Basic R Commands

Getting familiar with R (Mac OSX)

Let us start things up by playing around with some variables. Remember that variables are often referred to as objects in R.

As this exercise uses several functions not covered in the lecture, you may want to use the help system to familiarize yourself with them. You do this by writing: help(function_name) or simply: ?function_name.

NOTE: Some people find it easier to write commands in a text editor and then paste them into R. The real advantage to this is that you can save the commands from nedit to the disk and redo the calculations another day. There is not much reason to use a text editor with this exercise, but you are welcome to try it out. You can find a text editor (TextEdit) under Applications.

You start R by pressing the R icon, R (Mac OSX) or by writing R in the X11 terminal. When you want to exit R again you write q().

1) Assign the value 12 to a and the value 5 to b.

2) Add the two variables together using the '+' operator.

3) Now try to add them together using the sum() function.

4) The function morm() can generate random numbers. Use it to create two random vectors x and y with 10000 numbers each. (hint: use morm(10000))

5) Use the str() function to verify that x and y indeed do contain ten thousand numbers without actually printing the two variables to the screen.

6) Plot the two vectors using the plot() function plot(x,y). Do they look random to you?

7) Use the var() and mean() functions to calculate the variance and the mean of the two vectors. Are they similar?

8) Use the t.test() function to statistically test whether they are similar. Look to the p-value, it's the odds they are from the same distribution (hint: use t.test(x,y))

9) Use ls() to get an overview of your objects

10) Generate a new y vector so that it no longer is similar to the x vector. Confirm the difference with the t.test() function (the p-value should become extremely small). Use the help system on morm() to figure out how to generate the new vector.

11) Try running the graphics demo (write: demo(graphics)), it's pretty and it'll give you an idea of R’s capabilities. It also lists some potentially useful stuff in the tables in the end.