

PARTE XI: Introduzione all'ambiente R

Perché imparare ad usare R?

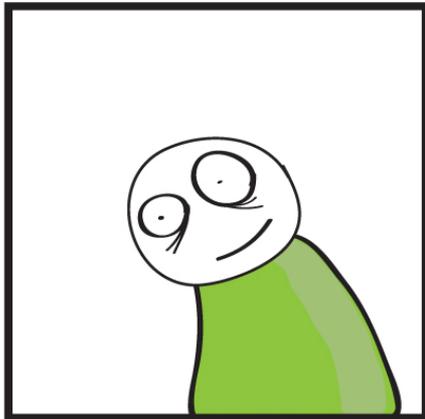
“Using R is a bit akin to smoking. The beginning is difficult, one may get headaches and even gag the first few times. But in the long run, it becomes pleasurable and even addictive. Yet, deep down, for those willing to be honest, there is something not fully healthy in it”.

Francois Pinard



La prospettiva del principiante...

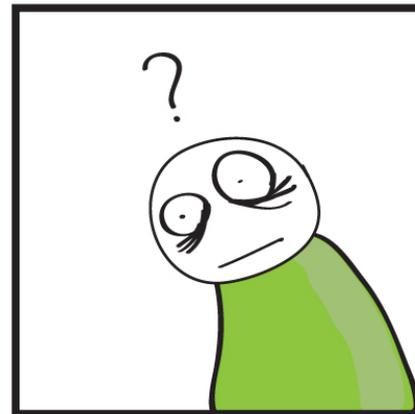
Learning to code



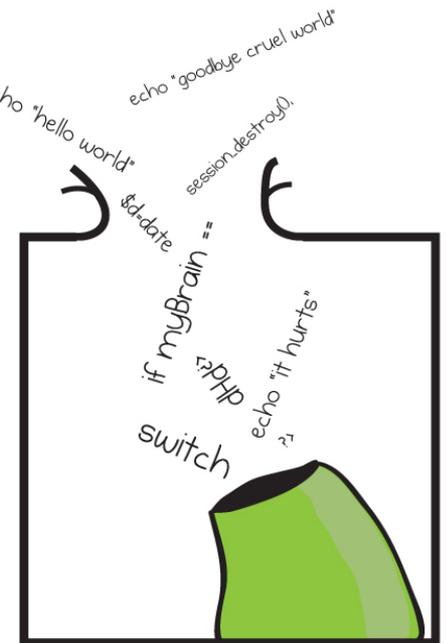
HTML



CSS



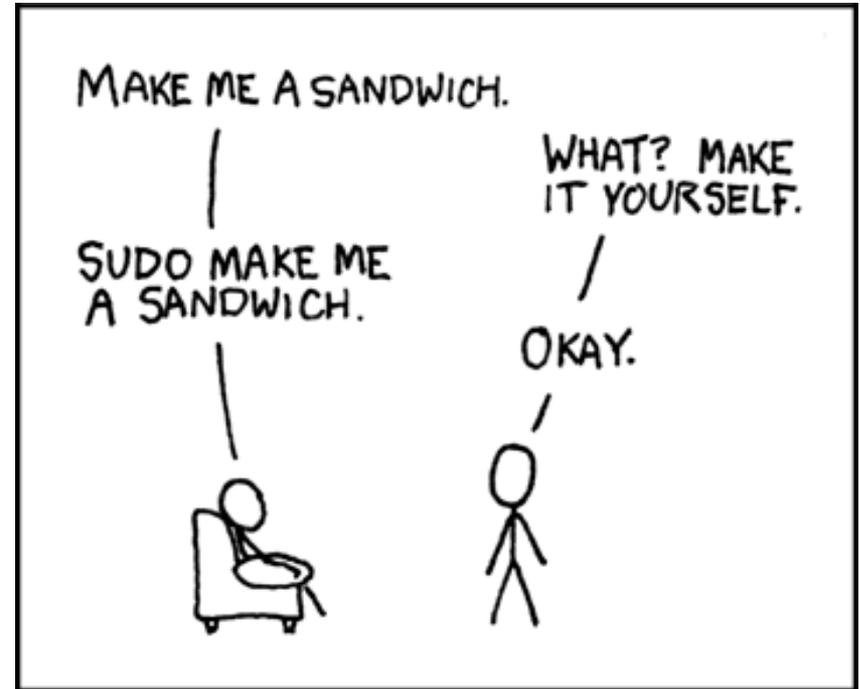
Javascript



PHP

```
if ($_SESSION['mybrain'] <= 2hours of sleep{  
  explode();}
```

Parlare in codice?



Cos'è R?

- R è un ambiente “open-source” orientato al trattamento, all'analisi ed alla visualizzazione dei dati (<http://cran.r-project.org>).
- 1976: nei Bell Labs nasce “S”, l'antenato di “S-plus”
- 1996 : introdotto da R. Ihaka e R. Gentleman, nasce R
- Oggi: R si è sviluppato enormemente. La sua community è stimata in milioni di utenti in tutto il mondo

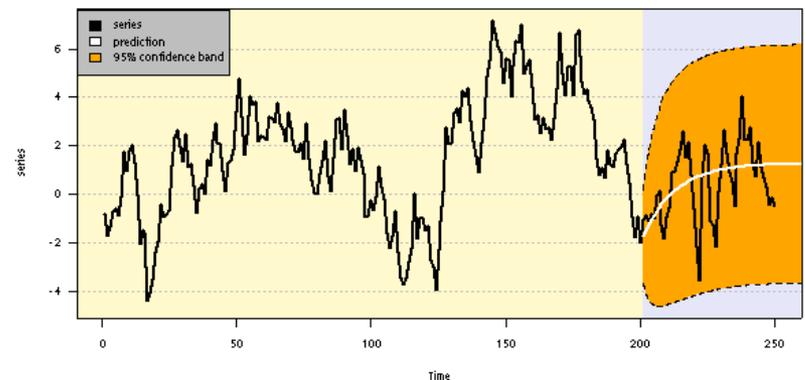
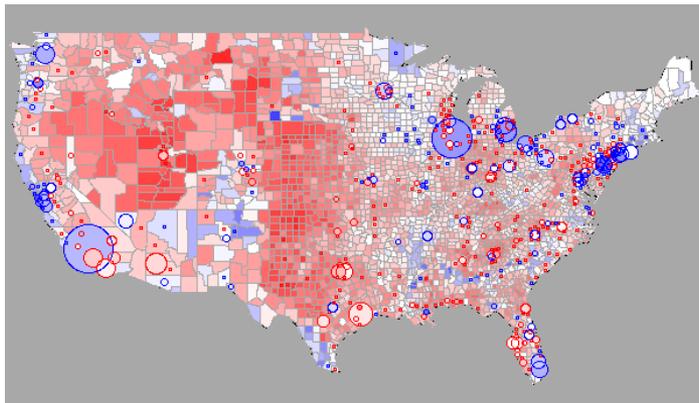
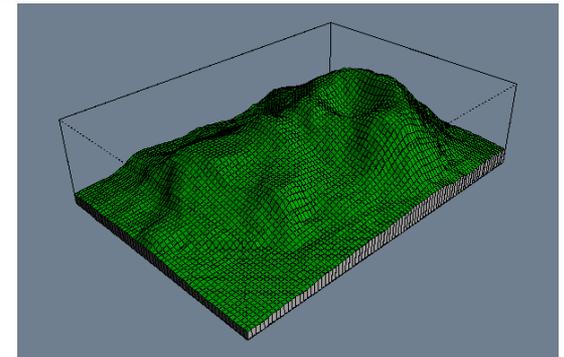
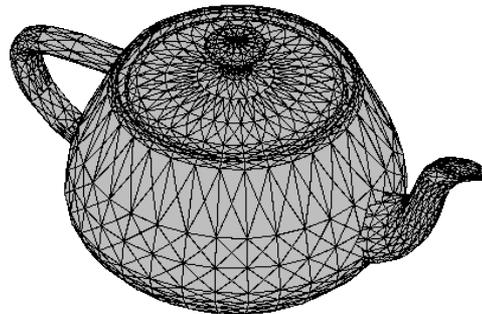
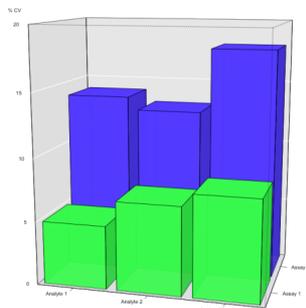
Cos'è R?

