TUTORIAL 5

Aim:

In this tutorial you will cover:

1. **Reading data files into R**
2. **Exploratory analysis**
3. **Finding informative attributes**
4. **Fitting a decision tree for classification and interpreting the output**

Mode of study:

You can work through this tutorial in your own time, or in one of the assigned venues during the tutorial slot. Help is available during the tutorial slot in the venues.

Solutions will be made available in Week 6.

Before you start

If you have not worked through Practical 5 yet, you should do so before starting this tutorial.

DATASET

In the textbook on pages 56 – 62, an example concerning mushrooms is discussed. The dataset in question was obtained from the machine learning data repository at the University of California at Irvine. This repository is usually referred to as the UCI Machine Learning Repository. The link for the dataset used in the textbook is http://archive.ics.uci.edu/ml/datasets/Mushroom. You can visit this webpage to read more about the dataset, including the origin of the data, a description of the attributes and some academic papers that have been published utilising this dataset. The dataset can also be downloaded from the website; however, there is no need for you to do this, as I have already downloaded the dataset and uploaded it onto SUNLearn. The dataset is called `mushroom.csv`.

Note that the attributes are also listed on pages 57 – 58 of the textbook. The data itself is coded (for example for the `cap-shape` attribute, bell shape is coded as `b`). I have also uploaded a file for you describing the different attributes and the codes used. This file can also be found on SUNLearn and is called `mush_attributes.pdf`.

Remember the importance of background / domain knowledge when you work with data! So spend some time getting to know the data before you jump straight into the questions. Some of the questions will involve exploratory data analysis which will also help you in this regard.

It is up to you whether you create a markdown file for this practical or not, but it is highly recommended.

QUESTIONS

1. **Load the following packages:**
   `tidyverse, CORElearn, tree, rpart, rpart.plot`

2. **Read the `mushroom.csv` dataset into RStudio and store it in an object named `mushroom`.** Make sure that the attribute data types are correct once you have imported the data. If not, write some code to fix this.
   [Hint: Remember that if all of your columns are showing as characters instead of factors, a quick way to change this is to include the argument `stringsAsFactors = TRUE` in your call to the `read.csv()` function. This is a lot quicker than having to fix them each individually! (Obviously this should not be used if some of the attributes should actually be text / strings…)]

3. **Check the dimensions of your dataset.** You should find that the number of observations differ from the number of observations mentioned in the textbook for this dataset. Try to find the reason for this difference. [Hint: it involves missing values…]

4. Missing values in R are usually denoted by `NA`. You might have noticed that in this dataset, there are no values that are truly missing; however, in the `stalk-root` attribute, one of the categories is `?`, which (according to the description of this dataset) indicates a missing value. You could decide to leave this as is. Another alternative is to drop the `stalk-root` attribute from the dataset. The approach the textbook authors took, was to delete all observations with missing values for `stalk-root` from the dataset. We will therefore mimic their approach.
   Write code to remove all observations for which `stalk-root = ?` from the dataset.
   Check whether you now have 5 644 observations in your dataset.

5. **Check the data types of the different attributes.**

   [Hint: use a function to do this, don't check it manually.]

6. **Construct suitable plots to visualise each of the following attributes:**

   (a) **`odor`**

   (b) **`spore.print.color`**

   (c) **`population`**

   (d) **`odor and spore.print.color`**

   [Hint for (d): `odor` and `spore.print.color` are both categorical attributes. You can portray them both together in a *stacked bar chart*. In a stacked bar chart, the bars are partitioned according to a second attribute. This is accomplished in `ggplot` by including a fill aesthetic specifying the attribute according to which you want to partition the bars. While by default this is done according to counts, it might make more sense to use scaled bars, to better capture proportions. This is accomplished by including a position argument within your `geom_bar()` function call, specifying `position = "fill"`.]

7. **Find the class sizes of the target variable `edible`.**

8. **Create a frequency table of `edible` and `ring.number`.**

9. **S**uppose you wanted to construct a decision tree to predict whether mushrooms are poisonous or edible, using information gain as splitting criterion.

   Write code to determine what the first attribute to split on would be.

10. **Create a frequency table of `edible` and `odor`.**

11. Fit a classification tree to your mushroom data set, using the `tree()` function, and also plot the tree.

> **Tip:**
> In Practical 5, the splits were based on numerical attributes. In this tree, splits are based on categorical attributes and you might have noticed that the labels on your tree plot did not correspond to the text rules obtained. This is because by default, factor split attributes are presented by 'a', 'b', 'c', 'd', …, 'z', etc. if the tree is plotted. To make sure that the factor level names that are used in your plot correspond to the actual values, include the argument `pretty = 0` in your `text()` function call.

12. Fit another classification tree to your mushroom data set, this time using the `rpart()` function. Make sure you specify the splitting criterion as information gain. (Check the help file for the `rpart()` function to see how to do this.) Plot this tree.