



Basic statistics with R



Goal

- Introduce R
- Give examples on use of the tool
- Guide to further knowledge



What is R?

- R is a software environment that includes a set of base packages for graphics, math, and statistics.
- You can make use of specialized packages contributed by R users or write your own new functions.



Why R?

- It's free. R can be downloaded at no cost from: cran.r-project.org.
- It runs on a variety of platforms including Windows, Unix and MacOS.
- It contains advanced statistical routines not yet available in other packages.



R paradigm

- R process is highly interactive (not setting up a complete analysis at once).
- Use runs a command, take the results and process it through another command, take those results and process it through another command.
- The cycle may include transforming the data, and looping back through the whole process again.
- Users can stop when they feel that they have fully analysed the data.



R overview

- R is a comprehensive statistical and graphical programming language.
- R: initially written by [Ross Ihaka](#) and [Robert Gentleman](#) at Dep. of Statistics of U of Auckland, New Zealand during 1990s.
- Since 1997: international “R-core” team of 15 people with access to common CVS archive.



R overview

- You can enter commands one at a time at the command prompt (>) or run a set of commands from a source file.
- There is a wide variety of data types, including vectors (numerical, character, logical), matrices, dataframes, and lists.
- To quit R, use >q()



R overview

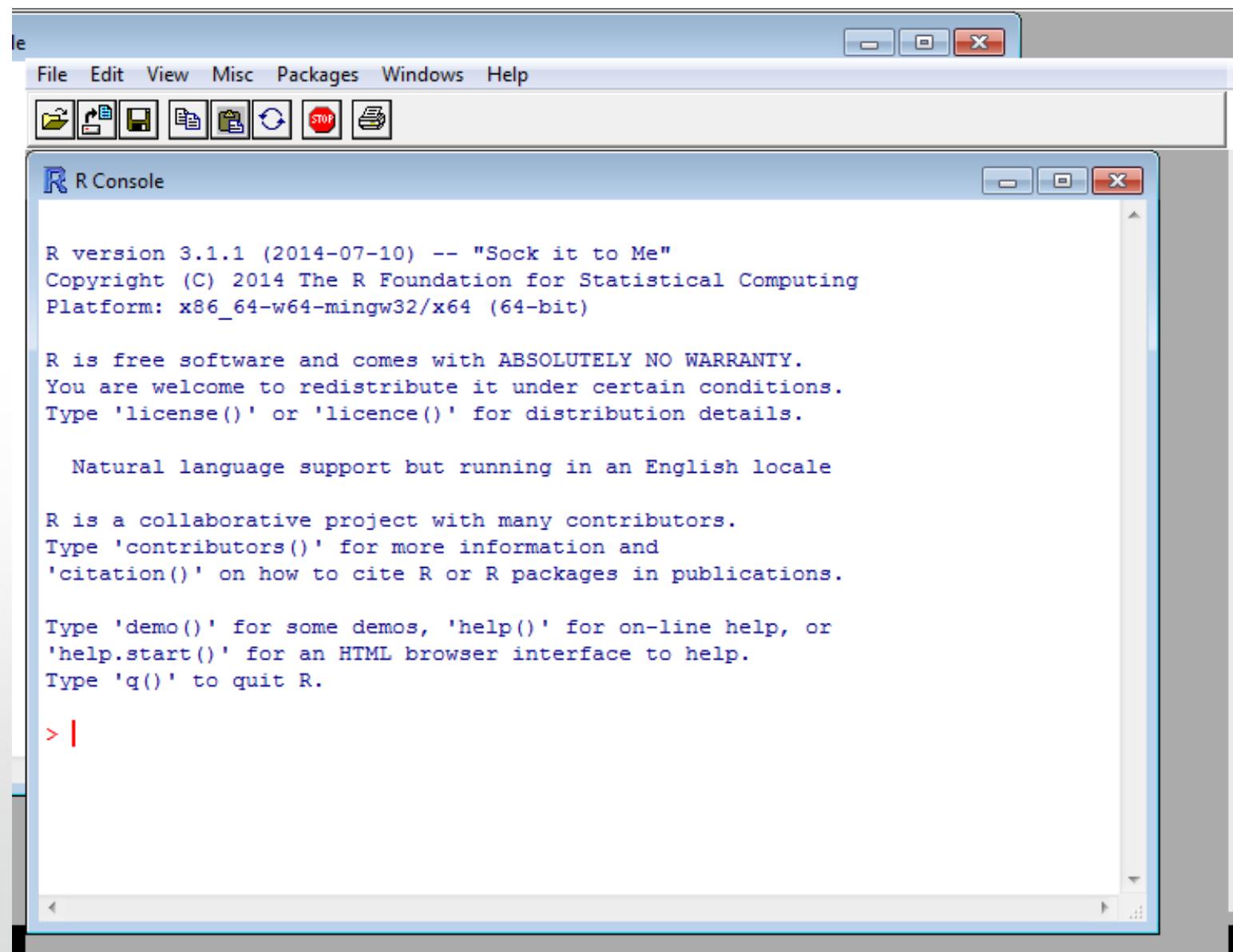
- A key skill to using R effectively is learning how to use the built-in help system.
- A fundamental design feature of R is that the output from most functions can be used as input to other functions.