- R è un linguaggio interpretato.
- La maggior parte delle funzioni sono scritte in R, usando come base un insieme più piccolo di “funzioni primitive”
- E' comunque possibile interfacciare R con C, C+, o FORTRAN
- E' possibile eseguire comandi di sistema all'interno di R

Cos'è R?

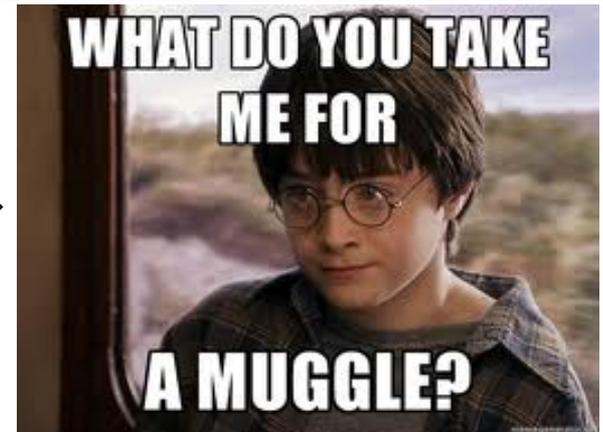
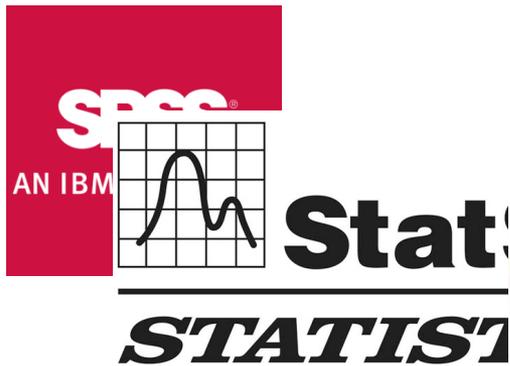
- R è “fatto” di operatori (+ - <- * %*% ...) per lavorare su vettori e matrici e un'ampia, coerente ed integrata libreria di funzioni (800+ pacchetti attualmente distribuiti)
- Fornisce la possibilità di produrre qualunque tipo di grafico, a qualsiasi livello di qualità

Grafici con R?



<http://gallery.r-enthusiasts.com/thumbs.php>

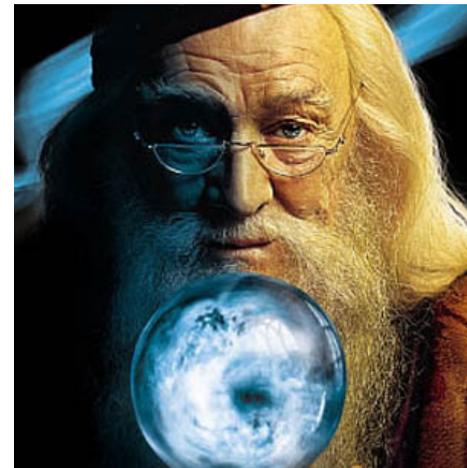
Perché R e non altro ?



Gli utenti di SPSS o PAST sono dei “**BABBANI**”:

- ▣ **non** hanno il potere di cambiare il loro ambiente;
- ▣ possono contare solo su algoritmi o funzioni sviluppati da altri;
- ▣ La maniera in cui approcciano ai loro casi di studio è “vincolata” dal software

Perché R e non altro ?



Gli utenti di R sono dei maghi:

- Possono sviluppare funzioni proprie;
- Non devono affrontare costi;
- Hanno bisogno solo di fare esperienza

“Open-source” significa gratis, giusto?

NO! Molto di più:

- Accesso totale agli algoritmi
- Possibilità di rimuovere bug o sviluppare ex-novo
- Un forum per la discussione e lo sviluppo
- E' comunque sviluppato da ~1000 esperti
- Promuove ricerche realmente riproducibile
- E' di facile accesso (la maggior parte di R è scritta.... in R!)

