Lab #5: Regression Trees and Random Forests

Biostat 216: Machine Learning in R for the Biomedical Sciences

Classification Trees

1. Load the `rpart` package
2. Read in the dementia data "dementia2.csv" into a data frame called `dat` (this is a doctored version of the dementia.csv dataset that you saw previously. If you want to see what was done to create this doctored data you can look at dataPrep.R
3. Generate a table of the elements in the `Dementia` variable
4. Fit a classification tree with `Dementia` as response and all other variables as predictors except for `MacCohort_kr`. Then run the `summary` command on the fitted tree.
5. Plot the tree and add text indicating the splits. Be careful to generate text of a legible size.
6. Output the decision branches of the tree
7. Set a random seed of 3
8. Select 75 rows at random to be the training set and call it `train`
9. Build a classification tree on the training data, plot it, and add text labels
10. Predict the classes of the test data based on the fitted tree to the training data
11. Generate a table of the predicted vs. true values in the test data and calculate the classification accuracy
12. This tree was grown to full depth and might be too variable. So now use cross-validation results from `rpart` to prune it. Use the `printcp` command to display results and `plotcp` to visualize.
13. Prune the tree to a good value of k based on the plot and plot the resulting tree
14. Test the pruned tree on the test data
15. Compare the classification accuracy of the pruned tree to the unpruned tree. What's going on?

Random Forests

15. Install the `randomForest` package
16. Load the `randomForest` package
17. Fit a random forest to the training data using the `randomForest` command

The out of bag (OOB) error estimate are a clever way that random forests gets honest error estimates (i.e. not over-fitted). The model reports that `mtry = 11` which is the number of variables randomly chosen at each split. There are 141 predictors so we could try all 141 possible values of `mtry`.

18. Try every possible value of `mtry` from 1 to 141 in a loop, record the accuracy results, and make a plot.
19. Make a plot of error rates in OOB vs. test set across values of `mtry`
20. Would you prefer to rely on OOB or test set error?
21. What do you think is a good choice for `mtry`?