R Interface

- Start the R system, the main window (RGui) with a sub window (R Console) will appear
- In the 'Console' window the cursor is waiting for you to type in some R commands.

R interface





Install R

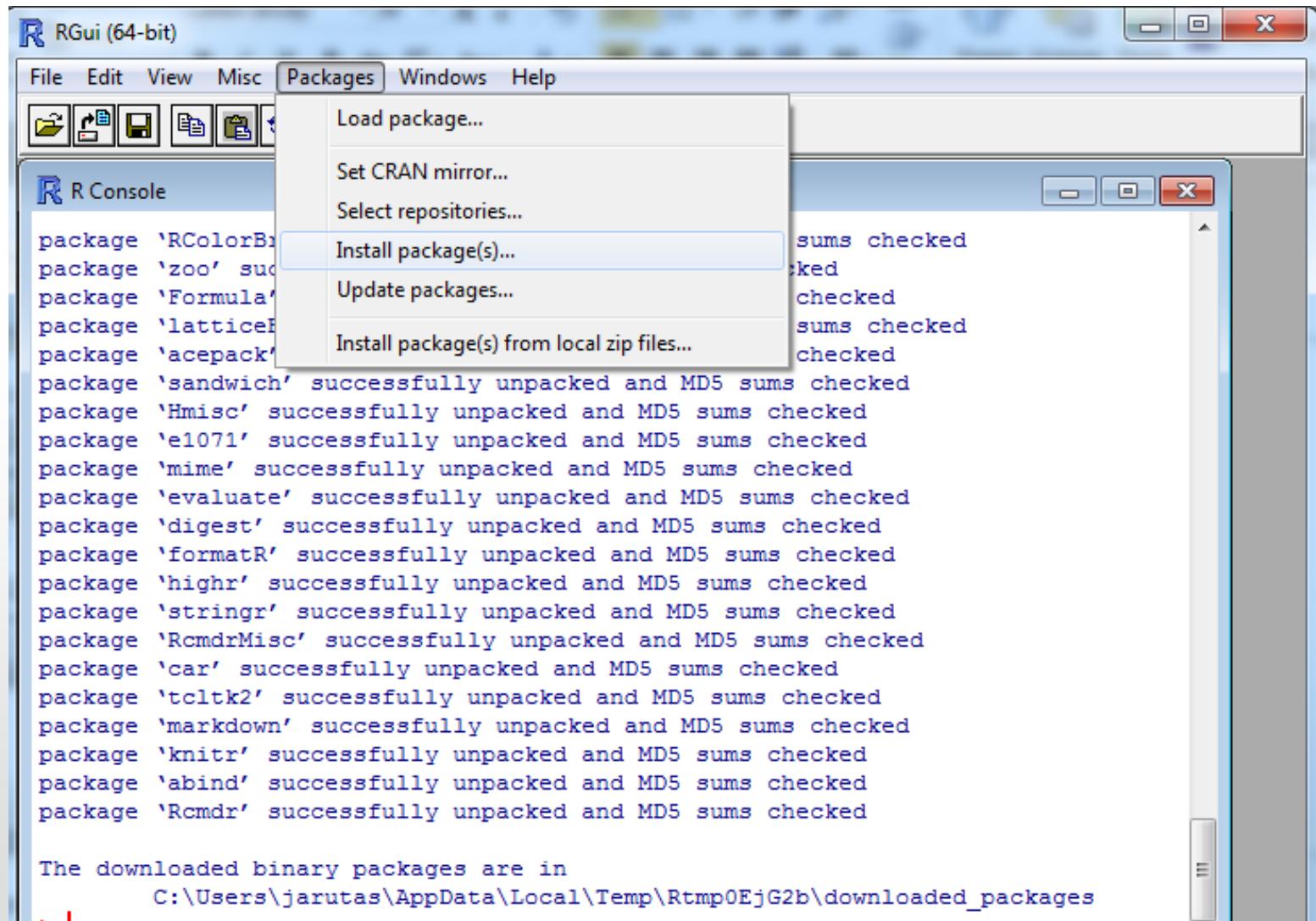
- For Windows:
 - <http://cran.r-project.org/bin/windows/base/>
- For Linux (Ubuntu etc):
 - Just search for R-base-core in Synaptic Package Manager and add it
 - <http://socserv.mcmaster.ca/jfox/Misc/Rcmdr/installation-notes.html>



