Christopher Morgan

CS478 : Brother Christophe

Oct 17, 2013

Complete the following:

* **Show that stacking *N* (base) algorithms with a majority (meta) learner (where the stacking considers only the predictions of the base learners) does not produce the same result as an ensemble of the same *N* (base) algorithms combined by majority voting. You need not write a formal proof; make your argument in prose (with diagrams as needed).**

The stacking described and ensemble described don’t produce the same prediction, unless the meta learner for the stacking algorithm is a simple majority vote. This is because in stacking the meta learner learns from the predictions of each intermediate learner, where as the ensemble just predicts the majority. In short one returns the majority and the other learners from the predictions, therefore they are not guaranteed to return the same result.

For example, imagine in the set of learning algorithms there are two algorithms that when their votes agree the two algorithms are always right, however, in aggregate with the other algorithms they are usually out voted. In this case, the stacking meta learner would have a chance to discover this truth and give their dual vote weight, where as the simple majority ensemble would miss this fact.

* **Using Weka and a couple of datasets of your choice, compare the 10-fold cross-validation accuracies of the following algorithms:**
  + Decision tree (use Weka's J48)
  + k-NN (use Weka's IBk algorithm and set k=3)
  + Naive Bayes
  + Backpropagation learning (Use Weka's MultilayerPerceptron)
  + An ensemble of the above combined with majority vote (Use Weka's Vote algorithm under "meta"; make sure to set it up for majority voting)
  + A stacking of the above with each one of them used in turn as the meta learner (you will get 4 different accuracies here)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data Set | DTree | k-nn | Naïve Bayes | BackProp | Majority Vote | Stacking w/ each as a meta Learner.  DTree, k-nn, NB,BP |
| Iris | 96 | 95 | 96 | 97 | 95.3 | 92, 93, 95, 95.3 |
| Voting | 96.3 | 92.6 | 90.1 | 94.7 | 94 | 94, 95.6, 96, 95.8 |
| WaveForm | 75.08 | 77.7 | 80 | 83.56 | 83 | Too Long |

* **Briefly discuss your findings.**

On the Iris data set, we had a range of about 2% from the lowest accuracy reported to the highest. The low was 95% and the high 97%. The majority vote didn’t appear to improve the accuracy in this case. Also, stacking the learning algorithm seemed to decrease accuracy and not improve it.

On the Voting data, there is a range of about 6% from the lowest accuracy reported to the highest. The low was 90% and the high 96%. The majority vote did seem to improve accuracy; it helped to bring up some of the lower counts. Also, stacking the learning algorithms seemed to improve accuracy as well. Stacking pulled the low counts up and settled close to the highest accuracy value of DTree at 96%. In other words, the stacking algorithms have an average accuracy of 95.3%, where the lowest individual score was 90.1%.

So it would seem that the answer to the question “does stacking or an ensemble of learning algorithms improve accuracy” is, it depends. In some cases it does and in other cases it doesn’t, it depends on the data.