Exercise 1: Introduction to R

For this exercise use the file *placebo.csv*. A description of the contained data can be found in the file *placebo_des.txt*.

1.1 Start R and create a data frame called `data` using the `read.csv` function. You can do this in two ways: download the file or read directly from the URL. Use `?read.csv` to examine the parameters of the function and to decide which ones you should use.

1.2 Check the structure of the data frame using the `str` function. What kind of information was gathered during this study? Use the last three digits of your enrollment number as an id. List some information about the study participant with that id. Use different ways to index the data (e.g. column number, variable name, `$` operator).

1.3 It happens that the coding for `sex` is 1 for male and 2 for female. Convert this variable to a factor with labels "`M`" and "`F`". Use the functions `summary` and `str` on the variable `sex` to observe the changes caused by your type conversion.

1.4 Consult the description of the data and convert at least other two appropriate variables to factors. After your conversion check the output of the functions `summary` and `str` on your chosen variables to make sure that the type conversion was successful.

1.5 The `id` variable serves no purpose because it is the same as the row number. Remove it. Save the modified data frame as a file named *data.RData*. Remove the data frame from the workspace. Load the file *data.RData* and check if the loaded data frame has the expected structure.

1.6 Define a new function which calculates the mean value of the variable `time_d`. Verify your result with `mean(data$time_d)`. What information was measured with this variable?

1.7 Create a new data frame that contains the variables `time_d` and `tx`. Look up the description of the variable `tx` and give appropriate names to the variables in your created data frame. Use a `for` loop and an `if` statement to determine the mean of `time_d` depending on the levels of `tx`.

1.8 Use the functions `which` and `mean` to determine the mean of `time_d` depending on the levels of `tx`.

1.9 Use the function `boxplot(time_d~tx)` to create a boxplot of `time_d` depending on the levels of `tx`. Save the plot to a png file and include this file in your protocol.

2.0 Use the function `write.table` to write the data frame created in point 1.7 to a csv-file. Use `;` as a separator, include the column names but not the row names in the file and don't use quotes. Include the first 10 lines of the file in your protocol.