Lab6_Answer

Anyi Guo

14/11/2018

```
library(readr)
library(dplyr)
titanic3 <- "https://goo.gl/At238b" %>%
        read_csv %>% # read in the data
        select(survived, embarked, sex, sibsp, parch, fare) %>%
        mutate(embarked = factor(embarked), sex = factor(sex))
```

1) survived is a numeric value. We need to first transform it to a categorical value. Use titanic3$survived = as.factor(titanic3$survived) to do so.

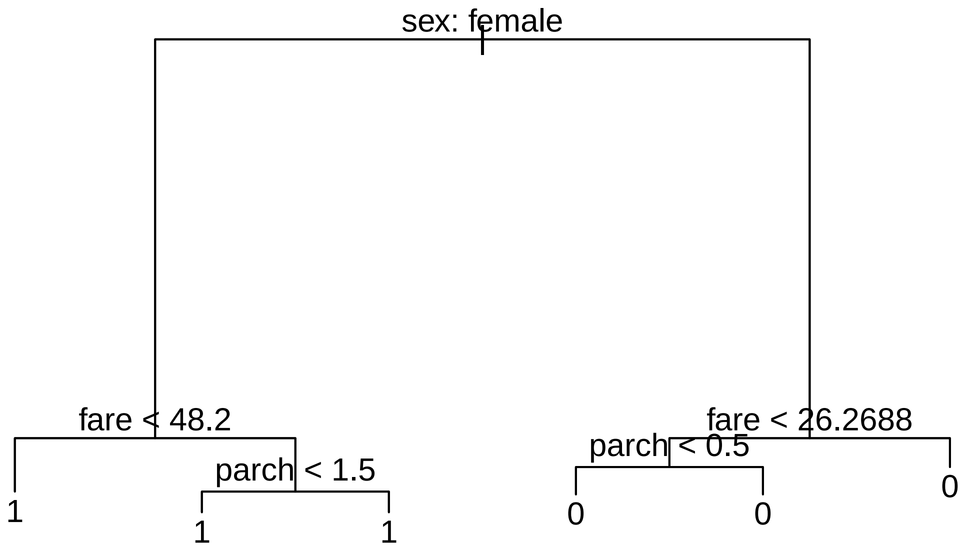
```
titanic3$survived = as.factor(titanic3$survived)
```

2) Fit a classification tree using all the observations. Find out which variables actually con- tribute to building this tree. Plot the tree.

```
library(tree)
tree.titanic<-tree(survived~.-survived,titanic3)
summary(tree.titanic)
```

```
##
## Classification tree:
## tree(formula = survived ~ . - survived, data = titanic3)
## Variables actually used in tree construction:
## [1] "sex"   "fare"  "parch"
## Number of terminal nodes:  6
## Residual mean deviance:  0.9582 = 1246 / 1300
## Misclassification error rate: 0.2205 = 288 / 1306
```

```
plot(tree.titanic)
text(tree.titanic,pretty=0)
```



Only three variables actually contribute to building this tree: `sex`, `fare` and `parch`.

3) Now we are going to estimate the test error:

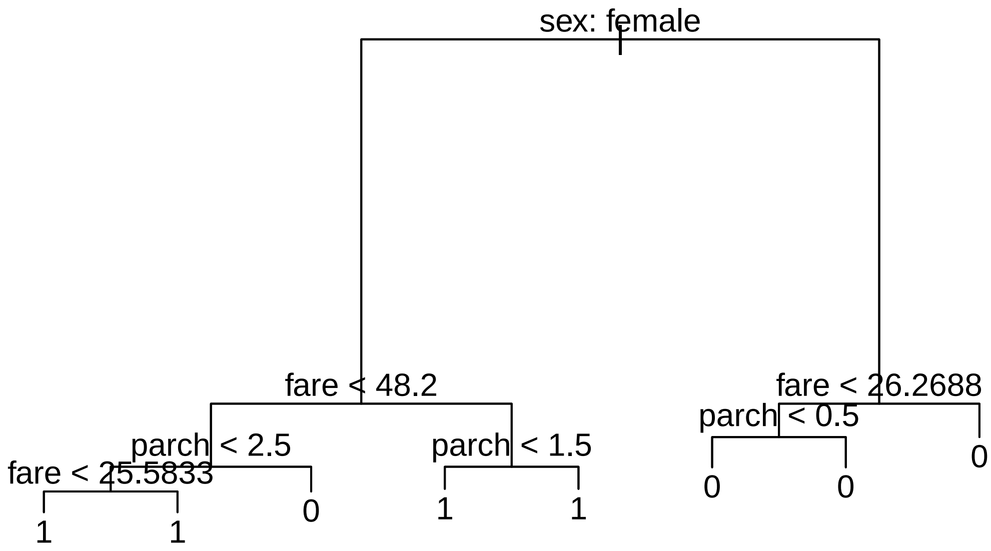
- a. Split the observations into a training set and a test set.
- b. Build the tree using the training set, and plot the tree.
- c. Evaluate its performance on the test data.

```
set.seed(2)
train<-sample(nrow(titanic3),nrow(titanic3)/2)
titanic3.test<-titanic3[-train,]
tree.titanic3.train<-tree(survived~.-survived,titanic3,subset=train)
tree.pred.test<-predict(tree.titanic3.train,titanic3.test,type="class")

table(tree.pred.test,titanic3.test$survived)
```

```
##
## tree.pred.test   0   1
##              0 347  85
##              1  61 162
```

```
plot(tree.titanic3.train)
text(tree.titanic3.train,pretty=0)
```



