05: A toy machine learning example

Boris Veytsman

TUG18 knitr workshop; July 2018*

In this tutorial we use a very simple machine learning model: Recursive Partitioning and Regression Trees [2].

Chunk: setup

```
opts_chunk$set(fig.path="05_figures/")
knit_hooks$set(
    chunklabel=
        function(before, options, envir) {
            if(before && options$chunklabel)
                sprintf(
                    "\\chunklabel{%s}", options$label)
        })
opts_chunk$set(chunklabel=TRUE)
```

Chunk: libraries

```
library(tidyverse)        # The Swiss Army knife of data processing
library(ggthemes)         # A better look for plots
theme_set(theme_tufte())
library(rpart)            # Recursive partitioning and
                          # regression trees
library(rpart.plot)       # Plotting of rpart trees
set.seed(201807)          # Setting the seed
library(Hmisc)            # for the latex tables
```

We divide the iris dataset into training and testing parts:

Chunk: dataTransormation

```
iris <- as.tibble(iris)
```

There are 150 data points. We take 2/3 of them as training, and 1/3 as test:

Chunk: testVsTraining

```
train_index <- sample.int(nrow(iris),
                      2*nrow(iris)/3)
iris_train <- iris[train_index, ]
iris_test  <- iris[-train_index, ]
nrow(iris_train)

## [1] 100

nrow(iris_test)

## [1] 50
```

Now we train a model

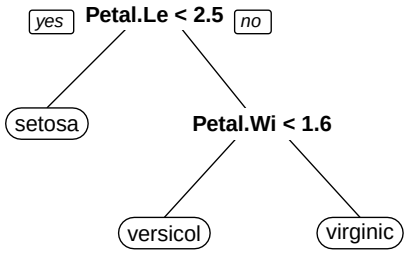
Chunk: model

```
model <- rpart(Species~ .,
               data=iris_train)
```

Here is the result:

Chunk: decisionTree

```
prp(model)
```



Note that model uses only petal data: sepal data are redundant!

Let us test the model. We calculate predicted species for the test model and the flag whether the prediction was correct:

Chunk: testing

```
iris_test <- iris_test %>%
    mutate(Species.Predicted=
               as.character(predict(model, iris_test,
                                    type='class')),
           Incorrect=Species != Species.Predicted)
iris_test %>% select(Species, Species.Predicted, Incorrect)
```

```
## # A tibble: 50 x 3
##    Species Species.Predicted Incorrect
##    <fct>   <chr>             <lgl>
##  1 setosa  setosa            FALSE
##  2 setosa  setosa            FALSE
##  3 setosa  setosa            FALSE
##  4 setosa  setosa            FALSE
##  5 setosa  setosa            FALSE
##  6 setosa  setosa            FALSE
##  7 setosa  setosa            FALSE
##  8 setosa  setosa            FALSE
##  9 setosa  setosa            FALSE
## 10 setosa  setosa            FALSE
## # ... with 40 more rows
```

The model was right 46 times out of 50, or 92%.
Confusion table:

Chunk: table

```
confusionTable <-
    table(iris_test %>% select(Species, Species.Predicted))
confusionTable
```

```
##             Species.Predicted
## Species      setosa versicolor virginica
##   setosa         16          0         0
##   versicolor      0         18         2
##   virginica       0          2        12
```

Table 1: Confusion table

confusionTable	setosa	versicolor	virginica
setosa	16	0	0
versicolor	0	18	2
virginica	0	2	12

Let us typeset confusion table using Hmisc [1].

Chunk: typesetTable

```
latex(confusionTable, file="", booktabs=TRUE,
     caption="Confusion table")
```

The option `results='asis'` is very important. Without it we get escaped code:

Chunk: typesetTable-no-asis

```
latex(confusionTable, file="", booktabs=TRUE,
     caption="Confusion table")

## %latex.default(confusionTable, file = "", booktabs = TRUE, caption = "Confusion table")%
## \begin{table}[!tbp]
## \caption{Confusion table\label{confusionTable}}
## \begin{center}
## \begin{tabular}{lrrr}
## \toprule
## \multicolumn{1}{l}{confusionTable}&\multicolumn{1}{c}{setosa}&\multicolumn{1}{c}{versicol
## \midrule
## setosa&$16$&$ 0$&$ 0$\tabularnewline
## versicolor&$ 0$&$18$&$ 2$\tabularnewline
## virginica&$ 0$&$ 2$&$12$\tabularnewline
## \bottomrule
## \end{tabular}\end{center}
## \end{table}
```

References

[1] Frank E Harrell, Jr. *Hmisc: Harrell Miscellaneous*, 2018. R package version 4.1-1; with contributions from Charles Dupont and many others.

[2] Terry Therneau and Beth Atkinson. *rpart: Recursive Partitioning and Regression Trees*, 2018. R package version 4.1-13.

