STAT 770 Dec. 9 Lecture 28

Decision-Tree and Random Forest Methods

Reading and Topics for this lecture: **rpart** and **randomForest** software descriptions (posted to special Decision Tree module in ELMS) and CRAN package descriptions, plus the R Scripts for this class: and `IntXPred.RLog` and `RandomForests.RLog`. Wikipedia article on different kinds of random forest algorithms is also interesting.

(1) High-level discussion of random forests

(2) Script case-studies, of `rpart` and `randomForest`

Course Evaluation Website Now Open

- website `https://www.courseevalum.umd.edu`
- Please evaluate me, the course, and the text
- Evaluation period closes (late pm) Wednesday, Dec. 15, 2020

Random Forest Idea

- Grow *many* trees, on randomly sampled subsets of data, with splits at each stage based on a small random sample of $\underline{X}$ coordinates

- aggregate over many trees by averaging predictions from mini-tree prediction rules.

- look in Scripts for examples

Further Aspects of Random Forests

General idea is called "Bagging" for **bootstrap aggregation**

Random re-samples are made from existing data, and also from fitting options e.g., variables to allow for each split, generally just a few in the `randomForest` software **(feature bagging)**

Each fitted tree is used to predict and (equally weighted) to contribute to a vote for class-membership proportions, over all data.

Random Forest Idea, Summary

• Decision tree methods like `rpart` tend to overfit; the idea in random forests is that simpler analyses aggregated over many different data and feature combinations will not, and will perform better under cross-validation.

• consistency of classification can be proved, under some ideal conditions, viz.

Breiman L (2001). "Random Forests". Machine Learning. 45 (1): 532. doi:10.1023/A:1010933404324.

• Other kinds of voting methods involve unequal weighting based on performance of individual analyses. That is the idea behind boosting, that we discuss in our final class, next time.