Lab 5
A Survey of Classification Learning Algorithms

Overview: In previous labs we've experimented with the following classification learning algorithms: *ZeroR, OneR, J48* (for generating decision trees), and *IBk* (nearest neighbor), though not necessarily in the order we've covered them in lecture. Today we review those algorithms, together with *Naïve Bayes* and *PRISM*, in a lab activity that should be both an effective exam review and preparation for the final project.

> You may work in teams of as many as 3 on this lab activity. Each team should submit one lab report. I suggest you get your final project team together. Otherwise, perhaps the people you work with today could become your team.

1. The Simplified Crime Dataset

Check Box for a file called `njcrimenominal.arff`. Save it to your USB drive. This is the dataset we'll be using today. (I call this the "simplified" crime dataset because next week we'll practice preparation of data using a much larger, messier crime dataset.)

a. Start Weka and open the file. Review previous labs for this and any other basic Weka steps that you may have forgotten.

b. Click the "Edit" button and spend some time with your data. Make sure you can interpret each attribute and the attribute values correctly. Discuss with each other. Ask me if you can't interpret something.

c. Back in the Explorer window, make sure `crime` is the class attribute. Click on each attribute in turn to view its properties.

d. Click on "Visualize All" and look at the histograms. Which attributes seem as if they are likely to contribute to a bad or ok crime rate? You don't have to write anything down – by now this should be a habit. Discuss with one another and with me.

> This is an important analytical step to think about for the final project. Looking at these histograms might give you some idea of creative data visualizations that you might find more useful for your final project.

2. Surveying the Classification Learning Algorithms

Now let's work our way through the five classification algorithms from Chapter 4 of Data Mining. Follow the instructions below, and fill in the table on the lab report as you go.

ZeroR

a. Under the Classify tab, choose the *ZeroR* classifier from the "rules" set. ***For this and all subsequent experiments today, use 10-fold cross validation for testing purposes.***

b. Run the classifier and record the performance as an accuracy percentage on the lab report.

c. For discussion only – you don't have to write down any answers – ask yourself and your teammates: How does this algorithm work? Why did it give the percentage it did? Is this a good percentage? If it is, does that mean *ZeroR* is a valid classifier to use in practice? If not, why not? If not, what is *ZeroR* good for?

OneR

d. Now choose *OneR* from the rules group. Use all the default settings.

e. Run the algorithm and record the accuracy percentage in the table.

f. Also record the rule generated by *OneR* in the table.

g. For discussion only – you don't have to write down any answers – ask yourself and your teammates: How does this algorithm work? Is the rule that was generated what you expected? Can you give an explanation for it in any case? Is this a good percentage? Is the percentage the same as it would have been running on the test data? Is *OneR* a valid classifier to use in practice? Under what circumstances is it valid? Under what circumstances is it not? When it isn't valid to use, what is it useful for?

Naïve Bayes

h. Next we get our first hands-on look at the Naïve Bayes statistical method of classification. In the "bayes" set, choose `NaiveBayesSimple`, which is the implementation of the algorithm that we learned in class.

i. Run the algorithm and record the accuracy percentage in the table.

j. Then take some time to scroll through the "Classifier output" window to read and understand it thoroughly. Record in the table the probabilities recorded for the values `hi`, `med`, and `low` of attribute `pop`, when the output class is `bad`. (Don't be thrown by the fact that the values don't line up in the Weka window.)

k. Use the "Edit" button under the "Preprocess" tab to look at the data again. Make sure that you are able to explain to yourself the meaning of the values that you just recorded in the table. Doing so will mean you understand this pretty well. Test your knowledge with other values in the Weka output. Perhaps even draw a portion of the probability table for this problem and see if you can predict one of the sets of values.

NOTE: You should quickly discover that Weka's implementation of this algorithm uses the Laplace estimator whether or not there is a zero probability. This differs from the examples we did in class.

l. For discussion only – you don't have to write down any answers – ask yourself and your teammates: How does this algorithm work? Do you see the connection between the Weka output and the probability tables that we've created in lecture and in problem sets? What characteristics of the dataset would make this algorithm particularly effective? Particularly ineffective? How would numeric data change the way the algorithm works? How would the algorithm be affected by missing values in the dataset or in the test instance?

J48

m. Now choose the *J48* classifier algorithm, as you have done several times before. Run the algorithm and then right-click on the latest model in the Result List and visualize the decision tree. Make sure it makes sense to you.

n. Record the accuracy percentage in the table. Also record the attribute that the algorithm chose to split on first.

o. For discussion only – you don't have to write down any answers – ask yourself and your teammates: How does the algorithm that we learned in lecture work? How is it different (in general terms) from *J48*? Why might *J48* have picked a different attribute to split first than *OneR* selected for its rule? How would numeric data change the way the algorithm works? How would the algorithm be affected by missing values in the dataset or in the test instance?

PRISM

p. Time for our first hands-on look at the PRISM covering algorithm that we learned in lecture. From the "rules" group, select PRISM and run it.

q. Record the accuracy percentage in the table. Also record the first rule generated by PRISM. Does it strike you as a good, valid rule? Record a brief explanation of what underlying real-world truth the rule may reveal.

r. Take some time to study the rules. Make up a few test instances and make sure you know how to apply the rules to yield a classification.

s. For discussion only – you don't have to write down any answers – ask yourself and your teammates: How does the algorithm work? How is the first rule generated? Take a moment with your teammates to run through the algorithm with pencil and paper to generate this rule. (It should take about a minute if you know what you're doing.) Maybe try the same with a more complex rule. How would numeric data change the way the algorithm works? How would the algorithm be affected by missing values in the dataset or in the test instance? In this case, you should try it by temporarily deleting one of the values from the dataset. Can you explain what happens?

IBk (nearest neighbor)

t. Select *IBk* from the "lazy" group and run it.

u. Record the accuracy percentage in the table. Then change the "neighborhood" – the value of k – to 3 and run it again, recording the accuracy. Repeat one more time with $k = 5$. Briefly explain in the table any change that you observed.

v. For discussion only – you don't have to write down any answers – ask yourself and your teammates: How does the algorithm work? Why isn't there anything in the Classifier output window under the words "Classifer model"? Why might eliminating some of the attributes improve the performance of the algorithm? How would numeric data change the way the algorithm works? How would the algorithm be affected by missing values in the dataset or in the test instance?

x. Complete the questions on the lab report and submit them any time before 2:30 on Tuesday, March 17.

Lab 5: A Survey of Classification Learning Algorithms

Names: _____

Complete the following table in accordance with the lab instructions:

Classification Algorithm	**Accuracy with 10-fold cross-validation**	**Additional Info (if required)**
ZeroR		N/A
OneR		Best rule:
NaiveBayesSimple		Class bad Attribute pop *hi:* *med:* *low:*
J48		Best split attribute:
PRISM		First generated rule: Explanation:
IBk	**k=1:** **k=3:** **k=5:**	Why did the accuracy change?