

TW3421x - 2.3 - An Introduction to R

Hi there, welcome to this introduction to R, the statistical programming language.

We will see some basic commands of this new software (for some of you), and we will see how we can compute very basic quantities.

Starting from next week, we will use R to perform some of the computations that we need to assess credit risk.

So, starting from this week, I really hope you will invest some time in learning this new programming language.

You will see this is not difficult.

So, let's start.

I assume you have already installed R on your PCs, Macs or what you prefer.

If you launch R, this is what you see: the so-called R console.

The message you get is relative to the open source license of R. In my case it is in Italian.

If you launch R Studio, a very nice graphical interface for R, this is what you get.

For those of you that know Matlab, you will see a very familiar environment in R Studio.

I strongly suggest you to use this, if you are approaching R for the first time.

In R Studio you have the usual license text.

Moreover you have the possibility to click on the many tabs in the menu bar.

For example, let's clear the console, so that we have an empty page.

In R, commands are written after the "greater sign".

A command which starts with a hash sign is a comment, that is to say something that R does not execute.

Basic arithmetic is very simple: just type the operation as you would do with a calculator, then press "enter" or "return" on your keyboard.

The result immediately appears on screen.

Notice that R follows the usual rules about the order of computations. Hence always use brackets when needed.

You can also use simple functions such as the exponential, the natural logarithm, the logarithm with base 10, as well as notable constants like pi.

If you need to store a given quantity into a variable, because you need to use it over and over again, you simply type something like: `x=5`.

In this case R creates a variable called `x`, which contains the value 5.

To see the content of a variable, you can type the name of the variable and press "enter" or "return", or you can use functions like "print".

In R Studio, all the variables you have defined in the present session are given in the "Environment" space.

Another way of creating variables is using the arrow "`<-`". This is equivalent to the equal sign.

Once you have defined a variable, you can work with it in any operation.

Notice that R is case sensitive: capital `X` is different from small `x`.

With the function `c()` we can define vectors.

Notice that in typing `x` equal to `c(2, 3)` and so on, we have overwritten the old value of `x`. Now `x` is no longer equal to 5, but it is a vector.

We can apply many functions to a vector. For example "sum", to compute the sum of all the elements in the vector.

If we want to store this information in a new variable `s`, we can simply type `s=sum(x)`.

And we can also compute the mean, the variance, the standard deviation, the median, and so on.

When you apply some specific operations, like the power operation or the square root, to a vector, R performs this task element by element within the vector. Look at the examples on screen.

To select one specific element within a vector, we can use the square brackets.

`x[1]` selects the first element in `x`, that is 2.

`x[3]` selects the third one, that is 5.

R possesses many useful functions to create special vectors. For more details, I refer you to the R guide, which you can find on the course platform. In particular, spend some time reading Chapters 2 and 5.

The function "length" tells us the length of a vector, that is the number of elements it contains.

Sometimes it may be useful to substitute a single element within a vector.

In R it is a very simple task.

Just select the element you want to change, and assign the new value, as you can see on screen.

With vectors we can also perform very useful logical operations.

For example, this command checks for the values, within vector `u`, that are strictly smaller than 8.

This information can also be used to perform more complex tasks.

Can you see what I am doing now?

To delete a variable, we can use the command `remove`, or `rm()`.

Let's now create two new variables.

Plotting them together is very simple. The command is `"plot"`, and the names of the variables.

And changing the graphical properties of the plot is simple as well. You can find more details in the guide.

R is extremely powerful in probability and statistics.

Generating 100 observations from a standard Normal distribution, with mean 0 and standard deviation 1, is extremely simple. The command is `rnorm`.

Let's plot a histogram. Easy, isn't it?

If you prefer a Gaussian distribution with mean 5 and standard deviation 2, the command slightly changes. And here it is.

We can plot another histogram, we can compute the mean, the standard deviation, the median, and so on.

To read the help guide about the use of a specific function, just type `?` (question mark) and the name of the function.

For example: `?rnorm`, or `?mean`.

As you will see in the R manual, we can perform many other operations with vectors and variables.

For example, with vectors, we can sum them, we can multiply them element-wise, but we can also perform linear algebra computations.

Can you see what the last line of code is doing?

Ok. Now, it is your turn.

Try to familiarize with R, by solving the simple exercises you can find in the next pages.

And, most of all, spend some time reading the manual and playing with code.

See you next time.

Ciao.