithm per hand.

glass-mini-train.arff and glass-mini-test.arff The data set that can be used to verify the result of your computation using the data set 'glass-mini-normalized.arff'.

1.2 Exercise A: Answer the following questions

Qu A1: Explain what is the accuracy measurement of a classifier?

Qu A2: Describe what is class noise in data?

Qu A2: Explain what are irrelevant attributes in a data set with respect to classification?

2 Perform nearest neighbor learning with the 'IBk' classifier

The first set of tasks are about performing nearest neighbor classification with 'IBk'. Tests are done on different data sets once with default option setting and twice with the 'number of neighbors' option changed to given values.

2.1 Exercise B: Apply the 'IBk' classifier to the 'glass' data

Task B1: Load the data set 'glass.arff' and answer the following questions: Which attribute is the class attribute? How many attributes does this data set have? List all the attribute names.

Task B2: Change to the **Classify** panel and run the classification algorithm 'IBk' (weka.classifiers.lazy.IBk) using cross-validation to test its performance. In order to do so, make sure that the **Cross-validation** button in the **Test options** box is selected before you press the **Start** button. The folds for cross-validation (see the field at the right to the cross-validation button) stay at their default value 10. Since you didn't change any of the options the algorithm is performed with its default options. You can see the default options in the window that pops up after you click on the field right to the 'Choose' button. As soon as you change any fields the value will stay and you have to set back to the default values by entering them in the fields changed (see for defaults Appendix A). The field 'KNN' sets the number of neighboring instances. Its default value is '1'.

Note down in the following 'Summary of results (Nearest Neighbor)' section the resulting accuracy given in the 'Classifier output' field.

Task B3: Run 'IBk' again, but this time set the number of neighboring instances to 10. This is done by entering the value '10' in the **KNN** field.

Test option is 'Cross-validation' for this task and all the following tasks if not stated differently.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

Task B4: Try setting the number of neighboring instances to 20.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

2.2 Exercise C: Repeat the tests using the 'glass-minusatt' data set

Task C1: Go back to the **Preprocess** panel and load the data set 'glass-minusatt'. Which attributes are left in this data set? List all the attribute names.

Task C2: Run 'IBk' using all default options (see for defaults Appendix A) on the data set 'glass-minusatt.arff' and cross-validation to test its performance. The number of neighboring instances might have to be set back to '1'.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

Task C3: Run 'IBk' again, this time setting the number of neighboring instances to 10.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

Task C4: Run IBk again, this time setting the number of neighboring instances to 20.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

2.3 Exercise D: Repeat the tests using the 'glass-withnoise' data set

Go back to the **Preprocess** panel and load the data set 'glass-withnoise.arff'.

Task D1: Run 'IBk' using all default options (see for defaults Appendix A).

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

Task D2: Run 'IBk' again, this time setting the number of neighboring instances to 10.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

Task D3: Run 'Ibk' again setting the number of neighboring instances to 20.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

2.4 Exercise E: Give your own summary of the Nearest Neighbor Learning results

Task E1: Comparing the results of the tasks B2, B3, B4, C2, C3, C4, D1, D2, D3, which test had the best results?

Task E2: With these results what do you conclude about the performance of the nearest neighbor classifier 'IBk'?

3 Perform Nearest Neighbor yourself

3.1 Exercise F: Perform Nearest Neighbor yourself

Load the data set 'glass-mini-normalized.arff'. Use the **Edit** button to view the data set in a table. The last instance is the instance you will classify yourself.

Task F1: Take the instance that is last in the data set 'glass-mini-normalized.arff' (instance No.33) with the values 0.463158 for 'Na' and 0.382979 for 'Al'. This is your test instance and all the other instances are your 32 training instances. Pretend that the number of nearest neighbors option is set to 1. You have to go through the testing steps yourself and and find the instance with the smallest Euclidean distance to the test instance. You don't have to normalize the values first, because in this data set all values are already between 0.0 and 1.0.

To find the nearest training instance you can either compute all distances to all other instances in the data set or you can use the **Visualize** Panel to find the nearest instance.

