# Exercises part 2

Course: Data analysis and visualization using R

## Section 1. `apply` and its relatives

In this section you will encounter some exercises revolving around the different flavours of apply.

### Exercise 1.1.

On the course website, you will find file whale_selenium.txt. You could download it into your working directory manually or use `download.file()`, but you can also read it directly using `read.table()` as shown here.

```
whale.selenium <- read.table(
    "http://www.bioinf.nl/~michiel/courses/R_minor/data/whale_selenium.txt",
    header = T,
    row.names = 1)
head(whale.selenium)
```

```
##   liver.Se tooth.Se
## 1     6.23   140.16
## 2     6.79   133.32
## 3     7.92   135.34
## 4     8.02   127.82
## 5     9.34   108.67
## 6    10.00   146.22
```

**(a)** Report the means of both columns using apply().

```
## YOUR CODE HERE
```

**(b)** Report the standard deviation of both columns, again using apply.

```
## YOUR CODE HERE
```

**(c)** Report the standard error of the mean of both columns, again using apply. The SEM is calculated as

$$\frac{s}{\sqrt{n}}$$

where $s$ is the sample standard deviation and $n$ the number of measurements. You should create the function calculating this statistic yourself.

```
## YOUR CODE HERE
```

**(d)** Using apply, calculate the ratio of $Se_{tooth}/Se_{liver}$ and attach it to the whale.selenium dataframe as column `ratio`. Create a histogram of this ratio.

```
## YOUR CODE HERE
```

**(e)** Using print and paste, report the mean and the standard deviationof the ratio column, but do this with an inline expression, e.g. an expression embedded in the Rmarkdown text.

**(f)** Using apply, calculate the ratio of $Se_{tooth}/Se_{liver}$ and attach it to the whale.selenium dataframe as column `ratio`. Create a histogram of this column.

```
## YOUR CODE HERE
```

### Exercise 1.2.

This exercise revolves around the `ChickWeight` dataset of the built-in `datasets` package.

**(a)** Use an R expression to report the number of chickens used in the experiment.

```
## YOUR CODE HERE
```

**(b)** Use `aggregate()` to get the mean weight of the chickens for the different Diets.

```
## YOUR CODE HERE
```

**(c)** Use `coplot()` to plot a panel with weight as function of Time, split over Diet.

```
## YOUR CODE HERE
```

**(d)**

Add a column called `weight.gain` to the dataframe holding values for the weight gain since the last measurement. Take special care with rows marking the boundaries between indivivdual chickens!

```
## YOUR CODE HERE
```

**(e)**

Split the `weight.gain` column on Diet and report the mean, median and standard deviation for each diet.

```
## YOUR CODE HERE
```

**(f)**

Create a (single-panel) boxplot for weight gain, split over Diet. Hint: read the `boxplot()` help page!

```
## YOUR CODE HERE
```

### Exercise 1.3.

The food constituents dataset on the course website holds information on ingredients for different foods. Individual foods are simply marked with an id.

**(a)** Report the different food categories.

```
## YOUR CODE HERE
```

**(b)** What is the mean energy content of chocolate foods?

```
## YOUR CODE HERE
```

**(c)** What is the food category with the highest mean fat content?

```
## YOUR CODE HERE
```

**(d)** What food category has the highest mean energy content, and which has the lowest?

```
## YOUR CODE HERE
```

**(e)** Create a boxplot showing the difference in sugar content between drink and solid food.

```
## YOUR CODE HERE
```

**(f)** Assuming both unsaturated fats and sugar are bad for you, what food category do you consider the worst? Thiink of a means to answer this, explain it and carry it out.

```
## YOUR CODE HERE
```

**Exercise 1.4.**

This exercise revisits the GOLDEN GATE AUDUBON SOCIETY dataset downloaded and prepared in the a previous exercise. I hope you still have the csv version of it. If it got lost, download it again. Again, open the file in Excel, replace all occurrences of the ";" character to "," and use "Save as.." to save it as ".csv" file (Comma-separated). Alternatively, download it from the course website. Load the dataset.

**(a)** Report the number of observations per `County`. Use both a textual as a barplot representation. With the barplot, you should order the bars according to observation numbers.

```
## YOUR CODE HERE
```

**(b)** Report the number of observations per `Observer.1`

```
## YOUR CODE HERE
```

**(c)** Report the species, using `Common.name`, for each genus.

```
## YOUR CODE HERE
```

**(d) Challenge!** Create a Dataframe holding the number of birds per day (use Date.start) and plot it with date on the x-axis and number of birds on the y-axis. Tip: use `as.Date()` to convert the character date to a real date field.

```
## YOUR CODE HERE
```