CS 374 Assignment #7
Evaluation of Learning Algorithms

Due the week of April 5, 2021

—

This week you will step back from your study of individual classifier-learning algorithms in order to evaluate them both empirically and analytically. The readings and exercises will focus on evaluation methodology, but the broader goal of this assignment is to give you the opportunity to reflect on all that you've already learned this semester.

Notes and advice as you prepare to do this assignment. This assignment involves an empirical comparison of the performance of several learning algorithms on many sets of data. I recommend that you set up the experiments and gather the raw data as soon as possible. That way, you will have it available whenever you are ready to analyze the data and write up the results. To give you a sense of timing, the experiments took slightly over six hours to run on my laptop.

What to turn in. For this assignment, you will need to complete four short exercises, perform an empirical study, and write a report. Solutions to the four exercises should be written up individually. The report can be done either individually or with your tutorial partner. Please turn in the problem set and report as two separate pdfs.

What to expect during the meeting. Your job during the meeting will be to formally present the material in your report. Whether you use slides, handouts, or the board is up to you, but I will expect a presentation of your results and analysis, with time for questions and discussion. I expect this will take approximately one hour. We will reserve the remaining 10-15 minutes for individual meetings, where I will provide you with feedback on our tutorial meetings. I try to provide suggestions throughout the semester, but it's always nice to have a formal check-in around the midpoint.

1 Assessing and Comparing Classifier-Learning Algorithms

1.1 Reading

Please read the following:

- Mitchell, Chapter 5 and
- Alpaydin, Chapter 20 (through 20.7 only) in the fourth edition or Chapter 19 (through 19.7) in the third edition.

I recommend reading Mitchell first.

1.2 Exercises

To check your understanding of the readings, please write up and turn in the following exercises:

- 5.1-5.4 on page 152 of Mitchell.

1.3 Report on the Evaluation of Learning Algorithms

Your central task this week is to write a report that discusses and compares the classifier-learning algorithms you have studied this semester, including:

- k Nearest Neighbor (specifically, 1-NN and 3-NN)
- Naive Bayes
- C4.5 (J48 in Weka)

- SMO (an SVM-learning algorithm; specifically the default linear case and a radial basis function kernel)
- Backpropagation for multilayer perceptrons (i.e., neural nets)

Your report should include an empirical evaluation as outlined below. It should also include a general analysis and discussion of the pros and cons of each of the methods evaluated. You might even find it useful to consider some of the major theorems that relate to each of the algorithms.

1.3.1 Empirical Evaluation: Comparison of Average Error Rates

Your report should include comparisons of the average error rates (alternatively, the average accuracies – it's up to you) of each of the algorithms on several data sets. You must consider at least the 20 data sets in

```
~andrea/shared/cs374/ComparData, i.e., /home/faculty/andrea/shared/cs374/ComparData
```

Note that they are also available in a tar'd and compressed file on Glow.

Most of these data sets are from the UC Irvine Machine Learning Repository:

```
http://archive.ics.uci.edu/ml/
```

where you can find high-level descriptions of them. Note that the data sets have been converted into arff format for Weka.

I selected the data sets above for several reasons:

- They include both large and small numbers of examples.
- They include both continuous and nominal attributes.
- Some have missing attribute values.
- They include both two-class and multi-class problems.

While you must include the specific data sets named here, you should feel free to extend your evaluation to other data sets as well.

To run the experiments, begin (as usual) by starting up Weka. Do so as follows if connecting to a CS lab machine:

```
java -Xmx4g -jar /usr/share/java/weka.jar
```

Or type the following if working on your own laptop:

```
java -Xmx4g -jar weka.jar
```

Note that this week I recommend making the maximum heap size bigger than usual. You'll need it.

Use the Weka Experimenter (recall the exercise you did in order to find the optimal value for k for the k-Nearest Neighbor assignment). You should perform 10-fold cross validation for each algorithm for each data set. For this exercise, I will not require that you repeat the 10-fold cross validation more than one time. (So be sure to adjust the parameter in the right half of the window that asks for the number of times to repeat. The default is 10 – change it to 1.) When you set up the experiments (in the Set Up panel), be sure to specify a Results Destination. This will save your experiment results to a file so that you can analyze them at a later time.

In running the experiments you will find that the multilayer perceptrons (MLP) take an especially long time to train. For this reason, I recommend you start this assignment early. You might also consider setting up a single experiment for all of the data sets and all the algorithms *except* MLP. Then you can gather the MLP results separately and (very carefully) add them to the arff output from the other experiments so that you can compare MLP against all the other algorithms. **Please run the experiments using Weka's default parameters in all cases.**

Weka's "Analyze" panel will allow you to generate lots of tables. Please include appropriate tables with the results in the report you write. How do the algorithms compare to each other? Which differences in error rates (accuracies) are statistically significant? Be sure to compare each algorithm to all of the others. This is easy to do in the Analyze panel. Just change the test base. Are there any general take-away messages you can glean from these results? Be sure to report those in your writeup. Don't be afraid to dig into the data sets a bit. Are there particular characteristics of the data that seem to point to the utility of specific algorithms?

Note that one great feature of the Analyze panel is the ability to save your Weka results in LaTex format. (To do this, set the output format, then perform the test, and then save the output.)

1.3.2 General Analysis of Learning Algorithms

In addition to performing the tests specified above, you should take this assignment as an opportunity to reflect on the general advantages and disadvantages of each of the algorithms you have studied this semester. Alpaydin notes (on page 600 in the fourth edition and page 550 in the third edition) that criteria for evaluating and comparing classification algorithms include

- training time and space complexity,
- testing time and space complexity,
- interpretability (or comprehensibility of the learned model), and
- easy programmability.

Be sure to include these in your report. You might also consider the number of hyperparameters to be set. As noted above, you might even want to refer to some of the major theorems you have proved.

1.3.3 Optimizing the Algorithms

As an extra challenge (this is not a requirement for the assignment), try out various parameter settings for each of the learning algorithms to find those that work best on the data sets above.