Install R commander

- Go to the “Packages” tab and click on “Install Packages”.
- The first time you’ll do this you’ll be prompted to choose a CRAN mirror.
- R will download all necessary files from the server you select here.
- Choose the location closest to you

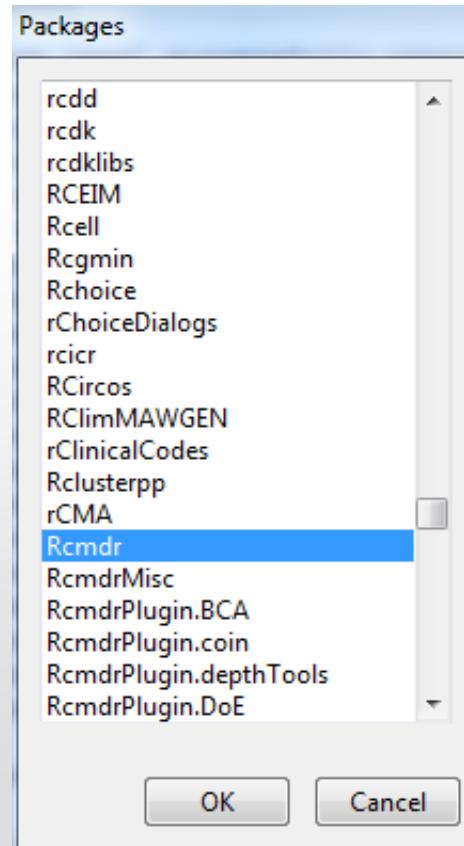
Install R commander





Install R commander

- Choose “Rcmdr” from a list
- Ensure that “Install dependencies” is checked, and click “Install”.





Some things to note first

- R is case-sensitive
 - help, Help, HELP and HELF are different...
 - Recommendation: Choose one style and stick to it
- Avoid using shortcut keys in R commander!
- If it's something you don't know?
 - There are lot's of good information on the web:
Particularly for R



Getting Started

The screenshot shows the RGui (64-bit) interface. The title bar reads "RGui (64-bit)". The menu bar includes File, Edit, Packages, Windows, and Help. Below the menu is a toolbar with icons for file operations. The main window is divided into two panes: the left pane is the "R Console" and the right pane is the "Untitled - R Editor".