Error rate on the testing data is `(85+61)/655=0.2229`

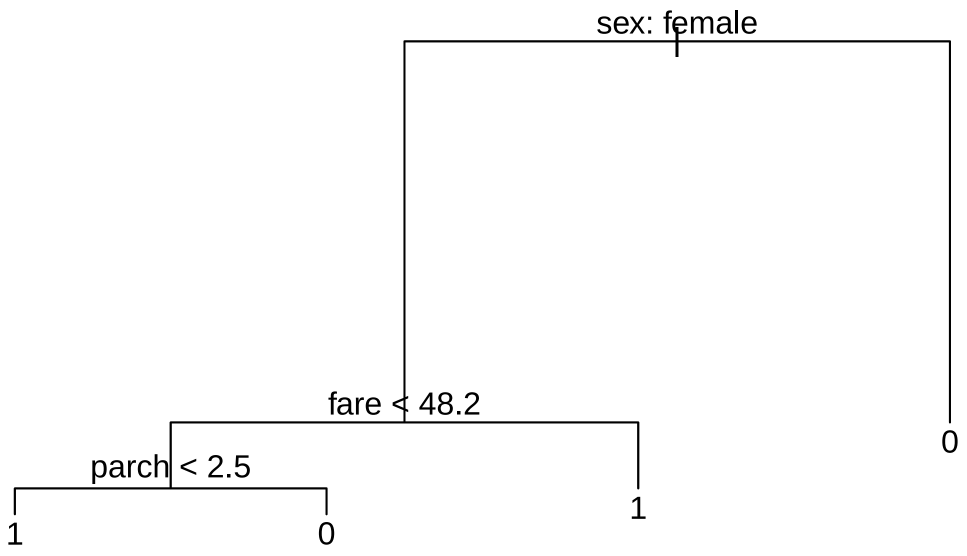
4) Next, let's find out whether pruning the tree might lead to improved results.

- a. Use cv.tree() to determine the optimal level of tree complexity.
- b. According to the result, do you think pruning is necessary? Why or why not?
- c. If you think it is necessary, or would like to give it a try, use prune.misclass() to prune the tree and evaluate the performance of the pruned tree.

```
set.seed(3)
# step 1: use cv.tree() to determine the optimal level of tree complexity
cv.titanic3<-cv.tree(tree.titanic3.train,FUN=prune.misclass)
cv.titanic3
```

```
## $size
## [1] 8 4 2 1
##
## $dev
## [1] 144 144 146 251
##
## $k
## [1] -Inf    0    3  106
##
## $method
## [1] "misclass"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

```
# step 2: use prune.misclass() to prune the tree
prune.titanic3 <- prune.misclass(tree.titanic3.train, best=4)
plot(prune.titanic3)
text(prune.titanic3,pretty=0)
```



```
# step 3: performance evaluation
tree.pred<-predict(prune.titanic3,titanic3.test,type="class")
table(tree.pred,titanic3.test$survived)
```

```
##
## tree.pred   0   1
##         0 347  85
##         1  61 162
```

Error rate on the testing data is `(85+61)/655=0.2229`

Based on the result, the error rate on the testing data is the same regardless of pruning. However, pruning the tree will increase interpretability, so we should prune the tree to 4 leaves.