Homework 5: Bagging and Boosting

Jef Harkay

November 17, 2011

1 Bagging

Bootstrap aggregating, or bagging, is an algorithm that can be used to improve regression or classification problems. To perform bagging, you first take a sample of rows with replacement–meaning you can have repeated rows–from your dataset. You then run this newly created sample through your regression or classification algorithm and repeat the process for other samples. After you've created and processed a certain amount of samples, you average the outputs together to form a final output. The idea is that this final, averaged output represents the entire dataset with little variance.

2 Boosting

Boosting is along the same lines of bagging, but the implementation is obviously a little different. To perform boosting, you send your data through a "weak" algorithm–or one that produces a training error that's lower than 50%. You then use the output from this algorithm to update the weights for each row. The correctly classified rows are assigned a lower weighting, and the misclassified rows are assigned a higher weighting. The idea is that when the data is sent back through the algorithm, the higher weighted values get higher priority. In the end, you have a dataset that has an error really close to zero, and if it's really close to zero, then you have a very unbiased dataset as well.

3 Bias and Variance

Bias means ones own opinions are getting in the way of making a clear decision. When applied with machine learning/statistics, it means the output of the algorithm or model contains some error.

Variance basically describes the spread, or how far apart the data is from one-another. In machine learning, it can be applied to how much the output changes if you alter the training set.

4 Results

I ran two datasets with bagging, boosting, and the plain old j48 algorithm. The two datasets are house votes and CPU with vendor information. The votes file contains 435 instances, with 17 features (including the classification), and the CPU file contains 209 instances, with 8 features (including the classification).

As described in the homework assignment, I ran the bagging algorithm with and without pruning and with numIterations set to 30. I did the same thing for the boosting algorithm but also changed the weightThreshold to 1000. For the j48 algorithm, the only thing I changed was whether pruning was on or off. I also ran each algorithm using the training set, not cross validation.