R Console Output:

```
R version 3.3.1 (2016-06-21) -- "Bug in Your Hair"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-w64-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,
'help.start()' for an HTML browser interface to help,
Type 'q()' to quit R.
```

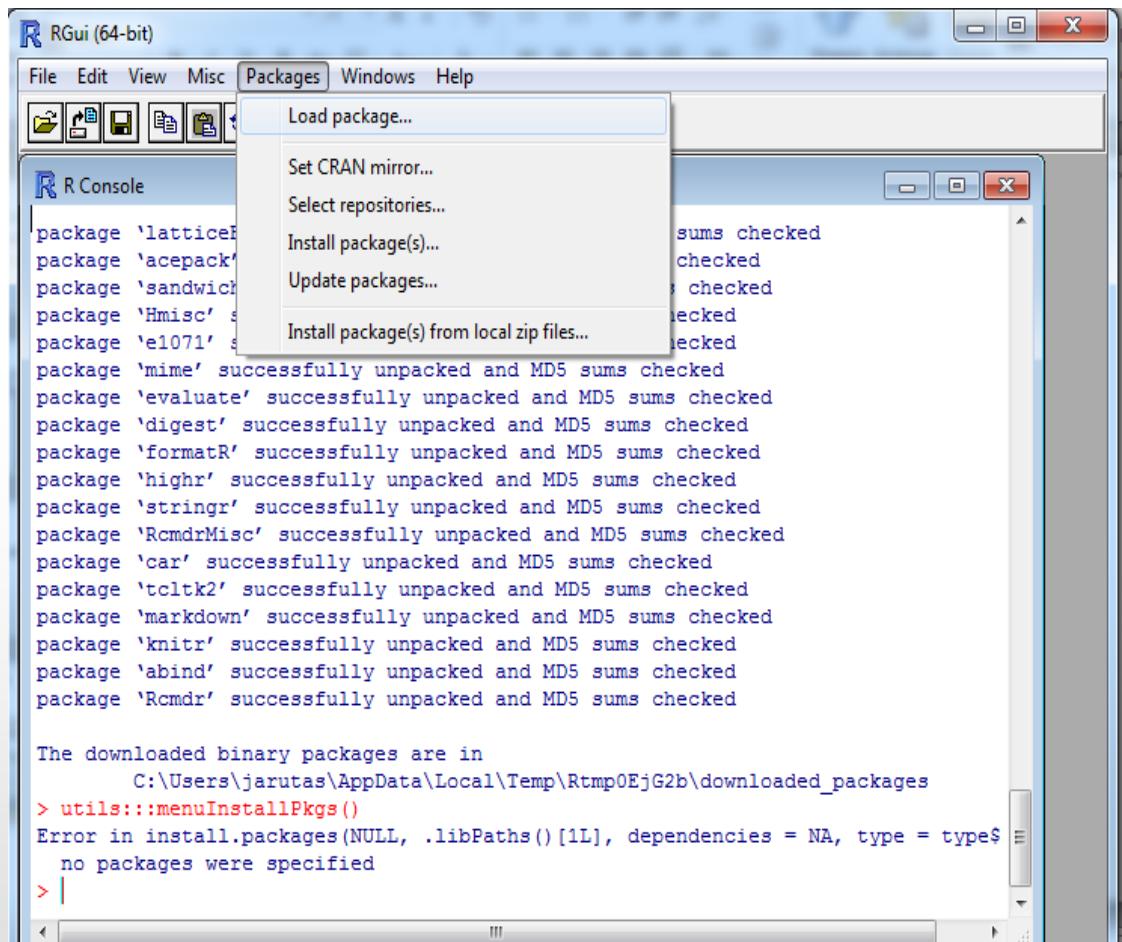
A red cursor is visible in the R Console input field at the bottom left.



Data representation

Start using R

- Open R
- Load packages
 - Rcmdr





R commander

The screenshot shows the R commander application window. At the top is a menu bar with File, Edit, Data, Statistics, Graphs, Models, Distributions, Tools, and Help. Below the menu is a toolbar with icons for R logo, Data set (set to <No active dataset>), Edit data set, View data set, and Model (set to <No active model>). The main area contains two tabs: R Script (selected) and R Markdown. Below these tabs is a large empty text area for writing R code. To the right of this area is a vertical scroll bar. At the bottom of this section is a horizontal scrollbar. Below the R Script tab is an "Output" section with a "Submit" button featuring a circular arrow icon. This section also has a vertical scroll bar on its right. At the bottom of the window is a "Messages" section containing the text:

```
RGui  
with the single-document interface (SDI); see ?Commander.
```



Menu

- File Menu:
 - items for loading and saving script files;
 - for saving output and the R workspace;
 - and for exiting
- Edit Menu:
 - items (Cut, Copy, Paste, etc.) for editing the contents of the script and output windows.
 - Right clicking in the script or output window also brings up an edit “context” menu.



