Assignment (1)

1 Download the dataset and import it into R

- **Download** the sonar dataset from the UCI Machine Learning Repository:

http://archive.ics.uci.edu/ml/datasets/Connectionist+Bench+%28Sonar%2C+Mines+vs.+Rocks%29

The values in the downloaded dataset are separated by comas. The last attribute (last column) is the class attribute. Import the dataset into R.

2 Train a C4.5classifier. Cross-validation

- **Construct** a C4.5decision tree using the entire dataset as the learning set. Experiment with the different parameters of the algorithm. Test the constructed classifier on the same dataset. **Calculate** the different classification evaluation measures(e.g., accuracy, mean error, precision,
- recall, f-score etc.).
- Training and testing a classifier on the same data leads to over-fitting and false results (too opti-mistic). One of most used testing setups is the k-fold cross-validation: the dataset is divided into k
- stratified folds (each fold has the same percentage of each of the classes as in the original dataset). The classifier is trained on k−1 folds and tested in the k-the fold. This is repeated k times (each fold serves once for testing) and the values of indicators are averaged. **Test** your C4.5classifier using a 10-fold cross-validation. **Compare** the values for the measures with the results obtained earlier.
- Note: See http://www.unt.edu/rss/class/Jon/Benchmarks/CrossValidation1_JDS_May2011.pdf for an overview of how to use cross-validation in R.

3 Train other classifier

- Similarly as for C4.5in Section 2, implement using R packages, train and test using a 10-fold
- cross-validation on the sonar dataset some of the most widely used classification algorithms:
- •Random Forest(see http://oz.berkeley.edu/users/breiman/Using_random_forests_V3.
- 1.pdf);
- •Support Vector Machines(SVM) http://www.jstatsoft.org/v15/i09/paper
- •Naive Bayes http://www-users.cs.york.ac.uk/~jc/teaching/arin/R_practical/
- •Neural Networks http://journal.r-project.org/archive/2010-1/RJournal_2010-1_Guenther+Fritsch.pdf

Implement and test two of the ensemble learning methods : bagging & boosting, using C4.5as the base classifier. Train and test, just like the above algorithms, using a 10-fold cross-validation. Compare the performances of bagging and boosting to those of the base classifier C4.

Assignment Cont'd

4 Test multiple algorithms on multiple datasets. Statistically significant results

- While comparing the different classification algorithms, as seen in Section 3, already gives an idea of their performances, it is impossible to accurately state the one algorithms is "better" than the other.

Two problems arise:

- **insufficient testing datasets:** some algorithms might be particularly adapted to certain types of datasets, while they perform modestly on other;
- **is the performance difference significant or just the hazardous?** the differences of performance (e.g., of accuracy) might show that an algorithm performs better than another, but the difference might just be a random effect, with no statistical significance.

Address the two issues.

- download and import other datasets from the UCI repository:
- Hepatitis http://archive.ics.uci.edu/ml/datasets/Hepatitis)
- Spect http://archive.ics.uci.edu/ml/datasets/SPECT+Heart
- Pima-indians http://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes

- For each pair (dataset ,algorithm) perform a 10 times 10 cross-validation.
- Present the results as a matrix, where the lines represent the datasets and the columns represent
- the algorithms (one matrix for each metric)

in order to detect if the metric differences are statistically significant:

- perform a Student's Paired T Test: all algorithms are compared two by two for each dataset.

- Which algorithm performs statistically significantly better overall(has the most wins)?
- Is there a clear and only winner?
- Repeat this analysis on multiple measures(accuracy, precision, recall,f-score).
- Interpret the results.

Note: In order to obtain the evaluation metrics, repeat 10 times the 10-fold cross-validation. For each iteration (each of the 10 times) randomize the order of the individuals in the dataset, so that the folds do not have the same content between iterations. The values for the indicators are the averages over the 10 iterations