Chunk: modelTestPetal

```
ggplot(iris_test) +
    geom_point(aes(Petal.Length, Petal.Width,
                   color=Species, size=Incorrect))
```

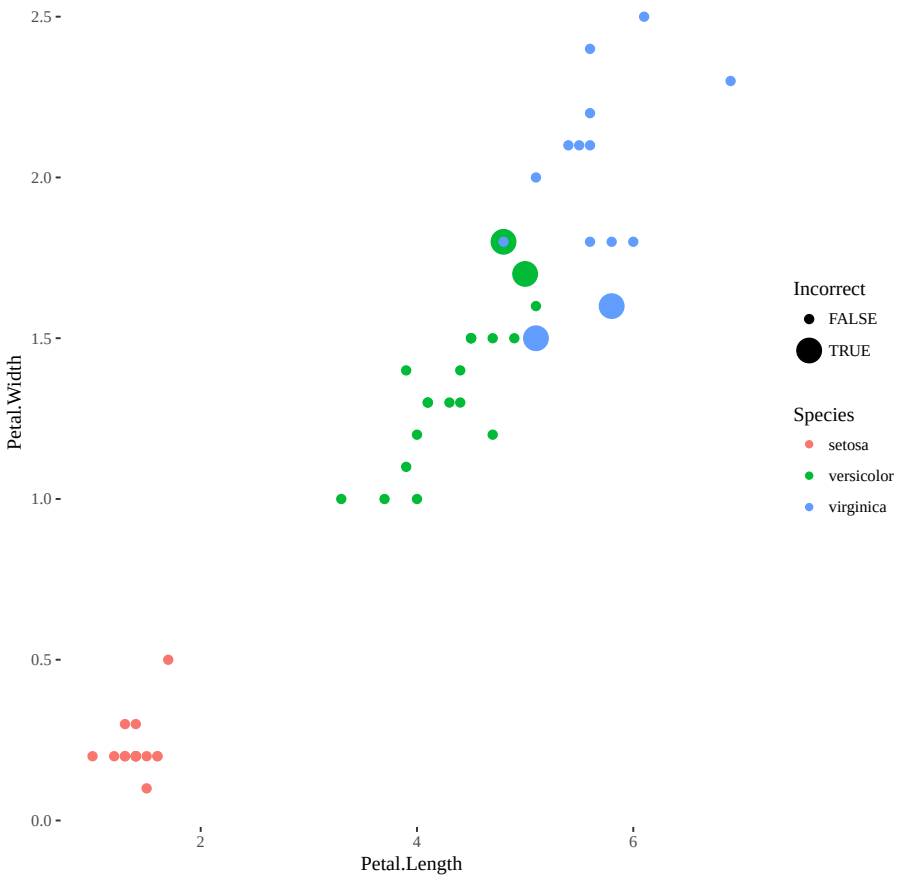


Figure 1: Testing the model, Petal space

Chunk: modelTestSepal

```
ggplot(iris_test) +
    geom_point(aes(Sepal.Length, Sepal.Width,
                   color=Species, size=Incorrect))
```

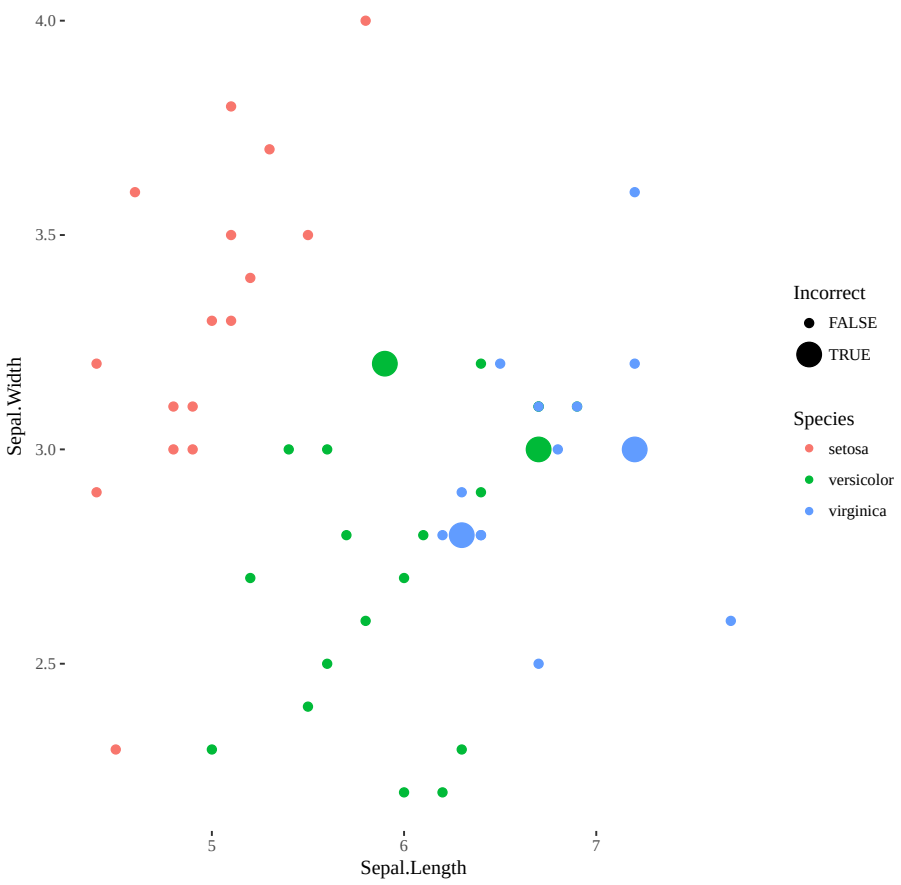


Figure 2: Testing the model, Sepal space