

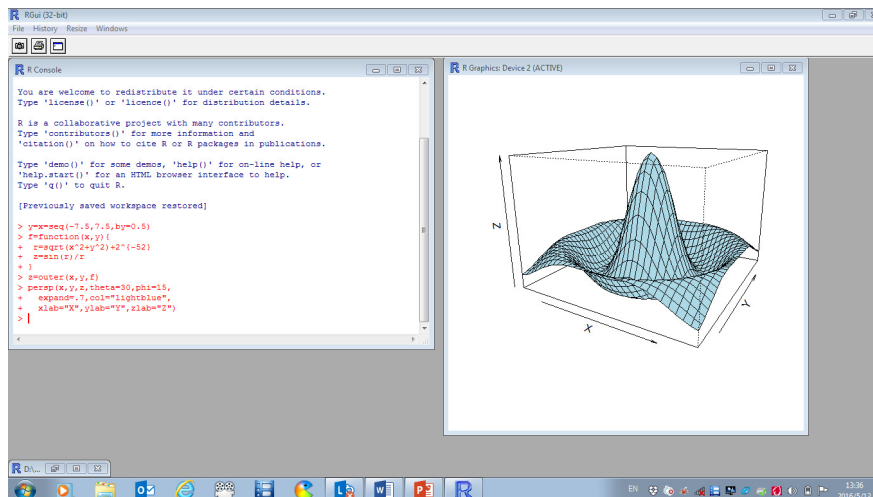
What is R?

R is a well developed software programming language which offers an environment for data manipulation, calculation and graphical display, particularly useful in statistical computation and graphics. It is supported by large number of user networks and can be downloaded free of charge from website <http://www.r-project.org>. It is a suite of operators for calculations on arrays (e.g. matrices) and has a large integrated collection of intermediate tools for interactive data analysis with user defined recursive functions and input/output facilities.

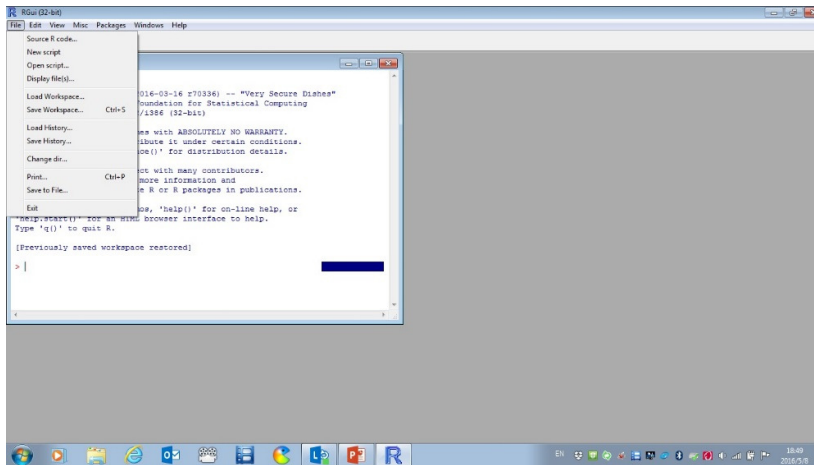
The R distribution contains functionality for large number of statistical procedures including but not limited to:

- Linear and generalized linear models
- Nonlinear regression models
- Classical parametric (such as independent/paired sample t -test) and nonparametric tests (such as chi square test)
- Time series analysis
- Clustering
- Smoothing

In addition, its flexible excellent graphical environment provided is interesting because it can create various kinds of data presentations such as the following example on 3-D graphic:



Upon downloading the software environment, we can start the work with R objects on the following workplace:



The R objects are:

- Names (case sensitive just like commands, never use command names as object names)
- Types of objects:
 - Vector (a list of numbers, such as `a <- c(1,2,3,4,5)` indicating command `c` creating a vector that is assigned to object "a")
 - Factor (a list of levels either numeric or string values)
 - Array / matrix (All columns must contain either numeric or string values, but these cannot be combined)
 - Data.frame (data frame `d` is converted into matrix `e` when we write: `e <- as.matrix(d)`)
 - List (containing a list of objects of possibly different types), *etc*
- Attributes: mode (numeric, character, complex, logical) with case sensitiveness, and length (number of elements in object)
- Creation: assigning a value or create a blank object

For convenience, it is advisable to open the "new script" from the tool bar to generate the computer codes first because they are editable before transferring to the first page workplace for completing the tasks. .



Below are some examples of simple R arithmetic and vector calculations as a calculator:

(A)

- `>`
- `> 5/7`
- `[1] 0.7142857`
- `> 5^3`
- `[1] 125`
- `> 5+7/3-12/4+6`
- `[1] 10.33333`
- `> 1+4*3^2`
- `[1] 37`
- `> (5+7)*3-(12/(6-4))`
- `[1] 30`
- `>`

(B)

- `>`
- `> x=c(1,2,3,4,5)`
- `> y=2`
- `> x*y`
- `[1] 2 4 6 8 10`
- `> x<-c(1,2,3,4,5)`
- `> y<-2`
- `> x*y`
- `[1] 2 4 6 8 10`
- `> x=c(1,2,3,4,5)`
- `> y=2`
- `> z=x+2*y`
- `> z`
- `[1] 5 6 7 8 9`
- `>`

(c)

- `>`
- `> k<-1:10`
- `> k`
- `[1] 1 2 3 4 5 6 7 8 9 10`
- `> 2*k`
- `[1] 2 4 6 8 10 12 14 16 18 20`
- `> k^2`
- `[1] 1 4 9 16 25 36 49 64 81 100`
- `>`
- `> j<-c(1:5)`
- `> j`
- `[1] 1 2 3 4 5`

- `> j<-1:5`
- `> j`
- `[1] 1 2 3 4 5`
- `>`
- `> log(1000, 10)`
- `[1] 3`
- `> sin(pi/3)^2 + cos(pi/3)^2`
- `[1] 1`
- `>`

It may be noted that generally the “<-“ indicating an assignment is preferred instead of “=” which is also a valid assignment.