Lab 06 - Classification Trees

[Sicily Xie #54385315]

2022-11-02

In the lectures last week, several classification methods were introduced: LDA, QDA, nearest neighbours and classification trees. The first two methods require the assumption of normality in the feature space, while the other two are nonparametric.

This week, we will fit a classification tree on the *Iris* flower data set used last week. The data set has two predictors, sepal and petal lengths, that are used to classify observations into one of the two flower species, *versicolor* and *virginica* (coded as 0 and 1, respectively).

```
library(tidyverse)
data("iris")
iris <- iris %>%
  filter(Species != "setosa") %>%
  select(contains("Length"), Species) %>%
  mutate(Species = fct_drop(Species)) # drop the unused setosa level
```

Questions

A classification tree separates the feature space into (hyperrectangular) regions, where the optimal split at each step is chosen to minimize a measure of homogeneity (Gini index or deviance). The majority class in each region is the predicted value.

We use the `rpart` function in the package of the same name to fit a classification tree. Use the control setting `myc <- rpart.control(minsplit=3,cp=1e-8)` and fit the tree using the following:

```
library(rpart)
myc <- rpart.control(minsplit=3, cp=1e-8)
set.seed(800)
```

```
# some code
model <- rpart(Species~Sepal.Length+Petal.Length, data = iris, method =
"class",control = myc, parms = list(split = "gini"))
model
```

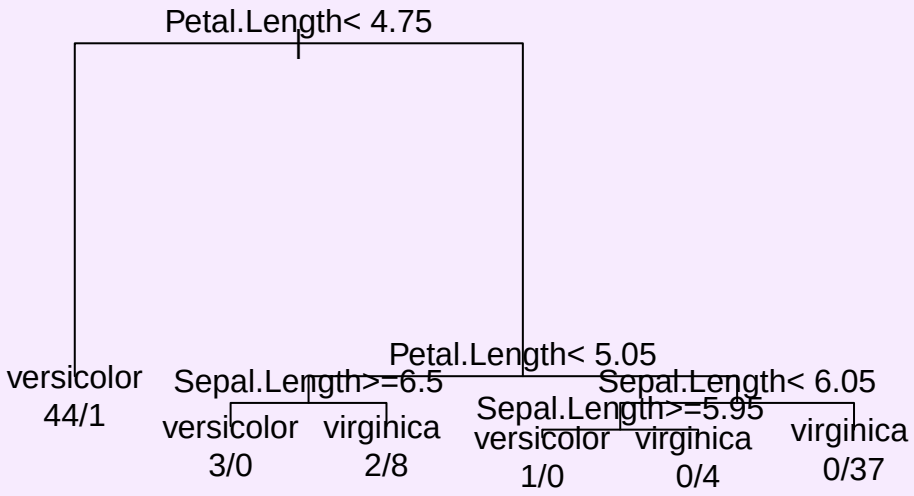
```
## n= 100
##
## node), split, n, loss, yval, (yprob)
```

```
##       * denotes terminal node
##
##  1) root 100 50 versicolor (0.50000000 0.50000000)
##    2) Petal.Length< 4.75 45  1 versicolor (0.97777778 0.02222222) *
##    3) Petal.Length>=4.75 55  6 virginica (0.10909091 0.89090909)
##      6) Petal.Length< 5.05 13  5 virginica (0.38461538 0.61538462)
##       12) Sepal.Length>=6.5 3  0 versicolor (1.00000000 0.00000000) *
##       13) Sepal.Length< 6.5 10  2 virginica (0.20000000 0.80000000) *
##      7) Petal.Length>=5.05 42  1 virginica (0.02380952 0.97619048)
##       14) Sepal.Length< 6.05 5  1 virginica (0.20000000 0.80000000)
##         28) Sepal.Length>=5.95 1  0 versicolor (1.00000000 0.00000000) *
##         29) Sepal.Length< 5.95 4  0 virginica (0.00000000 1.00000000) *
##       15) Sepal.Length>=6.05 37  0 virginica (0.00000000 1.00000000) *
```

The `control` parameter tells the function to use the settings given by `myc`, and `method='class'` specifies a classification (not regression) tree. The "`...`" part (model formula) is for you to fill in.

Q1 Use the `plot` and `text` functions to draw the fitted tree with labels. How many terminal nodes are there?

```
# some code
plot(model,margin = 0.1)
text(model,use.n = TRUE)
```



There are 6 terminal nodes.

Q2 Which of these [**terminal**] nodes is the most homogeneous, i.e., having the most even split between the two species? Write down the corresponding region of the feature space and the number of observations in each class within this region.

The terminal nodes that are the most homogeneous split are labelled as virginica, with a 20%/80% split. The feature space is 4.75 < petal length < 5.05, sepal length >= 6.5. The observations of Virginica is 8 while Versicolor is 2.

Q3 Suppose we have two new observations: $\boldsymbol{x}_1$ with (Sepal = 4.9, Petal = 5.3) and $\boldsymbol{x}_2$ with (Sepal = 5.9, Petal = 4.8). What are the predicted species based on this fitted tree?

```
# some code
x1 <- c(4.9,5.3)
x2 <- c(5.9,4.8)
newdata <- data.frame(rbind(x1,x2))
names(newdata)<-c("Sepal.Length","Petal.Length")
new_predict <-predict(model,newdata = newdata, type = "class")
new_predict
```

```
##        x1        x2
## virginica virginica
## Levels: versicolor virginica
```

The predicted specie for x1 is virginica, and the predicted specie for x2 is also virginica.

Q4

Is the misclassification rate (i.e., the proportion of observations wrongly classified) a good measure of the predictive ability of this tree? Why or why not?

No. Suppose we make the tree infinitely deep then we would guarantee to classify all our training data perfectly, it may over fits badly. Hence, checking only mis-classification rate would not be great.

Questions 5 and 6 below are optional; correct answers will earn credit if you do not score 100% in the preceding questions.

Q5 Find the observations that are misclassified by this tree. What is the misclassification rate?

Hint: Use `predict` with appropriate arguments to obtain the fitted classes and compare them with the observed response.

```
# some code
pred <- predict(model, iris, type = "class")
new_data <- cbind(iris, pred)
data_mis <-filter(new_data, Species != pred)
data_mis
```

```
##    Sepal.Length Petal.Length    Species       pred
## 21          5.9          4.8 versicolor  virginica
## 23          6.3          4.9 versicolor  virginica
## 57          4.9          4.5  virginica versicolor
```

```
n_mis <- nrow(data_mis)
n_mis
```

```
## [1] 3
```

```
mis_rate <- n_mis/nrow(iris)
mis_rate
```

```
## [1] 0.03
```

There are 3 observations misclassified, and the misclassification rate is 0.03.

Q6 The `printcp()` function displays the cross-validated error rates (refer to the `xerror` column of the output). Based on the result, do you think a simpler tree (with fewer splits) may be more useful for prediction? Why?

```
printcp(model)
```

```
##
## Classification tree:
## rpart(formula = Species ~ Sepal.Length + Petal.Length, data = iris,
##     method = "class", parms = list(split = "gini"), control = myc)
##
## Variables actually used in tree construction:
## [1] Petal.Length Sepal.Length
##
## Root node error: 50/100 = 0.5
##
## n= 100
##
##        CP nsplit rel error xerror     xstd
## 1 8.6e-01      0      1.00   1.16 0.098712
## 2 3.0e-02      1      0.14   0.18 0.057236
## 3 1.0e-02      3      0.08   0.20 0.060000
## 4 1.0e-08      5      0.06   0.22 0.062578
```

We have 5 splits.