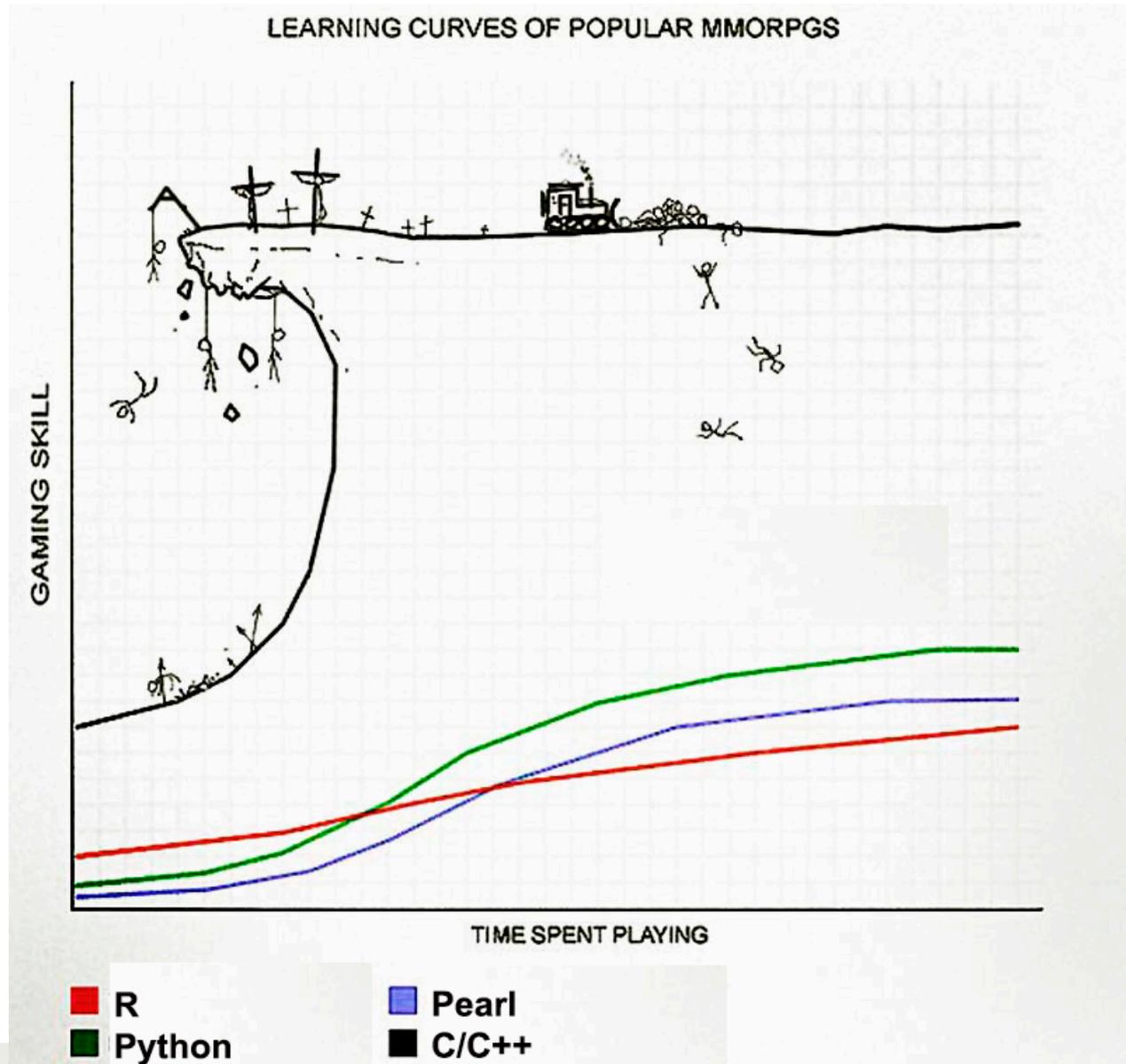
Vantaggi

- È gratis
- È aggiornato in tempo reale (SPSS, SAS e no!)
- 2nd solo a MATLAB per i grafici.
- Comunità vasta e attiva
- È eccellente per simulazioni e computazione intensiva
- Vi costringe a pensare alle vostre analisi
- È interfacciabile con software per l'archiviazione dei dati (SQL)
- C'è quasi sempre qualcuno che ha già fatto il VOSTRO lavoro!

Svantaggi

- Non è semplice da imparare
- Non ha un supporto commerciale: non potete prendervela con nessuno e questo può essere frustrante!
- È facile commettere errori
- Lavorare con dati di grandi dimensioni dipende dalla RAM
- La preparazione dei dati può essere complessa, lunga e foriera di errori
- L'help di R sembra "ostile"!!!!
- R è lento!

Linguaggi e curve di apprendimento...





Le 10 regole d'oro per imparare

1. E' possibile convincere te stesso a lavorare da riga di comando? ("Where there's a shell, there's a way")
2. Scegli un buon editor per gli script
3. L'errore è tuo amico e non ti deve spaventare
4. Chi ha detto che l'help di R è fatto male?
5. Google... questo amico stupefacente (e www.Rseek.org ancor di più....)!
6. Siate tignosi!

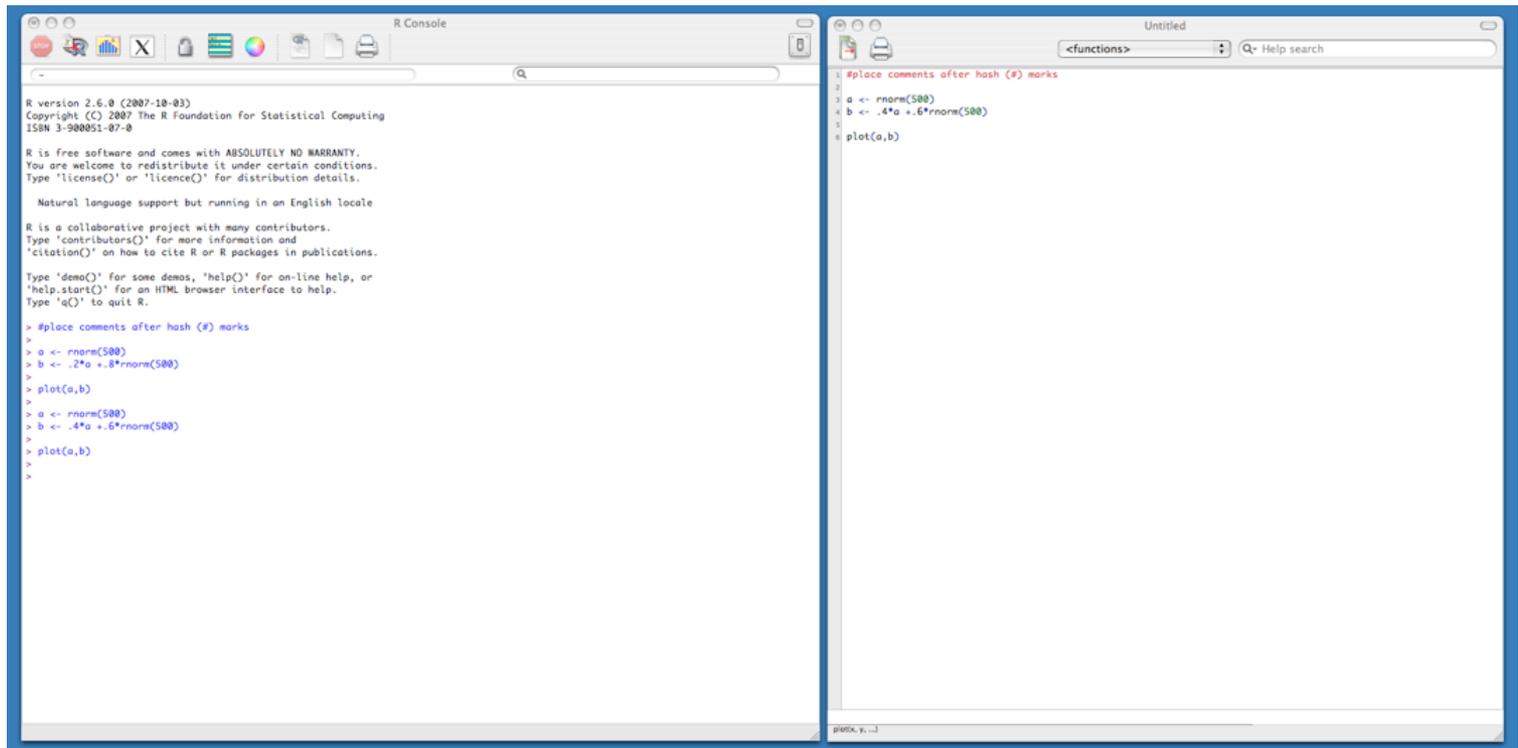


Le 10 regole d'oro per imparare

7. Cominciate fin da oggi a mettere da parte tutti i comandi che usate in uno script
8. Sforzatevi di usare un approccio simbolico che renda il vostro codice “generale”
9. Commentare SI, cancellare NO!
10. Ispezionate gli oggetti che maneggiate (di che classe sono? Come sono fatti? Ci sono dei NA?)

Facciamo conoscenza...