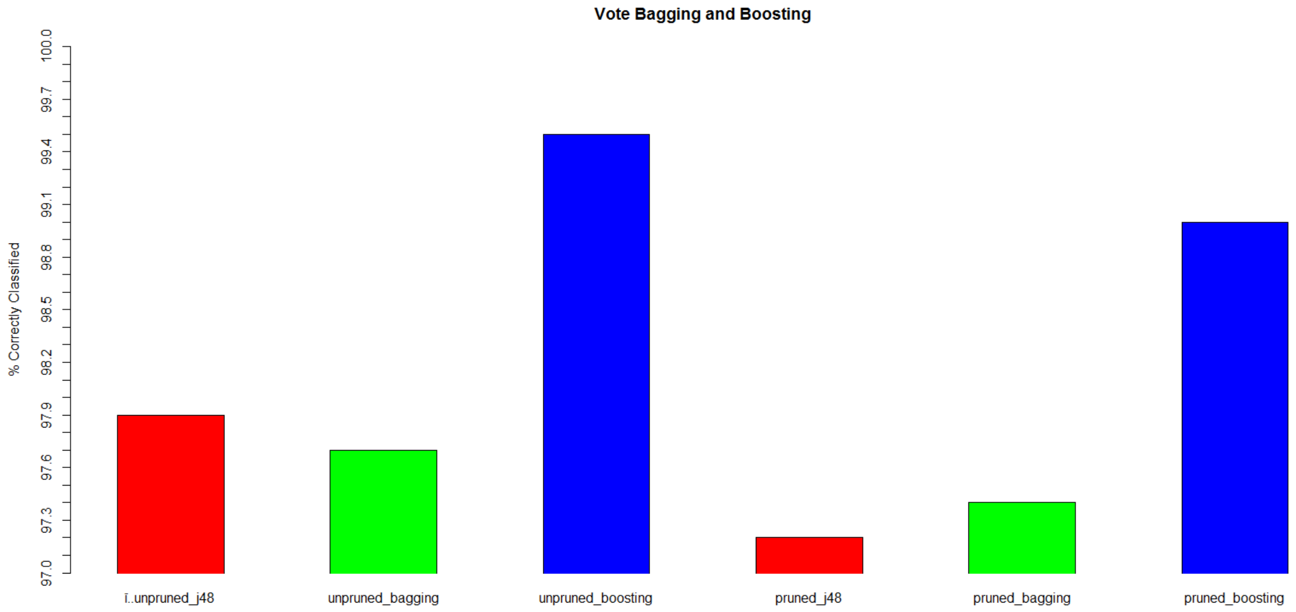

Figure 1: Vote dataset results

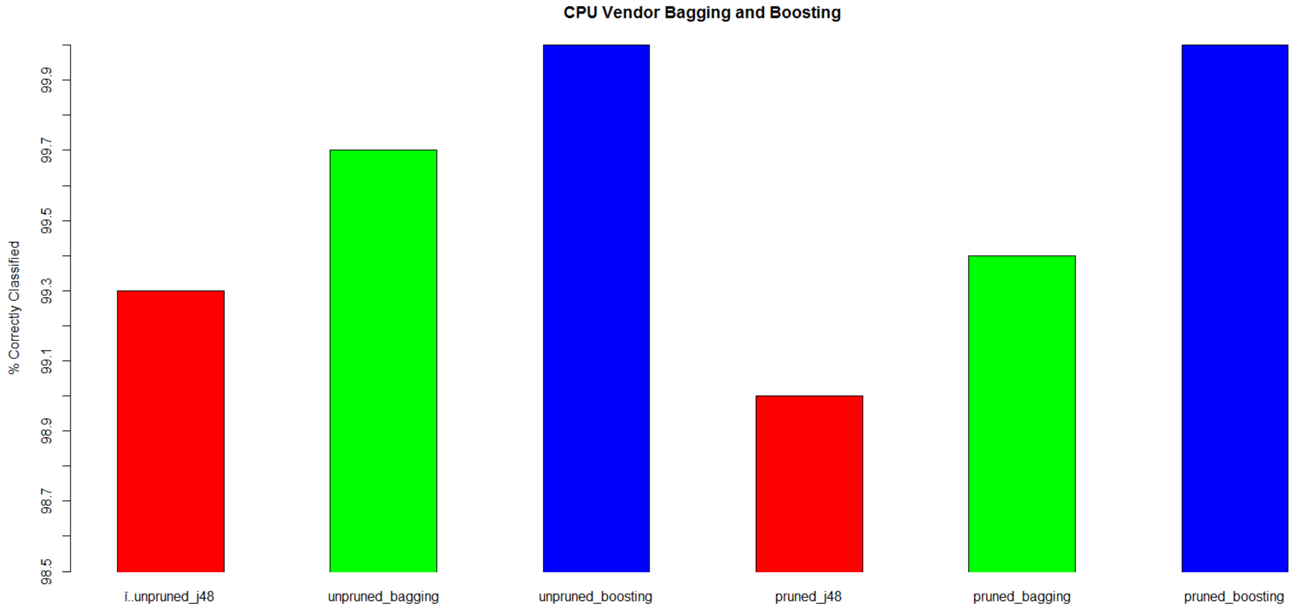

Figure 2: CPU dataset results

5 Conclusion

5.1 Results Discussion

As you can see from Figure 1 and 2, boosting always does better than bagging and the plain old j48 algorithm. Bagging does better than the plain old j48 algorithm when pruning is on, but seems to vary when pruning is off. What's interesting is that pruning almost always makes the accuracy of classification go down, with the exception being boosting in Figure 2–it gets 100% accuracy with and without pruning. The idea behind pruning is that it's supposed to help achieve better accuracy of classifying the data because you're reducing overfitting, but that doesn't seem to hold true in these cases.

5.2 Boosting Discussion

For boosting, the variance is lowered by altering the dataset on each iteration by applying weights to each row of data. If the row is misclassified, its value gets raised and vice-versa. By changing the value, we're increasing the variance, but that's only because we want the algorithm to notice it, so it can focus more on that row. If it focuses more on that row, then we're looking to lower the variance.

The unique thing about boosting is that this weighting scheme kind of goes hand in hand with reducing the bias. If the algorithm is looking for the best weights to reduce the error of misclassified data, then that means the output will have as little error in its prediction. The fact that it's doing this means the algorithm has the best interests of the data in mind–its sole purpose is to get as few misclassified data as possible–and therefore means it's unbiased.

5.3 Bagging Discussion

Bagging's primary focus is on lowering the variance. Bagging takes a sample of data, runs it through the regression or classification algorithm and gets an output. This process is repeated until you have completed a certain number of iterations. After all of the iterations are done, the outputs are averaged together, which lowers the variance because that's what happens when you average points together–you find a common ground between them. So the spread of the data has decreased.

The problem with averaging the data is that you aren't really caring about the error of misclassifying data. That's why it's best to use low bias classifiers when you're using bagging–if the classifier has a low bias, then you don't have to worry about misclassifying data as much. However, not all data can be fit into a low bias classifier, hence why we can't always use one.

5.4 Pruning and Unpruning

I think the reason why pruning produces a lower accuracy is because bagging and boosting are in place to reduce variance. The idea is if you reduce variance, then you are helping to reduce overfitting. The problem with that is, pruning is also in place to reduce overfitting. Therefore, if we're lowering the variance with pruning and bagging and boosting are also trying to attempt this, then we're bound to have a lower accuracy because we're starting to cut out data that actually has meaning.