Christopher Morgan

CS478 : Brother Christophe

Nov 16, 2013

Write a simple program to test the validity of the No Free Lunch. Your program should act on binary classification tasks over binary input spaces.

* Requirements for your program are as follows:
  + The user should be able to select the number I of input features. (For testing purposes, I would recommend choosing 3, or 4 at the most).
  + The user should be able to select the size M of the test set. You would then use N-M examples for training and M for testing (where N is 2^I).
  + Your program should interface with Weka, at least call the command line version of Weka from your code to run a classifier on a dataset.
  + The user should be able to select the name of the Weka classifier to use (remember that these are of the form 'class.name').
  + Your program should then run the selected classifier on all tasks (i.e., training sets) and test against the corresponding test set, and return the overall generalization performance (i.e., the sum of 'accuracy-50' across all tasks).
  + You may find this [code](http://dml.cs.byu.edu/~cgc/docs/mldm_tools/Assignments/NFLTest.py) (in python) useful. It has a number of functions to set up ARFF training and test sets for binary tasks.
* Run your program with a decision tree (J48), a back propagation learner (MultilayerPerceptron), Naive Bayes (NB), a majority learner (ZeroR). Record the generalization performance.

Number of Inputs == 3, Number of Tests == 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Sum Accuracy | Sum ‘accuracy-50’ | S.A. / NUM\_TASKS | S.A. – 50 / NUM\_TASKS |
| J48 | 13616.4352 | 816.4352 | 53.1892 | 3.1892 |
| MultiP |  |  |  |  |
| NB | 13433.6995 | 633.6995 | 52.4753886719 | 2.47538867187 |
| ZeroR | 13226.7039 | 426.7039 | 51.6668121094 | 1.66681210937 |

* Implement a simple method for a minority learner (i.e., one that always returns the least frequent class). Modify your code so that this learner can be used instead of Weka (you do not need to put things into ARFF files, but can do everything in RAM in this simple case).
* Run your program on your minority learner and record the generalization performance.
* Is the NFL verified in every case? How do you explain what happens with the majority and minority learners?