Menu

- Data
 - Submenus containing menu items for reading and manipulating data.
- Statistics
 - Submenus containing menu items for a variety of basic statistical analyses.



Menu

- **Graphs**
 - Menu items for creating simple statistical graphs.
- **Models**
 - Menu items and submenus for obtaining numerical summaries, confidence intervals, hypothesis tests, diagnostics, and graphs for a statistical model, and for adding diagnostic quantities, such as residuals, to the data set.
- **Distributions**
 - Probabilities, quantiles, and graphs of standard statistical distributions (to be used, for example, as a substitute for statistical tables) and samples from these distributions.



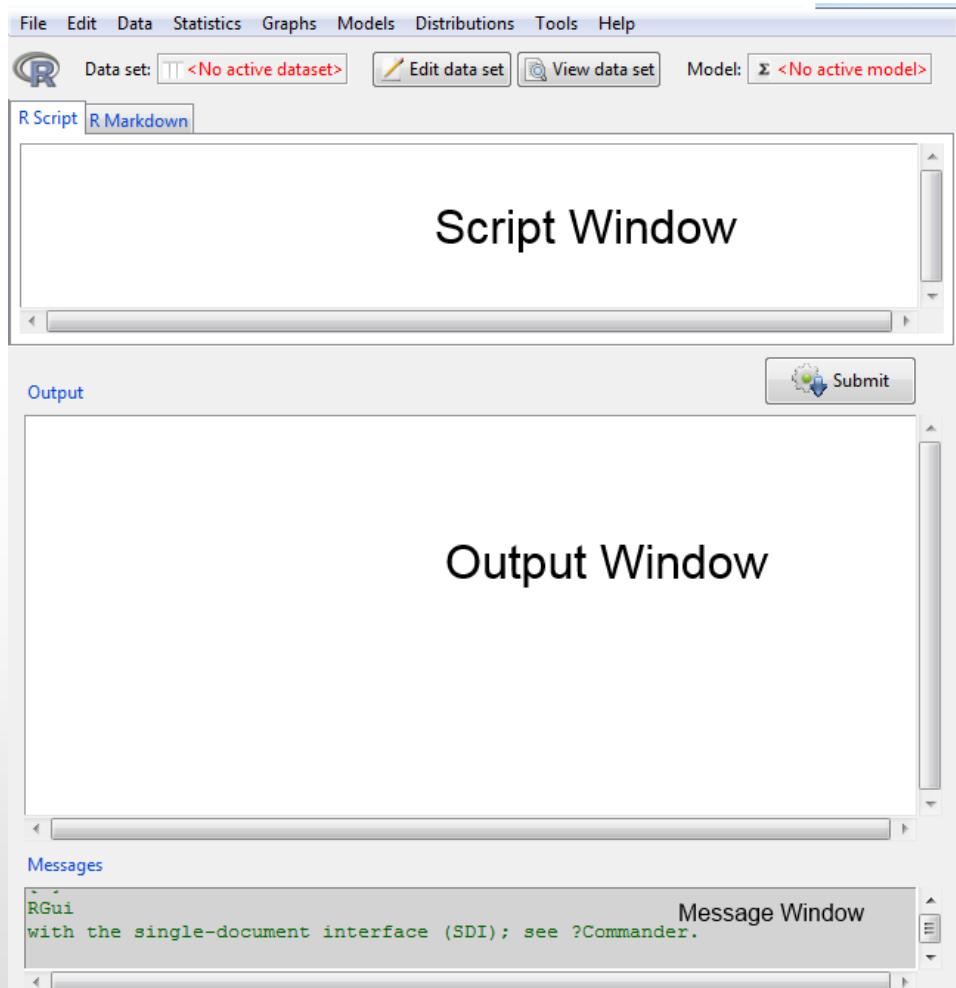
Menu

- Tools
 - Menu items for loading R packages unrelated to the Rcmdr package (e.g., to access data saved in another package), and for setting some options.
- Help Menu
 - items to obtain information about the R Commander (including this manual). As well, each R Commander dialog box has a Help button (see below).