- Cominciamo con l'interfaccia classico, in cui avremo due finestre: [terminale](#) ed [editor](#)



The screenshot displays the R classic interface with two windows. The left window, titled 'R Console', shows the R version 2.6.0 (2007-10-03) and copyright information. It also displays the R license and a list of commands for help and quitting. The right window, titled 'Untitled', shows a script with the following code:

```
#place comments after hash (#) marks  
a <- rnorm(500)  
b <- .4*a +.6*rnorm(500)  
plot(a,b)
```

Facciamo conoscenza...

- L'utilizzo classico prevede una sessione interattiva....

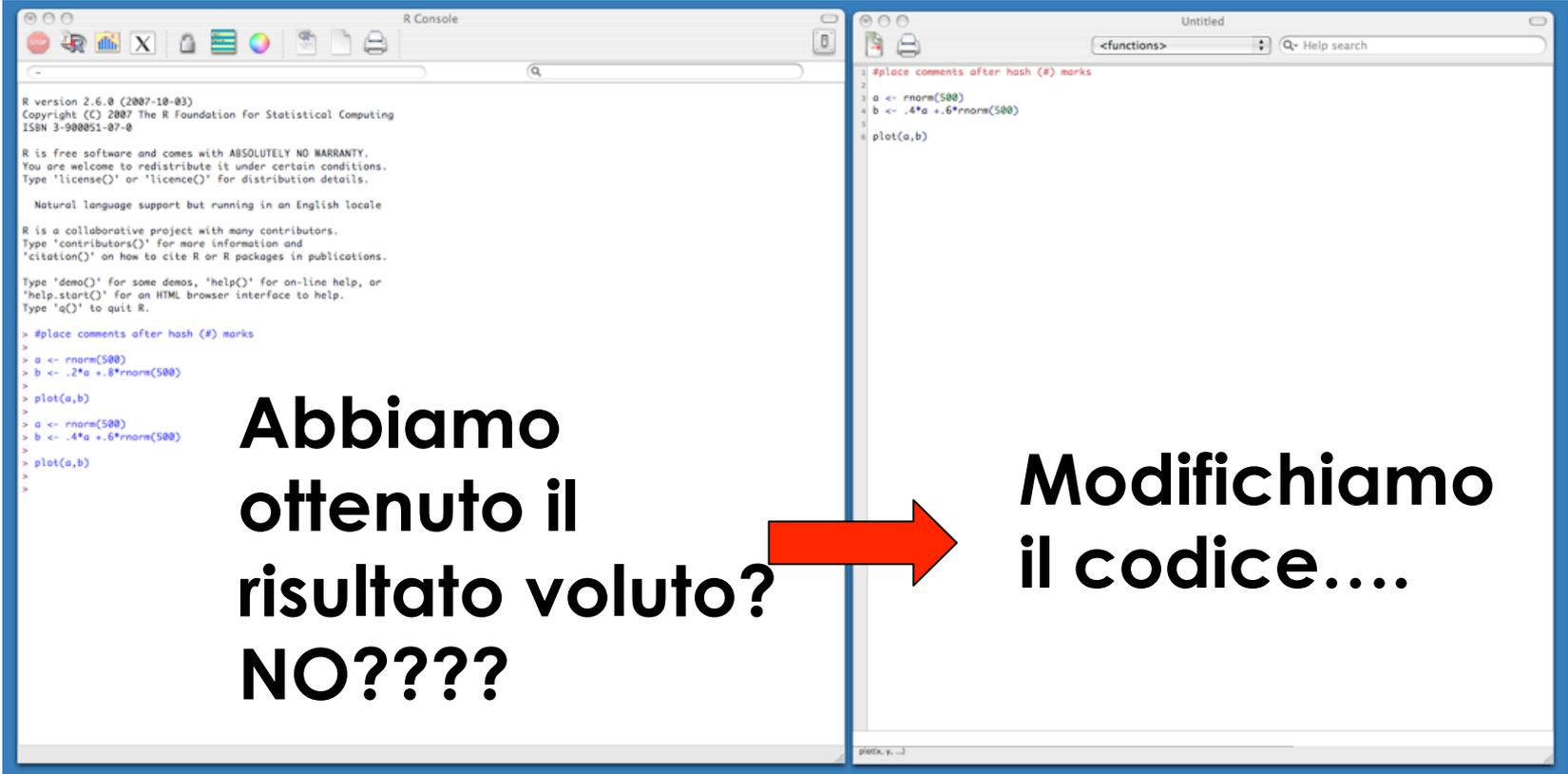
The image shows two windows from the R environment. The left window, titled 'R Console', displays the R version 2.6.0 startup screen and a series of commands and their outputs. The right window, titled 'Untitled', shows an R script editor with the same code as the console. A large red arrow points from the editor window towards the console window.

Terminale:
Caricamento dati,
esecuzione
comandi,
salvataggio output

Editor: scrittura
comandi/
codice/funzioni

Facciamo conoscenza...

- L'utilizzo classico prevede una sessione interattiva....



The image shows two windows from an R environment. The left window, titled 'R Console', displays the R version 2.6.0 startup screen and a series of commands and their outputs. The commands include generating random numbers and plotting them. The right window, titled 'Untitled', shows an R script editor with the same code as the console, but with a red arrow pointing from the console to the script editor, indicating a transition from execution to editing.

```
R version 2.6.0 (2007-10-03)
Copyright (C) 2007 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

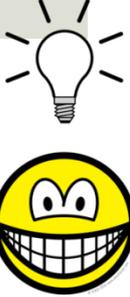
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> #place comments after hash (#) marks
>
> a <- rnorm(500)
> b <- .2*a +.8*rnorm(500)
> plot(a,b)
>
> a <- rnorm(500)
> b <- .4*a +.6*rnorm(500)
> plot(a,b)
>
>
```

Abbiamo ottenuto il risultato voluto? NO????

Modifichiamo il codice....

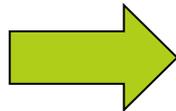


Oppure???

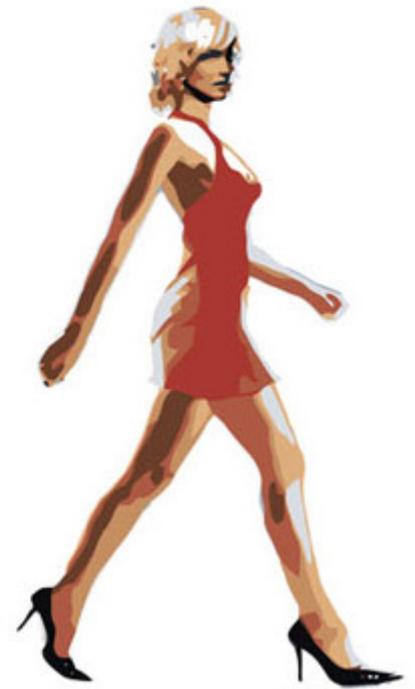
- Posso usare un editor che mi aiuta!!!

```
Tinn-R - [C:\Users\Utente\Documents\My Dropbox\Volanti\Simultaneo\Modello\Funzioni\boxplot3d.R]
File Project Edit Format Marks Insert Search Options Tools R View Window Web
R complex
boxplot3d.R
19 Xfloor[iS] = floorProjection$x
20 Yfloor[iS] = floorProjection$y
21 }
22 ### Plot
23 par(mai=c(2,2,1.5,1.5), cex=2)
24 if(plotRange==T){
25   plot(as.numeric(Xproj), as.numeric(Yproj), type='n', axes=F, xlab="", yla
26 } else {
27   plot(as.numeric(Xproj[2:4,]), as.numeric(Yproj[2:4,]), type='n', axes=F,
28 }
29 # Floor
30 if(plotFloor == T){
31   for(iX in 1:length(X)){
32     indici = which(Scenari[,1] == iX)
33     segments(Xfloor[iX], Yfloor[iX], Xfloor[indici], Yfloor[indici], col='li
34   }
35   for(iY in 1:length(Y)){
36     indici = which(Scenari[,2] == iY)
37     segments(Xfloor[which((Scenari[,1] == iY) & (Scenari[,2] == iY))], Yfloc
38 }
```

Oppure (2)?



Rstudio
www.rstudio.org



R studio

The image shows a screenshot of the RStudio application window. The interface is divided into several panes. The top-left pane is the Console, which is currently empty and has a dark blue background. The top-right pane is the Workspace, which is also empty. The bottom-right pane is the Files pane, showing a file explorer view of the user's home directory. The file explorer shows a list of files and folders with columns for Name, Size, and Modified. The files listed include .Rhistory, 20070724_data.xls, AutoHotkey.ahk, bak, client.rb, code, counts.xls, cuznsim_ss.xls, Default.rdp, Digsby Logs, docs, docs-archive, Downloads, eagle, ExpressPCB, facts-log.xls, FlowJo75.prefs, funding_ops__deadearly2007.txt, Geneious Backup 2011-04-27.zip, and Geneious Backup 2011-05-16.zip. The text 'workspace: your variables' and 'history: last command' is written in red in the Workspace pane. The text 'console (just like Rterm)' is written in red in the Console pane. The text 'files', 'plots', 'packages', and 'help' is written in red in the Files pane.

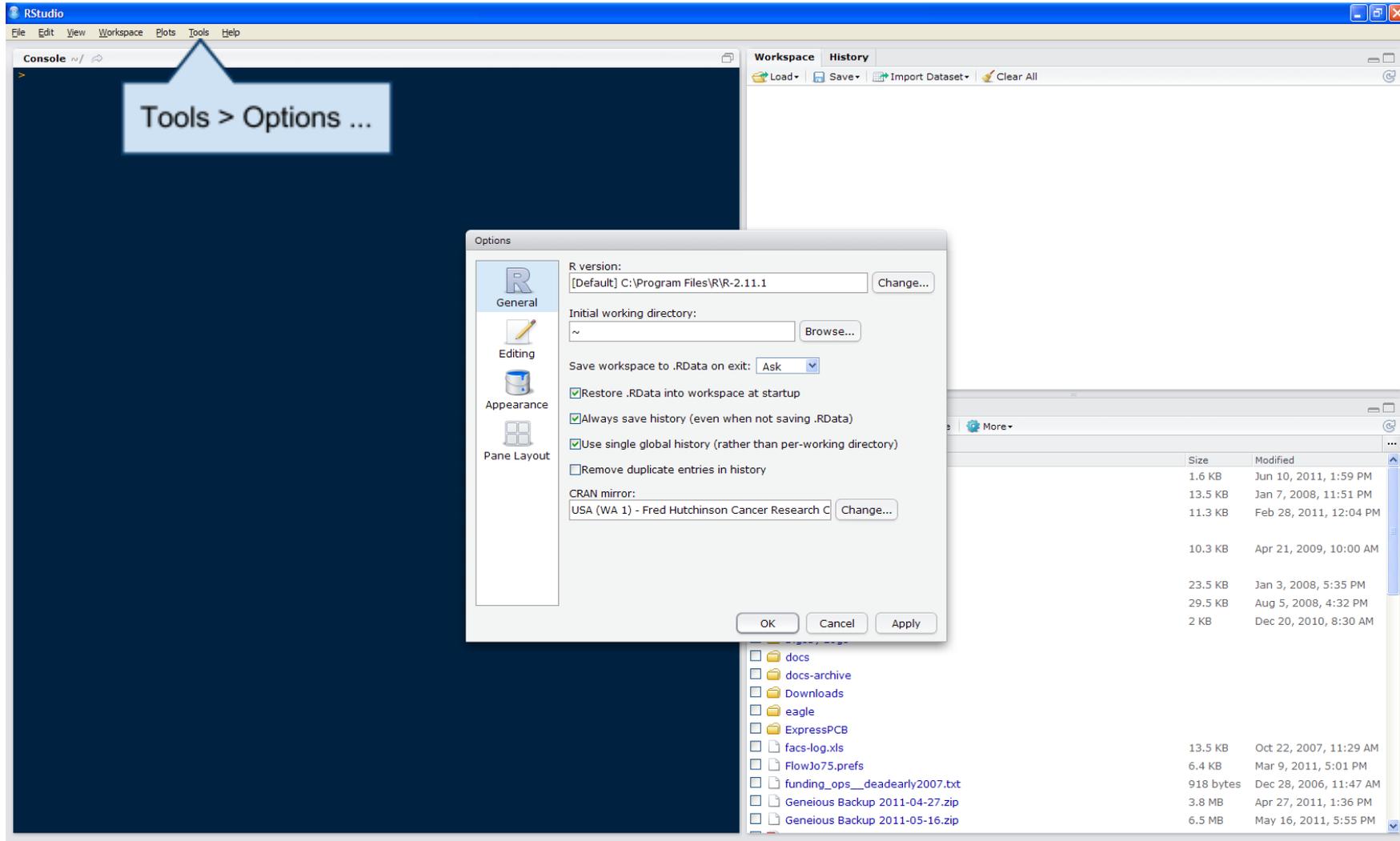
workspace: your variables
history: last command

console (just like Rterm)

files
plots
packages
help

Name	Size	Modified
.Rhistory	1.6 KB	Jun 10, 2011, 1:59 PM
20070724_data.xls	13.5 KB	Jan 7, 2008, 11:51 PM
AutoHotkey.ahk	11.3 KB	Feb 28, 2011, 12:04 PM
bak		
client.rb	10.3 KB	Apr 21, 2009, 10:00 AM
code		
counts.xls	23.5 KB	Jan 3, 2008, 5:35 PM
cuznsim_ss.xls	29.5 KB	Aug 5, 2008, 4:32 PM
Default.rdp	2 KB	Dec 20, 2010, 8:30 AM
Digsby Logs		
docs		
docs-archive		
Downloads		
eagle		
ExpressPCB		
facts-log.xls	13.5 KB	Oct 22, 2007, 11:29 AM
FlowJo75.prefs	6.4 KB	Mar 9, 2011, 5:01 PM
funding_ops__deadearly2007.txt	918 bytes	Dec 28, 2006, 11:47 AM
Geneious Backup 2011-04-27.zip	3.8 MB	Apr 27, 2011, 1:36 PM
Geneious Backup 2011-05-16.zip	6.5 MB	May 16, 2011, 5:55 PM

R studio



R studio

The screenshot displays the RStudio environment. The console window on the left shows the following output:

```
R version 2.15.0 (2012-03-30)
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: x86_64-pc-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[workspace loaded from ~/.RData]

> library("survival")
Loading required package: splines
> install.packages("GOSim")
Installing package(s) into 'C:/Users/cplaisie/Documents/R/win-library/2.15'
(as 'lib' is unspecified)
also installing the dependency 'igraph0'

trying URL 'http://cran.fhcrc.org/bin/windows/contrib/2.15/igraph0_0.5.5-2.zip'
Content type 'application/zip' length 5098030 bytes (4.9 MB)
opened URL
downloaded 4.9 Mb

trying URL 'http://cran.fhcrc.org/bin/windows/contrib/2.15/GOSim_1.2.7.2.zip'
Content type 'application/zip' length 492477 bytes (480 Kb)
opened URL
downloaded 480 kb

package 'igraph0' successfully unpacked and MD5 sums checked
package 'GOSim' successfully unpacked and MD5 sums checked

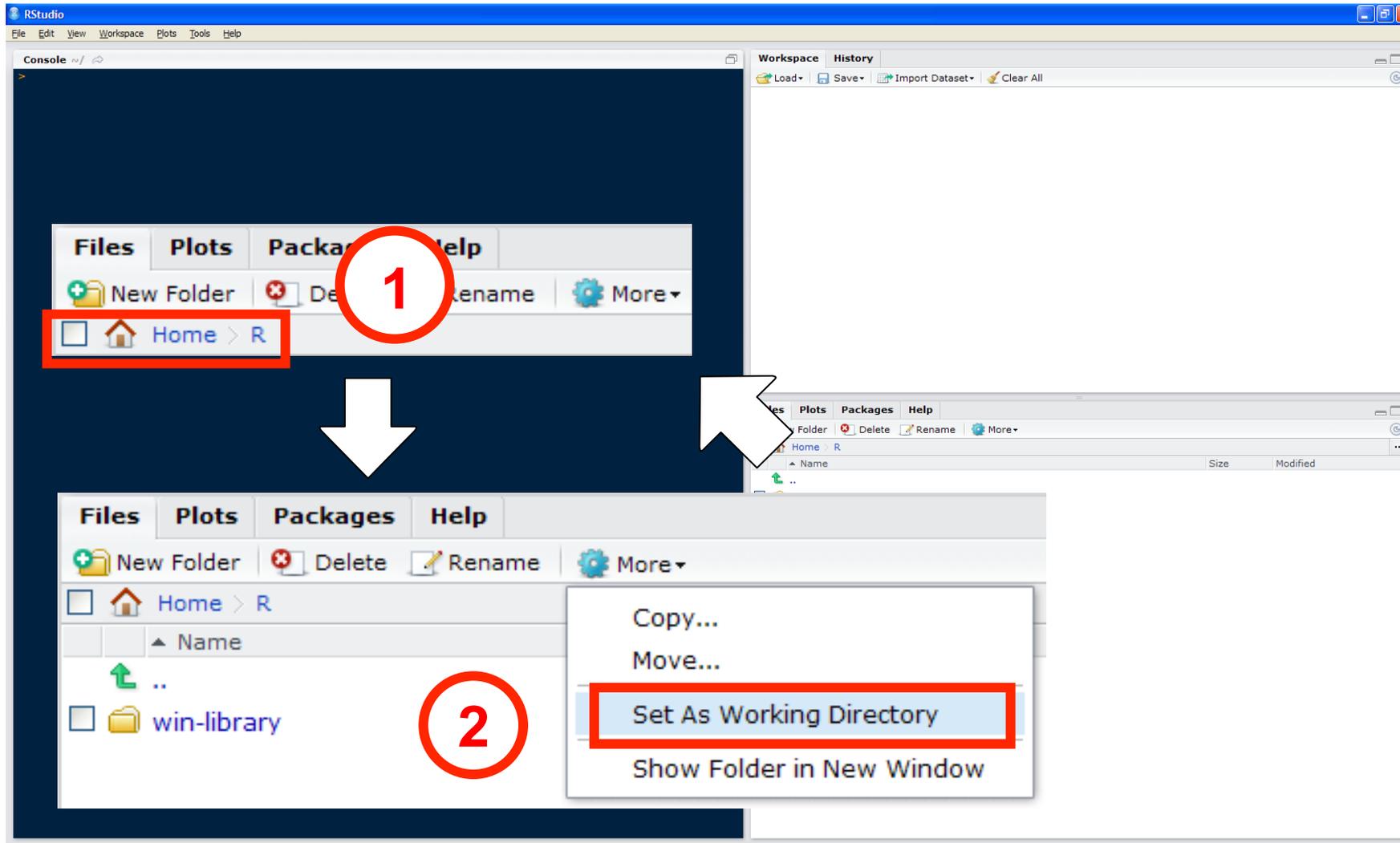
The downloaded binary packages are in
C:/Users/cplaisie/AppData/Local/Temp/RtmpqyehJM/downloaded_packages
> |
```

The Packages tab on the right shows a list of available packages:

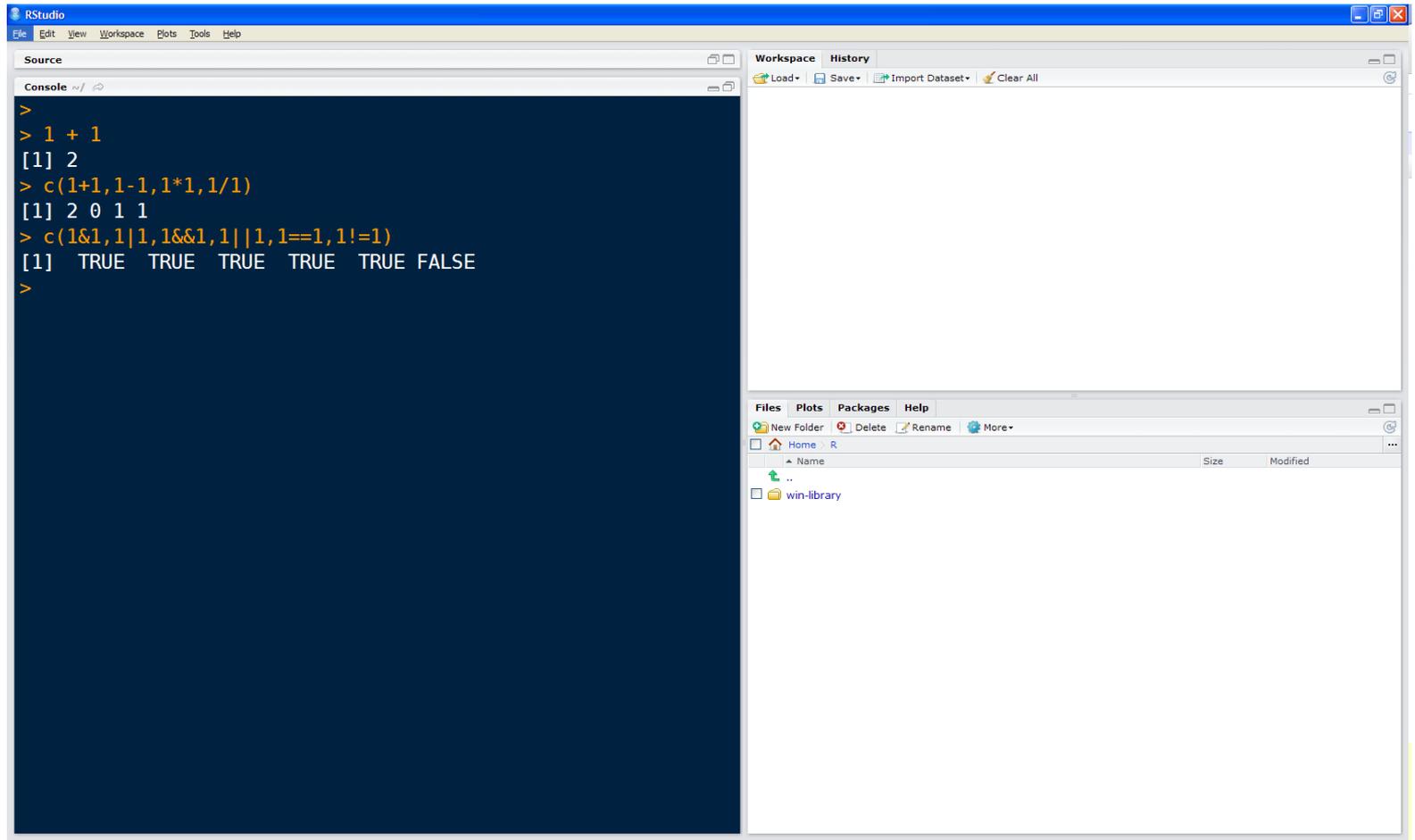
Package Name	Description
annotate	Annotation for microarrays
AnnotationDbi	Annotation Database Interface
Biobase	Biobase: Base functions for Bioconductor
BiocGenerics	Generic functions for Bioconductor
biomaRt	Interface to BioMart databases (e.g. Ensembl, COSMIC, Wormbase and Grmeme)
bitops	Functions for Bitwise operations
boot	Bootstrap Functions (originally by Angelo Canty for S)
class	Functions for Classification
cluster	Cluster Analysis Extended Rousseeuw et al.
codetools	Code Analysis Tools for R
compiler	The R Compiler Package
corpcor	Efficient Estimation of Covariance and (Partial) Correlation
datasets	The R Datasets Package
DBI	R Database Interface
flexmix	Flexible Mixture Modeling
foreign	Read Data Stored by Minitab, S, SAS, SPSS, Stata, Systat, dBase, ...
GO.db	A set of annotation maps describing the entire Gene Ontology
GOSim	Computation of functional similarities between GO terms and gene products; GO enrichment analysis
igraph	igraph: A package to handle graph data structures
graphics	The R Graphics Package
qrDevices	The R Graphics Devices and Support for Colours and Fonts

A green callout box with the text "Packages Tab" is overlaid on the Packages tab area.

