Grading Assignment 3

Thomas Lumley

10/1/2018

Q1. Use rpart to fit and prune (if necessary) a tree predicting spam/non-spam from the common word counts in the wordmatrix matrix. Report the accuracy with a confusion matrix. Plot the fitted tree (without all the labels) and comment on its shape.

10 points for fitting the tree. 10 for 'either' noting that the crossvalidation error for the largest tree was the smallest, or for pruning the tree to an appropriate point using `prune`. The appropriate point is either the tree with the minimum cross-validation error or the smallest tree whose cross-validation error is within 1 standard error of the minimum: they should say which they are doing. If the crossvalidation error for the largest tree is the smallest but they don't point it out, 5 points. They could get the cross-validation error either from the table produced by `printcp` or from the plot produced by `plotcp` 5 for a confusion matrix. 5 for a plot of the tree. 5 for noting that it always branches the same way, and 3 for giving some sort of explanation.

Q2. For each common word in wordmatrix , compute the numbers and that give the number of occurrences in spam and non-spam messages respectively. The overall evidence provided by having this word in a message can be approximated by . A 'Naïve Bayes' classifier sums up the for every (common) word in the message to get an overall score for each message and then splits this at some threshold to get a classification. Construct a naive Bayes classifier and choose the threshold so the proportion of spam predicted is the same as the proportion observed. Report the accuracy with a confusion matrix (It's called naïve Bayes because it would be a Bayesian predictor if the words were all independently chosen, which they obviously won't be)

5 for computing both y_i and n_i, 5 for computing e_i, 5 for adding them up. They can either add up e_i for each distinct word or count words multiple times if they appear multiple times. 5 points for choosing the cutoff so that (as closely as possible) the proportion of messages that are classified as spam is the same as the proportion that are spam. 5 for the confusion matrix. If they don't do the computations but instead use some naive Bayes classifier from a package they get 20/25 if the proportion of messages classified as spam is the same as the proportion that are spam, otherwise they get 15/25 if it looks like they've used the function as it was supposed to be used.

Q3. Read the description at the UCI archive of how the dataset was constructed. Why is spam/non-spam accuracy likely to be higher with this dataset than in real life? What can you say about the generalisability of the classifier to particular populations of text users?

Because the spam and non-spam messages came from different countries and user populations. 10 for anything sensible along these lines. An extra 5 for either noticing from the data or saying from prior knowledge that the Singapore English non-spam uses a lot of words that are unusual in British English (eg "lar")