Give the instance number of the nearest instance and the value of the Euclidean distance between instance No.33 and its nearest neighbor instance.

Task F2: If the instance No.33 is taken as test instance and all the other instances are the training instances, what class is instance No.33 classified as?

Task F3: To verify your result from the above task, run 'IBk' using 'glass-mini-train.arff' as training file and 'glass-mini-test.arff as test file. What class is assigned to the single test set instance?

4 Classify with the decision tree learner

Next you are asked to perform tests using the same datasets but the classification method 'J48'. 'J48' builds a decision tree as a model for the data.

4.1 Exercise G: Test with J48

Task G1: Load the data set 'glass.arff'. Run 'J48' with all options set to default (see for defaults Appendix B).

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

Task G2: Visualize the decision tree that was build in Task G1. This decision tree was build using the data set 'glass.arff' as training data. You have one test instance. It has the following values:

RI: 1.52127, Na: 14.32, MG: 3.9, Al: 0.83, Si: 71.5, K: 0.0, CA: 9.49, Ba: 0.0, Fe: 0.0

The class value of this instance is 'tableware'.

Using the given decision tree, and this instance as test instance, what value will this instance be classified as? Is this classification correct?

Task G3: Use the option **Visualize classifier errors**. Select one misclassified instance. Write down its instance number, its 'real' class value and its predicted class value.

Task G4: Load the data set 'glass-minusatt.arff'. Run 'J48' with all options set to default.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

Task G5: Load the data set 'glass-withnoise.arff'. Run 'J48' with all options set to default.

Note down in the 'Summary of results (Nearest Neighbor)' section the resulting accuracy.

4.2 Exercise H: Give your own summary of the Decision Tree Learning results

Task H1: Which test had the best results?

Task H2: With these results what do you conclude about the performance of the decision tree classifier 'J48'?

5 Compare the Nearest Neighbor Learning to Decision Tree Learning

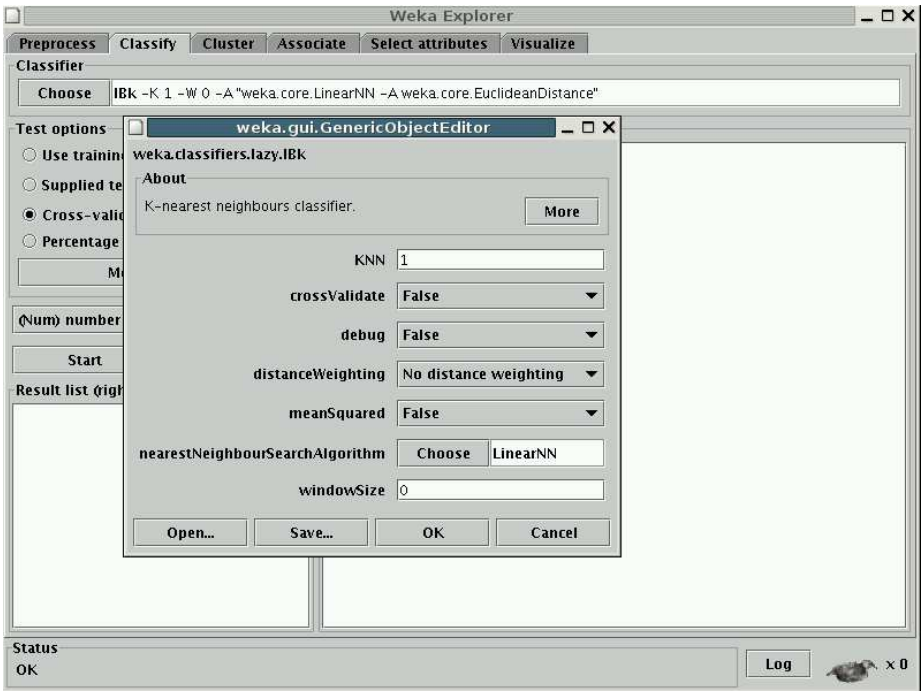
5.1 Exercise I: Compare the Nearest Neighbor Learning Results with the Decision Tree Learning Results

Task I1: Data set 'glass.arff'. Compare all results made with 'Ibk' with the results from 'J48'. Which are the best results? What are your conclusions from this comparison?

