Stat 462 Lab 8 solutions

March 14, 2014

Question 11.3

Get trees with sufficiently small c_p values that we can see the minimum cross-validated error. It turns out 0.002 is small enough. Note that I only used the covariates that weren't incomes from other years (74 and 75, specifically). If you used those incomes as predictors then your results will of course be different.

```
re78.rpart <- rpart( re78 ~ trt + age + educ + black + hisp + marr + nodeg
, data = nswpsid1, method = "anova", cp = 2e-3 )
printcp( re78.rpart )

Regression tree:
rpart(formula = re78 ~ trt + age + educ + black + hisp + marr +
    nodeg, data = nswpsid1, method = "anova", cp = 0.002)

Variables actually used in tree construction:
[1] age   black educ  marr  trt

Root node error: 6.8084e+11/2787 = 244291418

n= 2787

          CP nsplit rel error  xerror     xstd
1  0.1251577      0   1.00000 1.00039 0.045280
2  0.0578162      1   0.87484 0.87635 0.040160
3  0.0239012      2   0.81703 0.81849 0.039560
4  0.0174853      3   0.79312 0.80343 0.038483 *min + 1se*
5  0.0085582      4   0.77564 0.78288 0.038771
6  0.0064247      5   0.76708 0.77807 0.038941
7  0.0052595      6   0.76066 0.77932 0.039027
8  0.0045112      7   0.75540 0.77795 0.038852
9  0.0034441      8   0.75089 0.77836 0.038791
10 0.0029682      9   0.74744 0.77551 0.038932 *min*
11 0.0029452     10   0.74447 0.77670 0.038703
12 0.0027954     11   0.74153 0.77602 0.038779
13 0.0025590     12   0.73873 0.77799 0.038953
14 0.0020000     13   0.73617 0.78296 0.038991
```

This prediction has an error of $0.80343 \times 244291418 = 196271054$. Now let's fit a linear regression to the data instead.

```
re78.lm <- lm( re78 ~ trt + age + educ + black + hisp + marr + nodeg
                    , data = nswpsid1 )
summary(re78.lm)$sigma^2
```

The result is 187155210. The tree came close to matching the linear model, but still does not perform as well. In addition, modification of the linear model using suitable transformations and so on could very well improve the fit further.

Question 11.4

The data are in .csv format, without a header. They can be read in using

```
spam <- read.csv( "spambase.data", header = FALSE )
```

Now, we will get trees with sufficiently small c_p values that we can see the minimum cross-validated error. 10^{-4} is where I will start. Note that V58 is the final column of data containing whether the email was in fact spam.

```
library(rpart)
spam.rpart <- rpart( V58 ~ ., data = spam, method = "class", cp = 1e-4 )
printcp(spam.rpart)

Classification tree:
rpart(formula = V58 ~ ., data = spam, method = "class", cp = 1e-04)

Variables actually used in tree construction:
 [1] V12 V16 V18 V19 V2  V21 V22 V24 V25 V27 V28 V33 V36 V37 V45 V46 V49 V5
V50
[20] V52 V53 V55 V56 V57 V6  V7  V8

Root node error: 1813/4601 = 0.39404

n= 4601

           CP nsplit rel error  xerror     xstd
1  0.47655819      0   1.00000 1.00000 0.018282
2  0.14892443      1   0.52344 0.55929 0.015508
3  0.04302261      2   0.37452 0.46663 0.014493
4  0.03088803      4   0.28847 0.34253 0.012784
5  0.01047987      5   0.25758 0.28902 0.011885
6  0.00827358      6   0.24710 0.27027 0.011541
7  0.00717044      7   0.23883 0.26145 0.011373
8  0.00529509      8   0.23166 0.25483 0.011245
9  0.00441258     14   0.19581 0.23221 0.010787
10 0.00358522     15   0.19140 0.22780 0.010694
11 0.00275786     19   0.17705 0.22063 0.010541
12 0.00257400     22   0.16878 0.21677 0.010457
13 0.00220629     25   0.16106 0.21622 0.010445
14 0.00211436     27   0.15665 0.21456 0.010409 *min + 1se*
15 0.00165472     33   0.14396 0.21291 0.010372
16 0.00110314     36   0.13900 0.20463 0.010187 *min*
17 0.00082736     43   0.13127 0.20684 0.010237
18 0.00055157     47   0.12796 0.20684 0.010237
```

19	0.00036771	53	0.12466	0.20960	0.010299
20	0.00010000	62	0.12135	0.21346	0.010384

The min + 1 standard error rule gives us a c_p of 0.00212. This results in a cross-validation error estimate of $0.21456 \times 0.39404 = 0.0845$ or 8.45%. The previous cross-validation error was about 0.132 or 13.2%, so we are doing quite a lot better by using the extra covariates.