Practice sessions – Course 1 and Course 3

CART Trees and Random Forests - Jean-Michel POGGI

Master 2 Course in Statistics

Universidad de la República – Facultad de Ingeniería, Montevideo, Uruguay

February 2018

Guide for the practice sessions with the companion scenario, the documentation
`cran.r-project.org/web/packages/VSURF/index.html`
and the two articles:
`journal.r-project.org/archive/2015-2/genuer-poggi-tuleaumalot.pdf`
`hal-descartes.archives-ouvertes.fr/hal-01387654v2`

1. Data

1. Load the library **`kernlab`**
2. Load the dataset **`spam`** in R and build the *dataframes* of learning and test sets (the first will be used for designing trees, the second for evaluating errors)

2. CART trees

1. Load the library **`rpart`**
2. Compute the default tree provided by **`rpart`**
3. Build a tree of depth 1 (stump) and draw it
4. Examine splits primary splits and surrogate splits
5. Build a maximal tree and draw it
6. Draw the OOB errors of the Breiman’s sequence of the pruned subtrees of the maximal tree and interpret it
7. Find the best of them in the sense of an estimate given by the cross-validation prediction error
8. Compare the default tree of **`rpart`** with the one obtained by minimizing the prediction error. Same question with the one obtained by applying the 1 SE rule
9. Compare the errors of the different trees obtained, both in learning and in test

3. Random Forests

1. Load the library **`randomForest`**
2. Build a RF for *mtry=p* (unpruned bagging) and calculate the gain in terms of error with respect to a single tree
3. Build a default RF
4. Calculate an estimate of the prediction error and compare it to bagging
5. Study the evolution of the OOB error with respect to *ntree* using `do.trace`

4. Variable importance

1. Calculate the variable importance of the `spam` variables for the default RF
2. What are the most important variables?
3. Calculate the importance of `spam` variables for stumps RF
4. Illustrate the influence of the *mtry* parameter on the OOB error and on the VI

5. Variable selection using random forests

1. Load the library **`VSURF`**
2. Apply **`VSURF`** on a subset of 500 observations of the data table `spam.app`
3. Comment on the results of the different steps
4. Experiment with the parallel version based on the article on **`VSURF`**