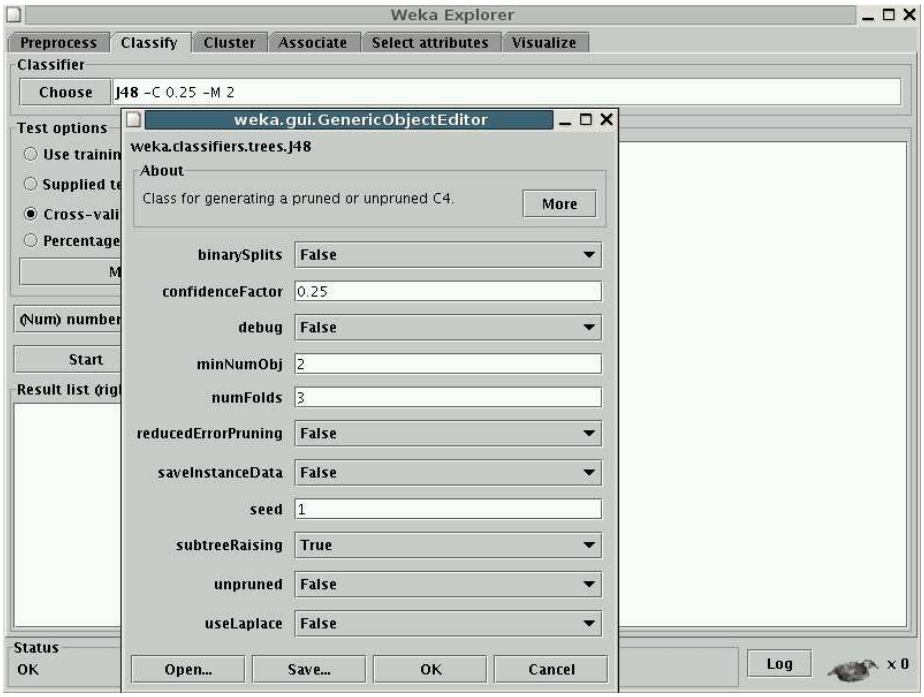
Task I2: Data set 'glass-minusatt.arff'. Compare all results made with 'Ibk' with the results from 'J48'. Which are the best results? What are your conclusions from this comparison?

Task I2: Data set 'glass-withnoise.arff'. Compare all results made with 'Ibk' with the results from 'J48'. Which are the best results? What are your conclusions from this comparison?

6 Appendix A: Default options of the 'IBk' classifier



7 Appendix B: Default options of the 'J48' classifier



References

[1] C.L. Blake S. Hettich and C.J. Merz. UCI repository of machine learning databases, 1998.

8 Answers

Answer A1:

Answer A2:

Answer A3:

Answer B1:

Answer C1:

Answer F1: No. ..... Distance .....

Answer F2: class .....

Answer F3: class .....

Answer G2: class ..... is correct (y/n) .....

Answer G3: No. .....

real class ..... predicted class .....

8.1 Summary of results (Nearest Neighbor)

Answer B2: IBk, KNN = 1, glass .....

Answer B3: IBk, KNN = 10, glass .....

Answer B4: IBk, KNN = 20, glass .....

Answer C2: IBk, KNN = 1, glass-minusatt .....

Answer C3: IBk, KNN = 10, glass-minusatt .....

Answer C4: IBk, KNN = 20, glass-minusatt .....

Answer D1: IBk, KNN = 1, glass-withnoise .....

Answer D2: IBk, KNN = 10, glass-withnoise .....

Answer D3: IBk, KNN = 20, glass-withnoise .....

Answer E1:

Answer E2:

8.2 Summary of results (Decision Trees)

Answer G1: J48, glass .....

Answer G4: J48, glass-minusatt .....

Answer G5: J48, glass-withnoise .....

Answer H1:

Answer H2:

Answer I1:

Answer I2:

Answer I3: