Practical Data Mining

COMP-321B

Tutorial 3: Naive Bayes and Support Vector Machines

Gabi Schmidberger
Mark Hall

July 31, 2006

1 Introduction

This tutorial will familiarize you with two new classification methods, Naive Bayes and Support Vector Machines. Naive Bayes will be compared with a decision tree learning method. The Support Vector Machine classification will be tested against itself varying the option settings.

In some of the exercises of this tutorial you will use the tool **Classification Boundary Visualizer**. An introduction to this tool is given in the next section. All other exercises have tests that are started from the 'Classify' panel of the WEKA Explorer.

The Naive Bayes method implementation used in these tests is the

'weka.classifiers.bayes.NaiveBayes' classifier.

The Support Vector Machines method implementation is the

'weka.classifiers.functions.SMO' classifier.

Each classification is performed on data that has been selected and prepared for this tutorial. All data that is to be used is in WEKA's ARFF format. The measurement taken to compare the tests is the accuracy.

2 Data Set Files

All Files used in this tutorial are in the folder:

```
/home/ml/321/Tutorial3
```

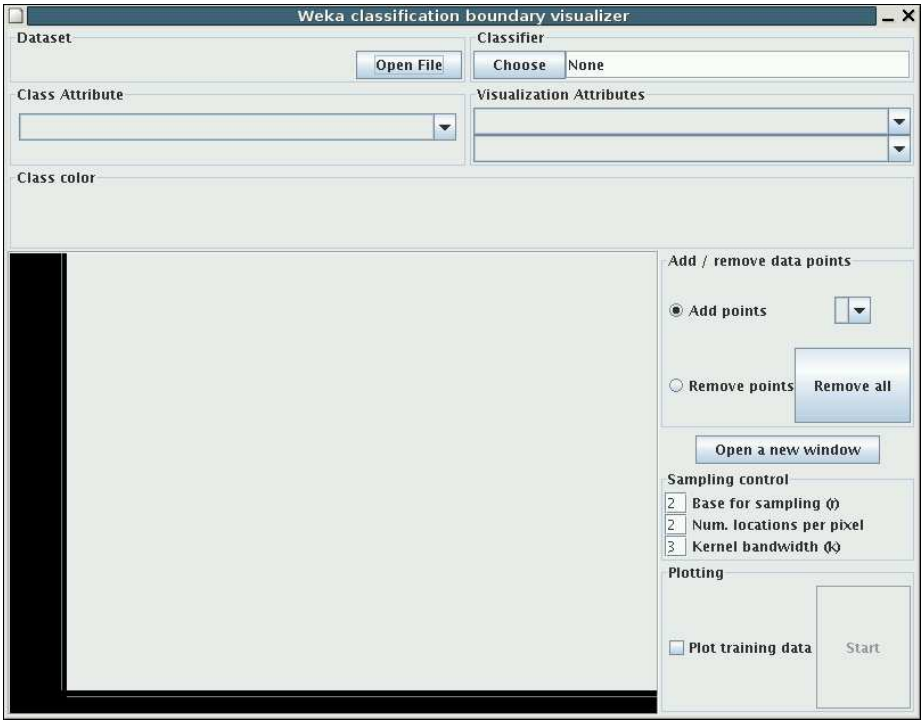
2.1 Introduction the Tool Boundary Visualizer

The Boundary Visualizer builds two-dimensional pixel plots. One colour is automatically chosen for each class and all the area on the plot where the classifier predicts a particular class is plotted with its respective colour. The decision boundaries of a classifier in the selected two dimensions can be recognized by the change of colour. With the select fields on the top of the window the attributes on x-axis and y-axis can be changed (just like in the Visualize panel of the Explorer). The user selects a classifier and a data set. First the classifier is trained on the supplied data set. Then for each pixel in the two-dimensional plot several data instances are generated and classified. The colour of the pixel is set according to the predicted class values of these instances.

This is how the Boundary Visualizer is started from the command line:

```
java weka.gui.boundaryvisualizer.BoundaryVisualizer
```

After you entered this the **WEKA classification boundary visualizer** window opens. The pixel plot area in the lower left corner is still empty.



Press the button **Open File** in the 'Dataset' field and select the file 'iris.arff'. Select the classifier 'J48' (**Choose** button in the Classifier field.). To start the building of the plot press the **Start** button in the right corner, but select **Plot training data** first. The bounday plot is build and gradually improved. After it is finished the change in colour shows the three different decision areas for the three classes in 'iris.arff'.

In the fields 'Class Attribute' and 'Visualization Attributes' the attributes for the x and y-axis of the plot and the attribute used for class attribute can be changed. (Default class attribute is again the last attribute in the dataset.) After a change to any of these the 'Start' button must be pressed again to rebuild the boundary plot.

The rebuilding of the plot can be speed up with changing the value of 'Num. locations per pixel' to 1. (We want to ignore all other fields in the 'Sampling control' field.)

Points can be added or removed by selecting the option 'Add points' or 'Remove points' in the 'Add / remove data points' field and left clicking in the pixel plot area.

3 Compare Naive Bayes with J48

We are already familiar with the classifier J48. J48 is a decision tree learner and was used in tutorial 2. We want to compare the Naive Bayes classifier 'NaiveBayes' with J48. The data sets we use for this comparison are explained in the following subsection.

