Exercises for the course

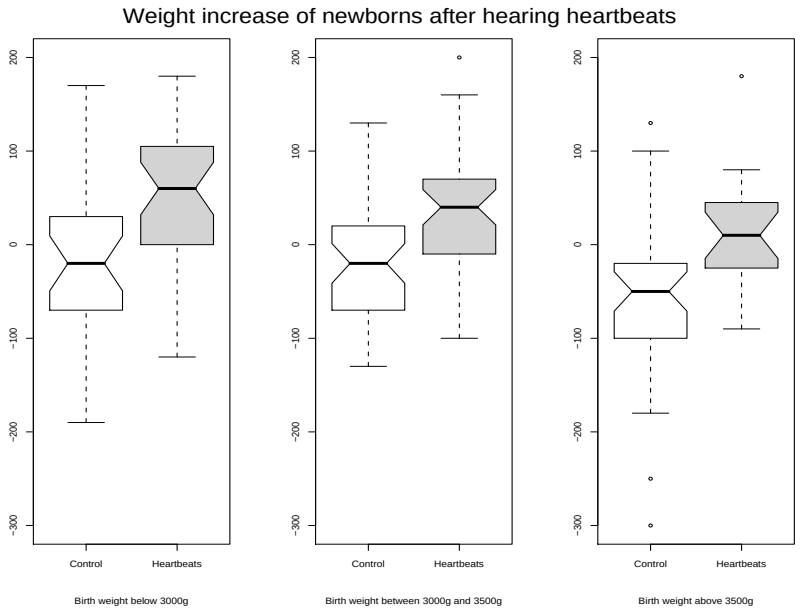
"An introduction to R"

Sheet 06

Exercise 31: Download `heartbeats.txt` and load it into `heartbeats`. Produce a picture which resembles the multi-figure below. Hints: One way to boxplot `wghtincr` as a function of `treatment` for each weight class is as follows. Split `heartbeats` according to `wghtcls` and denote the resulting list of data frames as `L`. Then use the command

```
boxplot(weight~treatment, data=L$'1',ylim=c(-300,200),
        col=c("white","lightgrey"), ...)
```

to produce the left boxplot and adapt the above command to produce the other boxplots. The option `ylim=c(-300,200)` ensures that all y-axes have the same range. Moreover the main title is magnified with factor 1.5. You can change the ratio of height and width of your multi-figure by using the mouse to change the plotting window.



Exercise 32: Write a function `stderr()` which calculates the standard error

$$\frac{\mathrm{sd}(x)}{\sqrt{\mathrm{length}(x)}}$$

of its argument `x`. What happens if you apply this function to `c(3,5,"a",7)` or to `c(3,NA,8,2)`? Improve the definition of `stderr()` as follows. Use `is.numeric()` to check whether the argument is numeric. If it is not numeric, then print the warning message `"Argument is not numeric: returning NA"` with the command `warning()` and return `NA`. Furthermore add an argument `na.rm` to the definition of your function and let its default value be `FALSE`. If that argument is `TRUE`, then remove all `NA`s from the argument vector and continue as before. Here is how it should work:

```
> stderr(c(3,5,"a",7))
[1] NA
Warning message:
In stderr(c(3, 5, "a", 7)) : Argmunt is not numeric: returning NA
> stderr(c(3,NA,8,2))
[1] NA
> stderr(c(3,NA,8,2),na.rm=TRUE)
[1] 1.855921
```

Exercise 33:
Download `ccrt.txt` and read the data into a variable. Why shouldn't you call this variable `ccrt`?

- Attach the data frame.
- Calculate the sample mean and the sample standard error for the two subvectors of `ccrt` corresponding to flies from Bangkok and Kathmandu, respectively.
- Now plot the density functions of two normal distributions which have the mean and standard error of the ccrt data of Bangkok and Kathmandu, respectively. How much do these densities overlap? Would you guess from this plot that the sample means are significantly different?
- Now apply the two sample t-test. The hypothesis you want to support is that flies from Bangkok and from Kathmandu have different chill coma recovery times.

Exercise 34:
Write a function `which.NA()` which returns the vector of indices at which the function argument has `NA`s. Here is how it should work:

```
> which.NA(c(1,2,NA,4,NA,6))
[1] 3 5
```

Hint: `is.na()`. Write a function `rm.NA()` which returns its argument without `NA`s.

```
> rm.NA(c(1,2,NA,4,NA,6))
[1] 1 2 4 6
```

Exercise 35: Write a function `sum.sq()` which calculates the sum of squares

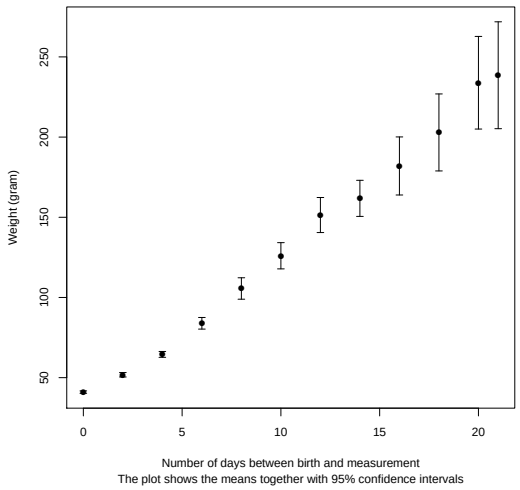
$$\sum_{i=1}^{\text{length(x)}} x_i^2$$

of its argument `x` with a loop.
Of course we know that `sum(x^2)` is way more efficient. Let us test this. Compare the run time of `sum.sq(x)` with the run time of `sum(x^2)` for $x \in \{1e3, 1e5, 1e7\}$. You get the runtime of a command by passing it as argument to `system.time()`.

Exercise 36: Recall the ChickWeight data from Exercise 26. Define a subvector `weight4` of `weight` corresponding to Diet 4. As in Exercise 26 calculate the mean of `weight4` for each day. In addition calculate the confidence intervals for these means. Recall that the 95%-confidence interval for the mean `mean(x)` of some vector `x` is given by

$$\Big[\texttt{mean(x)} - \frac{\texttt{sd(x)}}{\texttt{sqrt(length(x))}} * \texttt{qt}(0.975, \texttt{length(x)} - 1),$$
$$\texttt{mean(x)} + \frac{\texttt{sd(x)}}{\texttt{sqrt(length(x))}} * \texttt{qt}(0.975, \texttt{length(x)} - 1)\Big]$$

Represent the confidence intervals through vectors `top4` and `bot4` which contain upper and lower interval boundaries. Plot the vector of means and the confidence intervals using the command `errbar()` from the library `sfsmisc`. Your result should resemble the following figure.



Hint: The point character used for `errbar()` is 16 in this plot.