

CS 374 Assignment #10 Ensemble Methods

Due the week of April 21, 2008

As you discovered last week, computational learning theory makes it possible for us to analyze learning algorithms. It provides tools that allow us to answer questions such as “what is learnable”, “how many examples do we need in order to learn a classifier with certain accuracy”, etc. In addition, theoretical work sometimes leads to the development of very practical algorithms, such as AdaBoost. This week you will continue your study of AdaBoost, as well as other *ensemble* methods.

1 Boosting and Bagging

1.1 Reading

Please read the following:

- Alpaydin, Sections 15.1, 15.2, 15.4, and 15.5.
- Since the reading in Alpaydin refers to the bias-variance tradeoff, you might also find it useful to read Sections 4.3 and 4.7.
- For another introduction to Boosting and Bagging, you might consider reading Duda, Hart, and Stork, Sections 9.5.1 and 9.5.2.

1.2 Exercise 1

At the top of page 10 in “A Short Introduction to Boosting”, which you read last week, Freund and Schapire discuss the advantages of AdaBoost. One of the claims they make is that AdaBoost has no parameters to tune except for the number of rounds, T . Do you agree with their claim? Why or why not? Briefly itemize the advantages and disadvantages of AdaBoost.

2 An Empirical Evaluation of Decision Tree Ensemble Methods

Please read “A Comparison of Decision Tree Ensemble Creation Techniques” by Banfield, Hall, Bowyer, and Kegelmeyer, which was published in IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) in January 2007. Then write a summary and critique of the paper. Be prepared to discuss the paper in your tutorial session.

3 The Effect of Noise

We have often discussed the potential impact of noise on classifier learning. This week you will have the opportunity to explore the effects of class noise on ensembles of decision trees and other learners.

3.1 Reading

Please read “An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization” by Dietterich, which appeared in the Machine Learning Journal in 2000. (Note that the PAMI paper referred to this one quite a bit.) In this paper, Dietterich compares the effectiveness of randomization, bagging, and boosting for improving the classification accuracy of C4.5 on 33 data sets from the UC Irvine Machine Learning Repository. He finds that AdaBoost performs better, on average, than Bagging. However, when class noise is added to the data sets, he finds that, on average, Bagging outperforms AdaBoost. This has been observed by others as well. (Why might we expect AdaBoost to be affected more significantly by noise in the data?)

3.2 An Experimental Comparison of Two Methods for Constructing Ensembles: Bagging and Boosting

The results presented in Dietterich’s paper are quite extensive and convincing. However, one of the issues we’ve discussed frequently this semester is reproducibility. This week you will empirically assess boosted and bagged decision trees, both with and without class noise.

In addition, it is interesting to consider whether we would observe the same general behavior with a different base classifier. So you will also explore the behavior of Boosting and Bagging with other learners as your base classifiers. During your tutorial session you will be assigned one of three: decision stumps, Naive Bayes, and 3-NN.

The product of your investigation will be a report that summarizes your findings.

3.2.1 Empirical Evaluation: Comparing Algorithms on Data Sets with No Noise

Before assessing the impact of class noise on learning, it is important to determine the baseline performance of each of the algorithms:

- Decision Trees
- AdaBoostM1 with Decision Trees as base classifier
- Bagging with Decision Trees as base classifier

as well as the same for your assigned base learner (of stumps, Naive Bayes, and 3-NN). You already know how to find decision trees, Naive Bayes, and 3-NN in Weka. In addition, you can find Decision Stumps in the “trees” folder, and you’ll find AdaBoostM1 and Bagging in the “meta” folder. You will need to set some parameters for AdaBoostM1 and Bagging. For AdaBoostM1, set the number of iterations to 50. You can also modify the seed for the random number generator. Don’t modify “useResampling”, and don’t worry about the weight threshold. For Bagging, set the number of iterations to 100 and the bagSizePercent to 100. Again, you can modify the seed for the random number generator. Be sure to set the base classifier appropriately in all cases. That includes setting the parameters for J48. For instance, set the confidence factor to 0.10 as Dietterich did.

The data sets to be evaluated are as follows:

- autos
- credit-a
- kr-vs-kp
- labor

- segment
- sonar
- splice
- vehicle

You can find the data sets in

```
~andrea/shared/cs374/UCI
```

You'll note that some of these are were tested by Dietterich, while some were not. Please be sure to perform 10-fold cross validation. Your report should give the error rate of each algorithm on each of the data sets. You can refer to Table 1 in Dietterich's paper for ideas on how to present the results.

3.2.2 Empirical Evaluation: Comparing Algorithms on Data Sets with Noise

Now you're ready to assess the impact of noise on each of the algorithms. You should begin by injecting class noise into each of the data sets above. There should be three noise levels per data set: 5%, 10%, and 20%. To inject noise, please follow the approach suggested by Dietterich at the bottom of page 9 in his paper.

Now run each of the algorithms again, this time on the noisy data sets. Give a table with the resulting error rates. You can, again, follow the format suggested in Table 1 of the paper. How do your results compare to Dietterich's? Are you observing the same general trends?

Feel free to extend your analysis to any of the other UCI data sets.