3.1 Introduction to the data sets used in the 'NaiveBayes' tests

Most data sets used in the exercises of this tutorial are again taken from the UCI data sets (See tutorial 2).

vehicle.arff This data set comes from the Turing Institute Glasgow, Scotland it contains examples of vehicle silhouettes. The purpose of the data set is to classify one of four types of vehicles. The vehicle may be viewed from one of many different angles.

kr-vs-kp.arff The instances describe a chess game position. The name is short for king+rook versus king+pawn on a7. The pawn on a7 means the game is one square away from queening. It is the king+rook side (white) to move.

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).

waveform-5000.arff The data was given by David Aha in the year 1988. It is generated by a waveform database generator. It contains instances of 3 classes of waves. The 21 attributes all contain noise.

generated.arff This is an artificial data set that has been generated using the WEKA data generator. The values of the attributes are normally distributed.

3.2 Exercise A: Apply 'NaiveBayes' and 'J48' on several data sets

Task A1: Load the data set 'vehicle.arff' and run 'Naive Bayes' and 'J48' on it. Set the test options to 'cross-validation' with number of folds is '10'. Note down the two resulting accuracies.

Task A2: Load the data set 'kr-vs-kp.arff' and run 'Naive Bayes' and 'J48' on it. Set the test options to 'cross-validation' with number of folds is '10'. Note down the two resulting accuracies.

Task A3: Load the data set 'glass.arff' and run 'Naive Bayes' and 'J48' on it. Set the test options to 'cross-validation' with number of folds is '10'. Note down the two resulting accuracies.

Task A4: Load the data set 'waveform-5000.arff' and run 'Naive Bayes' and 'J48' on it. Set the test options to 'cross-validation' with number of folds is '10'. Note down the two resulting accuracies.

Task A5: Load the data set 'generated.arff' and run 'Naive Bayes' and 'J48' on it. Set the test options to 'cross-validation' with number of folds is '10'. Note down the two resulting accuracies.

3.3 Exercise B: Give your own summary of the Naive Bayes learning results

Task B1: Which of the two algorithms performed better or equal and how often did it perform better or equal?

Task B2: Use the functions in the 'Preprocess' panel to have a look at the attributes of the data set(s). Can you say anything about the nature of the attributes of those data sets where Naive Bayes outperformed J48?

4 Classify with the Support Vector Machine method

4.1 Introduction to the data sets used in the Support Vector Machine tests

glass.arff In this section we use the data set 'glass.arff' from Tutorial 2 again. It is one of the UCI data sets. 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).

Further data sets have been derived from the 'glass' data set by taking some of the attributes away and by taking only instances of certain types.

glass-RINa.arff The data from 'glass.arff' but with just the instances of the classes 'build wind float' and 'headlamps' and only the attributes 'RI', 'Na' and the class.

vehicle.arff See the description in the previous section.

vehicle-sub.arff Part of the data from 'vehicle.arff'. It contains only two of the attributes and is reduced to two of the class values.

4.2 Exercise C: Apply 'SMO' (linear hyperplane)

The method 'SMO' has many options. The following exercises ask for changes of only two of the options, 'c' the complexity value and 'exponent' the exponent of the dividing hyperplane. The default of exponent is 1.0 which means that the dividing hyperplane is linear.

Task C1: Load the data set 'glass-RINa.arff'. Run 'SMO' with all options set to default and with the test option set to 'Use training set'. Note down the resulting accuracy.

Task C2: First use the 'Visualize classifier errors' option and second, start the WEKA classification boundary visualizer from a new commad window and plot a classification boundary plot with the dataset and classifier used in Task C1. Describe the model built and explain the classification errors.

Task C3: Use the same data set 'glass-RINa.arff'. Change the option 'c' to 20.0 and run 'SMO' again with test option again set to 'Use training set'. Note down the resulting accuracy.

Task C4: Again visualize the classification errors and build a classification boundary plot. Explain the differences in the test results of C1 and C3.

4.3 Exercise D: Apply 'SMO' (linear and non-linear hyperplane)

Task D1: Load the data set 'vehicle-sub.arff'. Run 'SMO' with all options set to default and with the test option set to 'Use training set'. Note down the resulting accuracy.

Task D2: First use the 'Visualize classifier errors' option and second plot a classification boundary plot with the WEKA classification boundary visualizer and with the dataset and classifier used in Task D1. Describe the model built and explain the classification errors.

Task D3: Use the same data set as in Task D1. Run SMO again but change the 'exponent' option (is an option of the kernel 'PolyKernel') to 2. Note down the resulting accuracy.

Task D4: Again visualize the classification errors and build a classification boundary plot. Explain the differences in the test results of D1 and D3.

Task D5: Add or remove points in the plot to change the shape or position of the decision boundary. Report how many points (approximately) you have added or removed, at which area of the plot and the change to the boundary.

5 Answers

5.1 Results of the comparison NaiveBayes with J48

——————————— Naive Bayes ———-+——— J48 ————

Answer A1: vehicle.arff ....

Answer A2: kr-vs-kp.arff ....

Answer A3: glass.arff ....

Answer A4: waveform-5000.arff ....

Answer A5: generated.arff ....

————————————————————————————————————-

Answer B1:

Answer B2:

5.2 Results of the Tests with SMO

Answer C1: glass-RINa.arff, default .....

Answer C3: glass-RINa.arff, c = 20 .....

Answer C2:

Answer C4:

Answer D1: vehicle-sub.arff, default .....

Answer D3: vehicle-sub.arff, c = 20 .....

Answer D2:

Answer D4:

Answer D5: