

introductoryR

Austin Haffenden

a.haffenden@keele.ac.uk
William Smith: WS1.16

www.github.com/ahaffenden

What is R?

An interpreted computer language.

- User-visible functions written in R
- Possible to interface with other programming languages
- System commands can be called from within R

Used for:

- data manipulation, statistics and graphics

Is made up of:

- Operators (+ - <- * %*% ...) for calculations on arrays & matrices
- Large, coherent, integrated collection of functions
- Facilities for making unlimited types of publication quality graphics
- User written functions & sets of functions (packages); 800+ contributed packages so far & growing

Open Source

- Not just free:
 - Provides full access to code
 - Ability to fix bugs and extend software
 - Forum allowing researchers to explore and expand methods
 - Cutting Edge! Constant development
 - Ensures that scientists are the co-owners of software tools needed to carry out research
 - Promotes reproducible research by providing open and accessible tools
 - Most of R is written in... R! Making it (relatively) easy to see what functions are actually doing.

R Advantages

- Fast and free.
- State of the art
- 2nd only to MATLAB for graphics.
- Active user community
- Excellent for simulation, programming, computer intensive analyses, etc.
- Forces you to *think* about your analysis.
 - You need to understand what you are doing
- Interfaces with database storage software (SQL)

R Disadvantages

- Steep learning curve
- No commercial support; can be frustrating
 - (see above)
- Easy to make mistakes (= learning)
- Working with very large datasets is limited by RAM (physical memory)
 - Newer packages developed to deal with this
- Little, but improving, support for 3D or dynamic graphics

R Packages

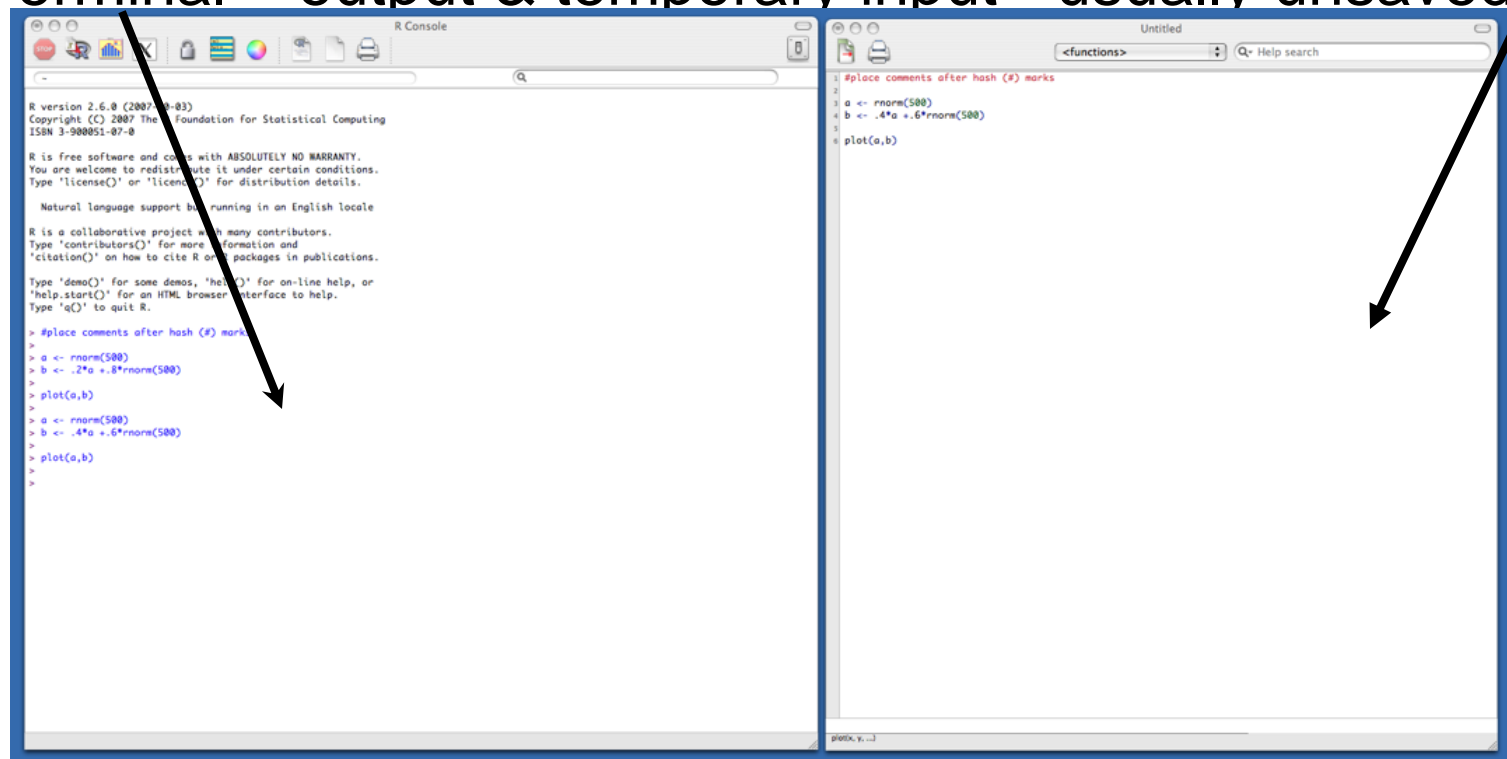
Repository:

<http://cran.r-project.org/src/contrib/PACKAGES.html>

- Enormous advantage - new techniques available without delay and you already know the language
- Allows construction of customized statistical programs
- Around 4000 packages on CRAN

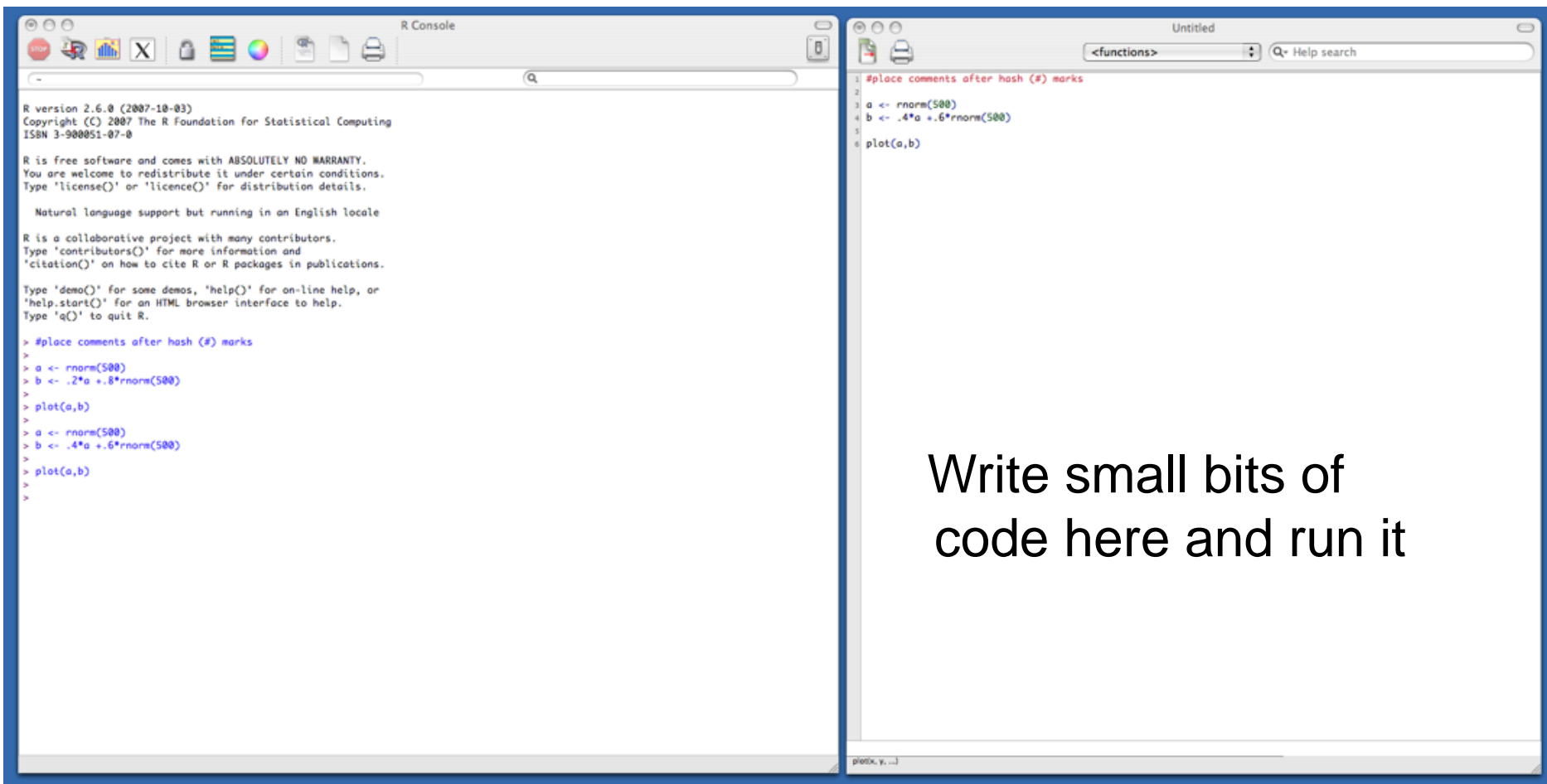
Typical R session

- Start up R via the GUI or favorite text editor
- Two windows:
 - 1+ new or existing scripts (text files) - these will be saved
 - Terminal – output & temporary input - usually unsaved



Typical R session

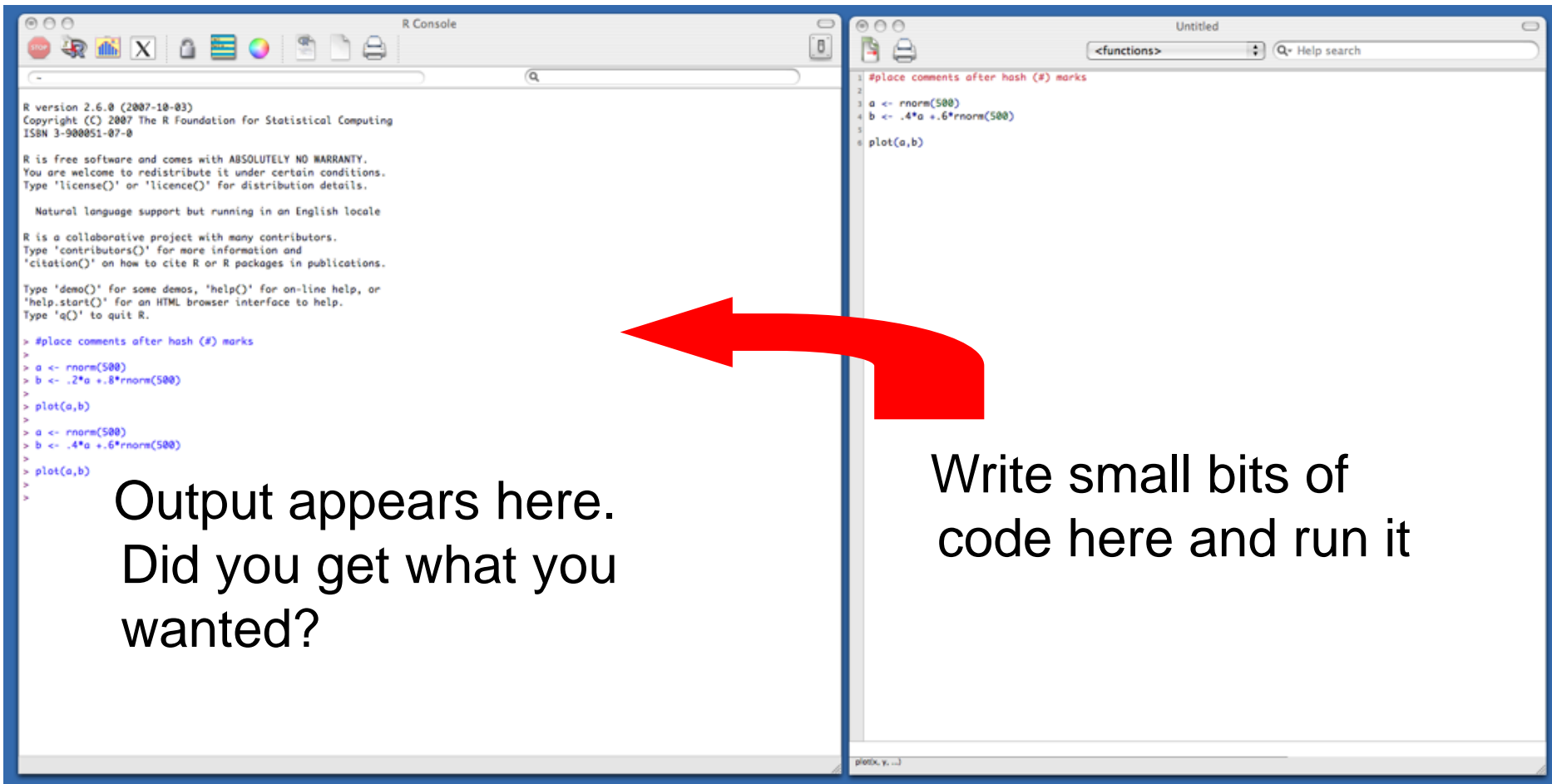
- R sessions are *interactive*



Write small bits of
code here and run it

Typical R session

- R sessions are *interactive*



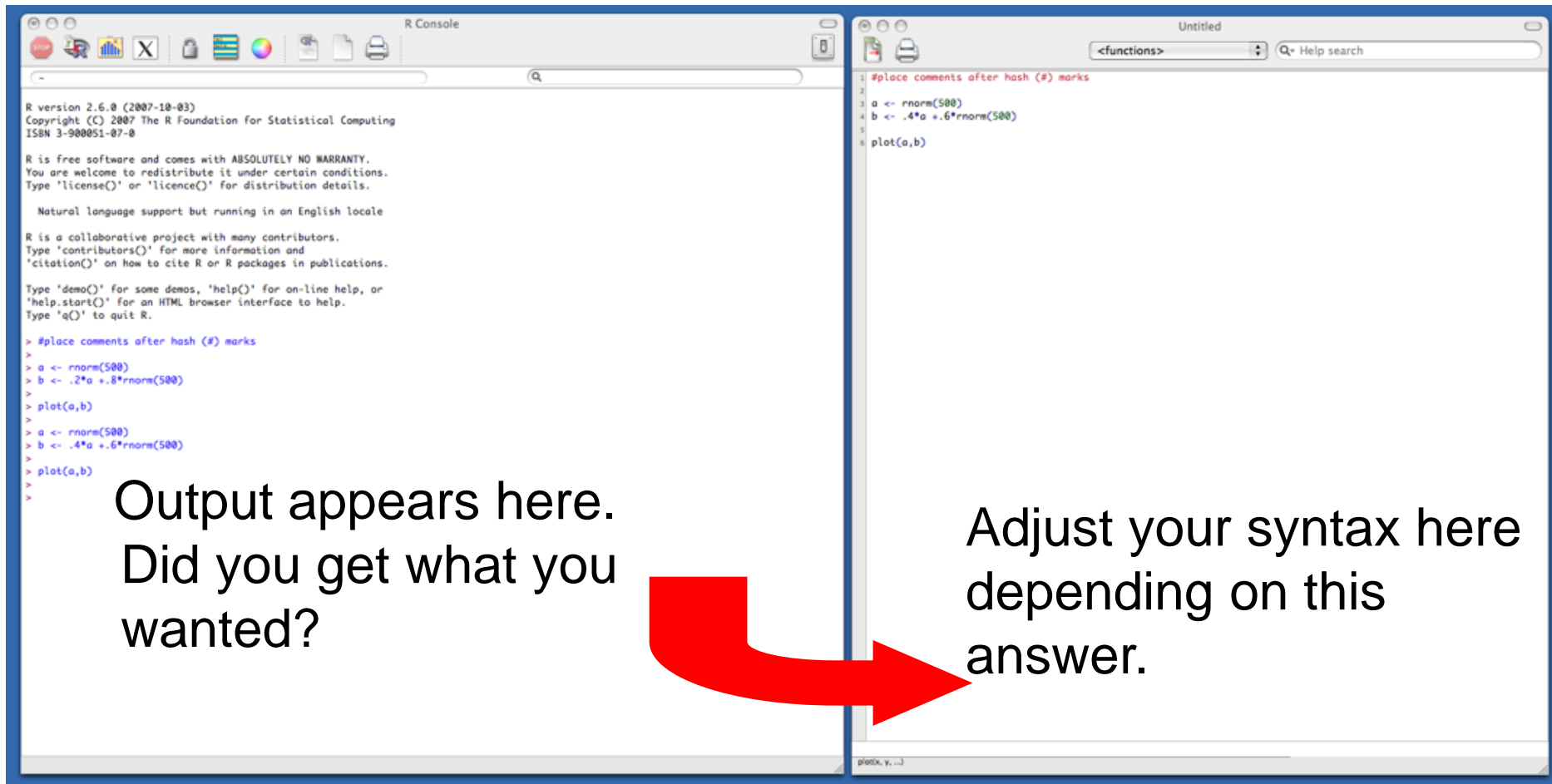
The image shows a typical R session interface. On the left is the R Console window, titled "R Console", which displays the R version (2.6.0) and copyright information. It also shows the R startup message, including the disclaimer "R is free software and comes with ABSOLUTELY NO WARRANTY." and instructions on how to use R. Below this, the console shows the execution of several R commands: `> #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)`, and `> plot(a,b)`. On the right is an Untitled script editor window, titled "Untitled", which contains the same R code as the console. A large red arrow points from the script editor to the console, indicating that the code written in the script editor is executed in the console. Below the console, the text "Output appears here. Did you get what you wanted?" is displayed. Below the script editor, the text "Write small bits of code here and run it" is displayed.

Output appears here.
Did you get what you wanted?

Write small bits of
code here and run it

Typical R session

- R sessions are *interactive*



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.

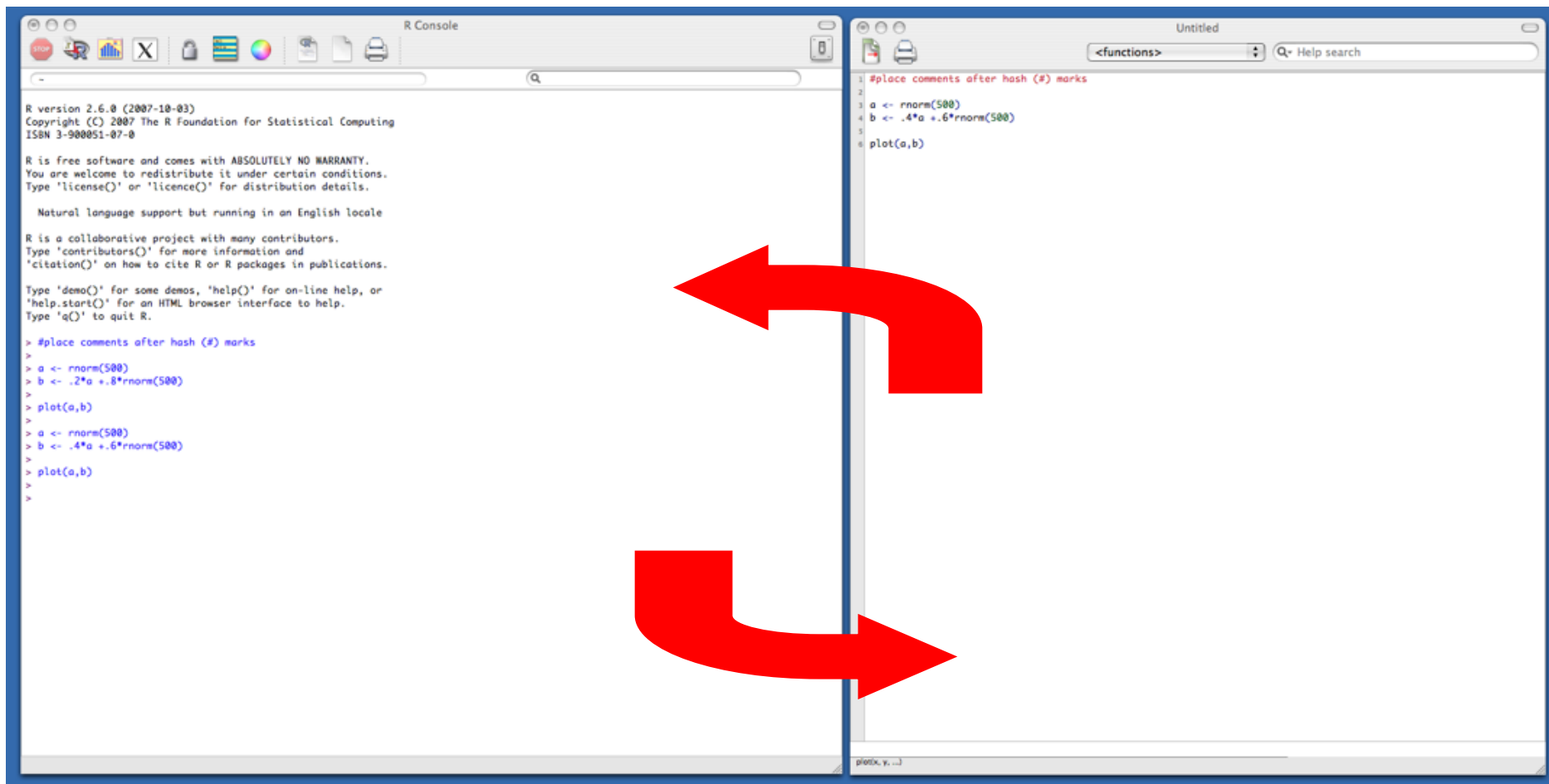
```
> #place comments after hash (#) marks
> 
> a <- rnorm(500)
> b <- .2*a +.8*rnorm(500)
> plot(a,b)
> 
> a <- rnorm(500)
> b <- .4*a +.6*rnorm(500)
> plot(a,b)
> 
```

Output appears here.
Did you get what you wanted?

Adjust your syntax here
depending on this
answer.

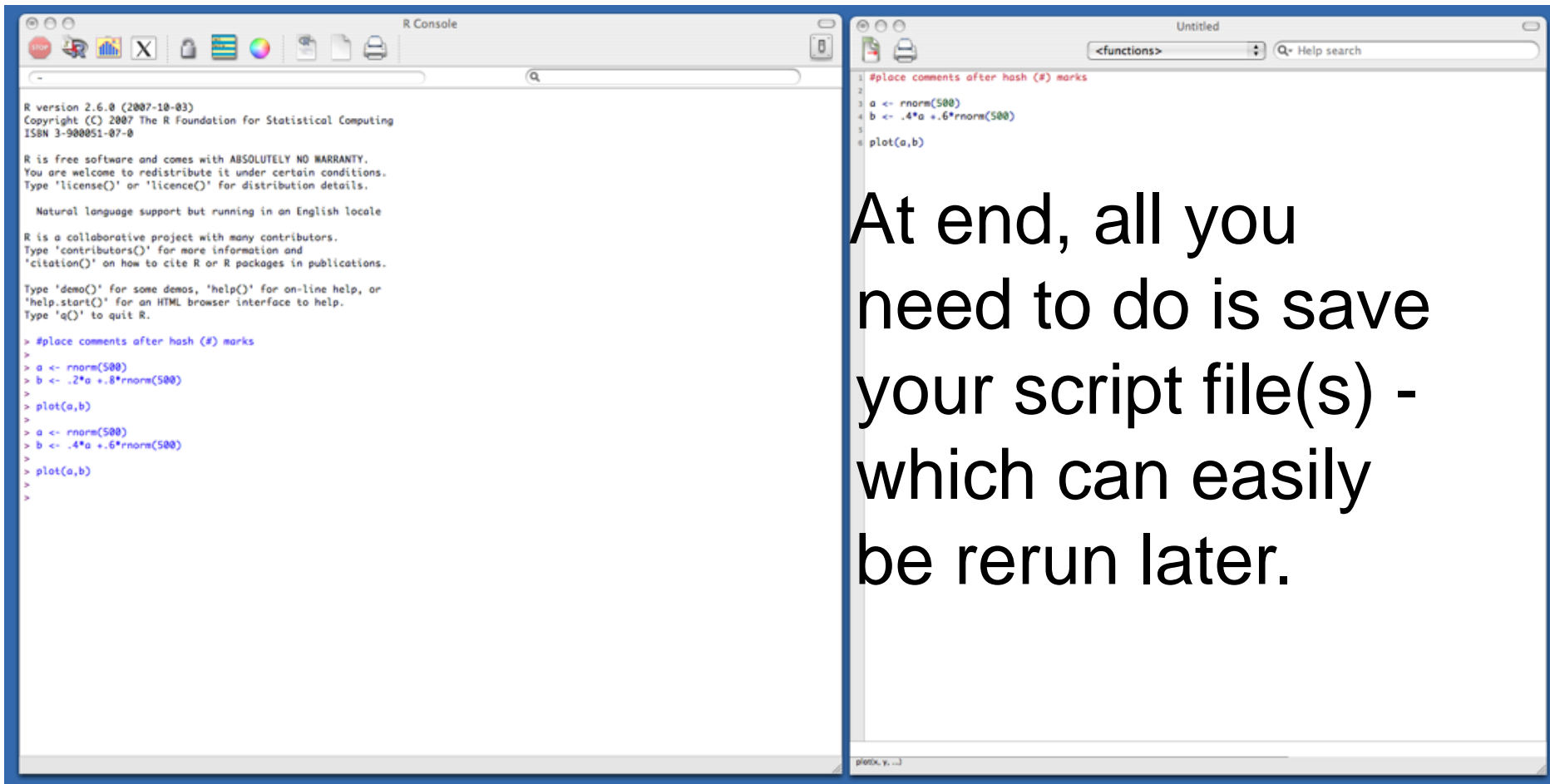
Typical R session

- R sessions are *interactive*



Typical R session

- R sessions are *interactive*



At end, all you need to do is save your script file(s) - which can easily be rerun later.

Learning R

- Swirl package
- Search for answers on Google
- Play with your data and make mistakes
- `help(lm)` or `?lm` gives you help on `lm` function.
 - Read the help files.
 - Try the example code provided
- Subscribe to r-help and ask questions (politely)
 - <https://stat.ethz.ch/mailman/listinfo/r-help>
- The more time that you spend using R the easier it gets.

R Document Resources

- Available from CRAN (<https://cran.r-project.org>)
 - An Introduction to R
 - Writing R Extensions – package development
 - R Data Import/Export
- A multitude of tutorials, cheat sheets, scripts etc. available on the web.

For next session

- Continue with the swirl package
- **‘Open Intro:** A very basic introduction to statistics, data analysis, and data visualization’

```
library(swirl)
install_from_swirl("Open Intro")
swirl()
```