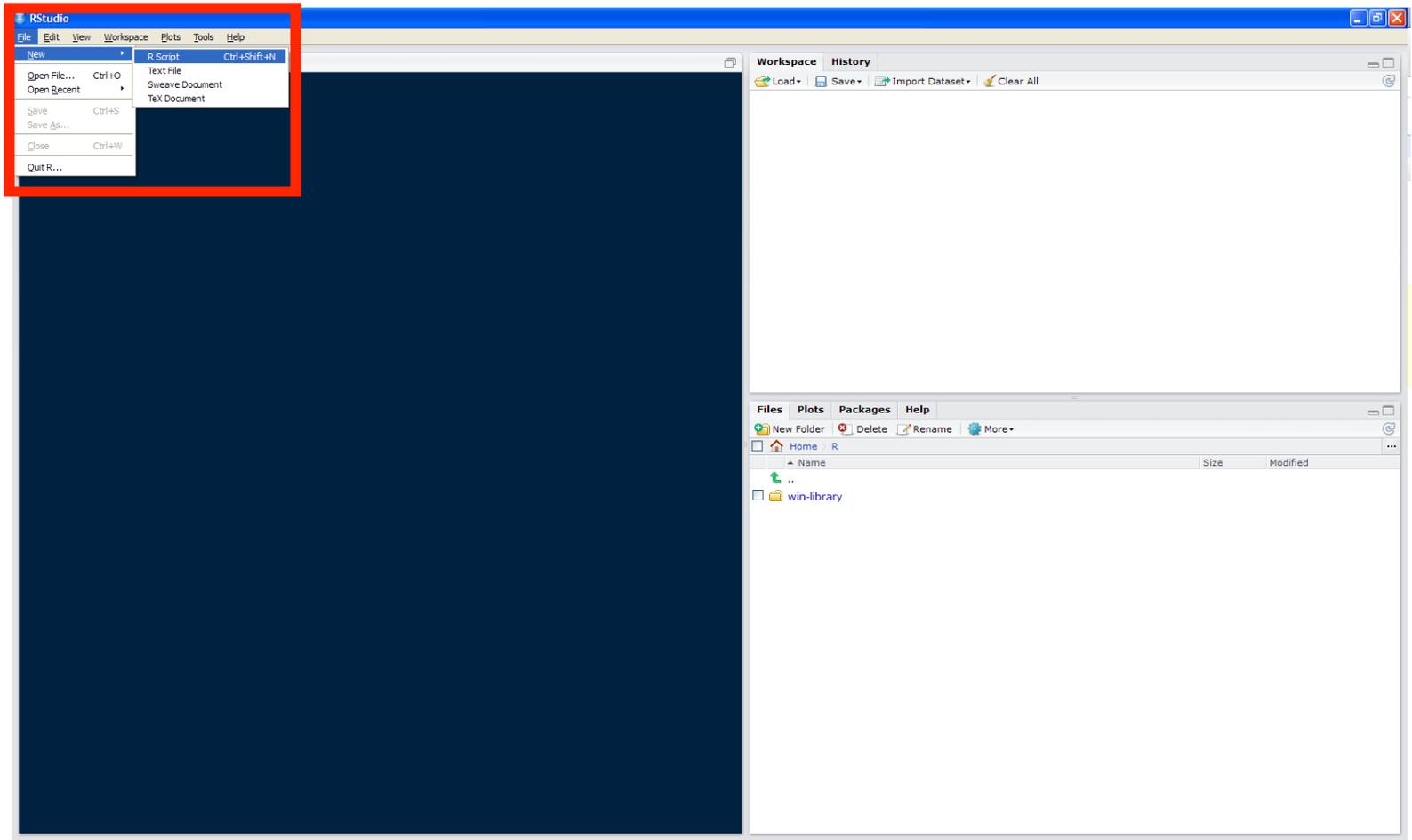
R studio



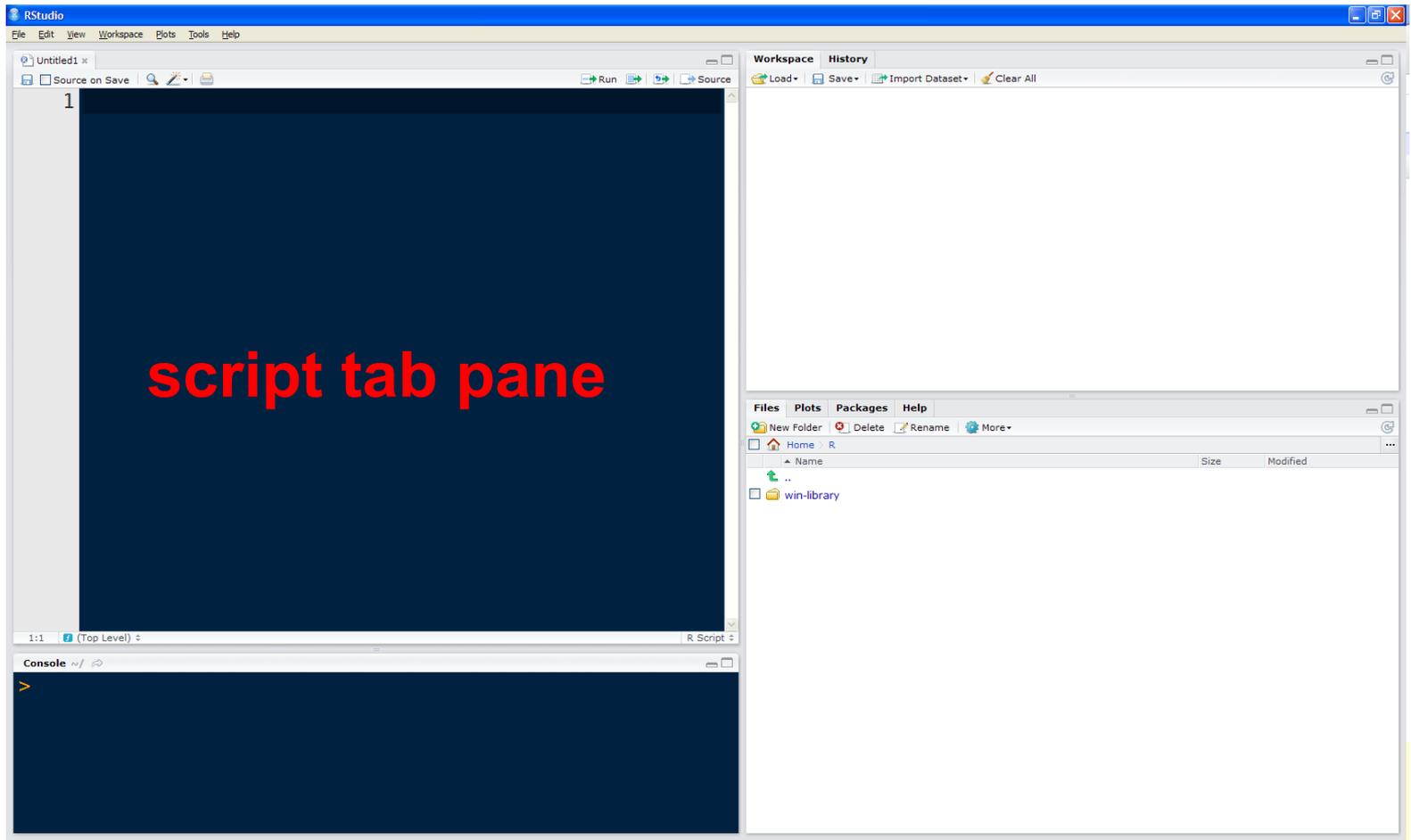
R studio



R studio



R studio



R studio

The screenshot displays the RStudio environment with the following components:

- Source Editor:** Contains R code for variable assignment and printing.
- Workspace:** Shows the current environment with variables v1, v2, and v3.
- Console:** Shows the execution output of the code.

```
1 v1 = 1 + 1
2 v2 = c(1+1, 1-1, 1*1, 1/1)
3 v3 = c(1&1, 1|1, 1&&1, 1||1, 1==1, 1!=1)
4
5 print(v1)
6 print(v2)
7 print(v3)
8
```

Variable	Value
v1	2
v2	numeric[4]
v3	logical[6]

```
> source('~/.active-rstudio-document')
[1] 2
[1] 2 0 1 1
[1] TRUE TRUE TRUE TRUE TRUE FALSE
>
```

**Don't type.
Use Ctrl/Cmd+Shift+S**

R studio

The screenshot displays the RStudio environment with the following components:

- Source Editor:** Contains R code for variable assignment and printing.

```
1 v1 = 1 + 1
2 v2 = c(1+1, 1-1, 1*1, 1/1)
3 v3 = c(1&1, 1|1, 1&&1, 1||1, 1==1, 1!=1)
4
5 print(v1)
6 print(v2)
7 print(v3)
8
```
- Workspace:** Shows the current environment with variables v1, v2, and v3. A red box highlights this area, and a context menu is open over it with options: "Save Workspace As..." and "Save As Default Workspace".
- Console:** Shows the output of the executed code.

```
> source('~/.active-rstudio-document')
[1] 2
[1] 2 0 1 1
[1] TRUE TRUE TRUE TRUE TRUE FALSE
>
```
- Files Panel:** Shows the file explorer with a folder named "win-library".



La natura di R

- Per la flessibilità e la modulabilità delle funzioni, è più corretto parlare di 'ambiente' che di pacchetto statistico. In termini tecnici, R è un linguaggio di tipo espressione; l'istruzione elementare può essere:
- espressione: viene eseguita l'istruzione, il risultato stampato su schermo, ma non salvato in memoria (es. $7+3$)
- assegnazione: viene eseguita l'istruzione, il risultato non viene stampato su schermo ma associato al nome della variabile (es. `nome<- 7+3`). L'operatore di assegnazione è "`<-`"



Piccola digressione (1)

L'assegnazione, in R, può essere fatta in due modi:

1. Usando “=”

2. Usando “<-”

□ “=” è semplice, veloce, ma ha valore solo nel contesto specifico (locale)

□ “<-” è più lento (devo usare 2 tasti) ma ha valore in tutto lo spazio



Piccola digressione (1)

```
RGui (32-bit) - [R Console]
File Modifica Visualizza Varie Pacchetti Finestre Aiuto
[Icons: Home, Refresh, Save, Print, Copy, Paste, Undo, Redo, Stop, Print]

>
> sum(a=2, b=3)
[1] 5
> a
Errore: oggetto "a" non esiste
> |
```

```
RGui (32-bit) - [R Console]
File Modifica Visualizza Varie Pacchetti Finestre Aiuto
[Icons: Home, Refresh, Save, Print, Copy, Paste, Undo, Redo, Stop, Print]

> sum(a<-2, b<-3)
[1] 5
> a
[1] 2
> b
[1] 3
> |
```



Piccola digressione (2)

- La sintassi delle istruzioni in R è “case-sensitive”:**
- R distingue tra **MAIUSCOLE** e **MINUSCOLE**, quindi il comando **DGET** non corrisponde a **dget**
 - R non fa caso agli spazi che lasciamo purché non alterino i nomi degli oggetti o delle funzioni

L'uso delle parentesi

In R, l'uso delle varie tipologie di parentesi è strategico:

- ▣ Le parentesi **tonde** `()` servono sempre e solo per fornire alle funzioni i loro argomenti;
- ▣ Le parentesi **quadre** `[]` servono sempre e solo a indicizzare gli oggetti;
- ▣ Le parentesi **tonde** `{ }` servono sempre e solo per i cicli `if`, `for`, etc.

Risorse e Riferimenti:

- Il materiale di questa lezione è stato assemblato utilizzando le seguenti risorse disponibili online:
 - http://psych.colorado.edu/wiki/lib/exe/fetch.php?media=courses:keller:5541:lecture1_2015.ppt