R commander interface

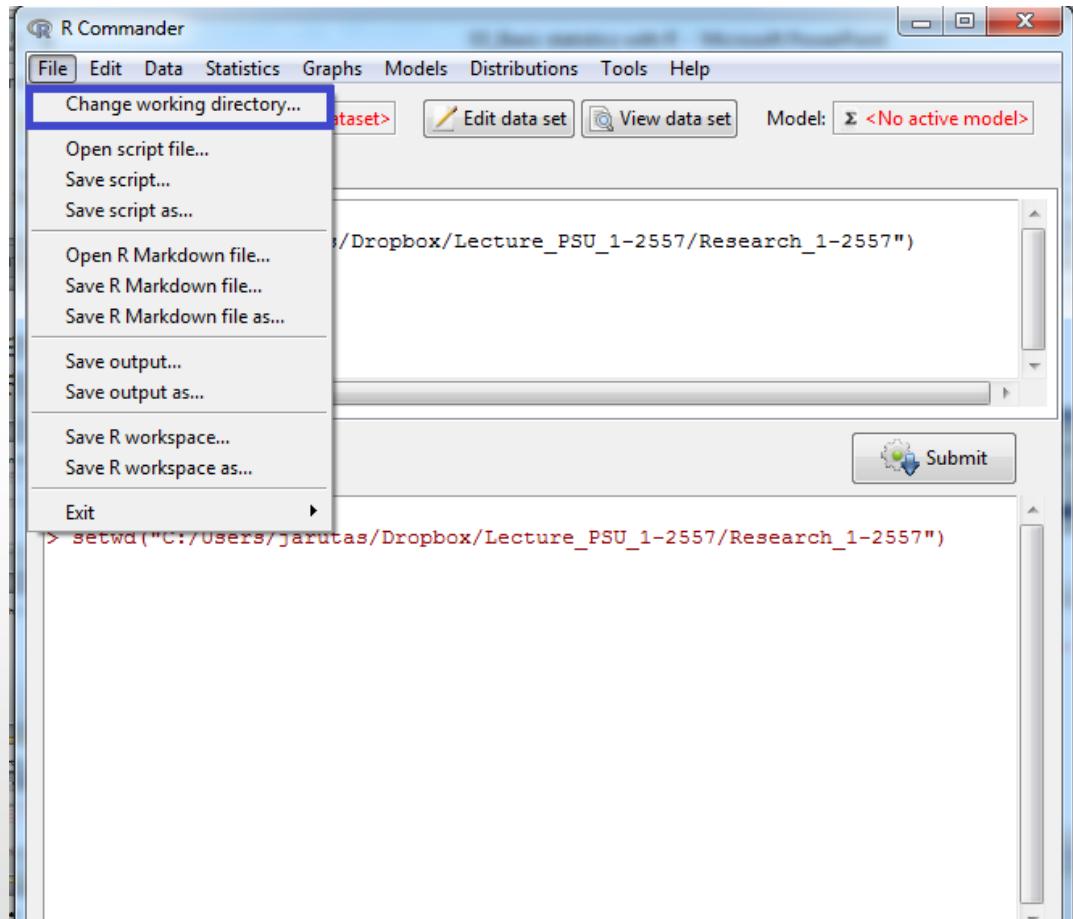
- **Script Window**
 - R commands generated by the R Commander
 - You can also type R commands directly into the script window or the R Console
 - The main purpose of the R Commander, however, is to avoid having to type commands.
- **Output Window**
 - Printed output
- **Messages Window**
 - Displays error messages, warnings, and notes
- **Graphics Device window**
 - When you create graphs, these will appear in a separate window outside of the main R Commander window.





Starting with R

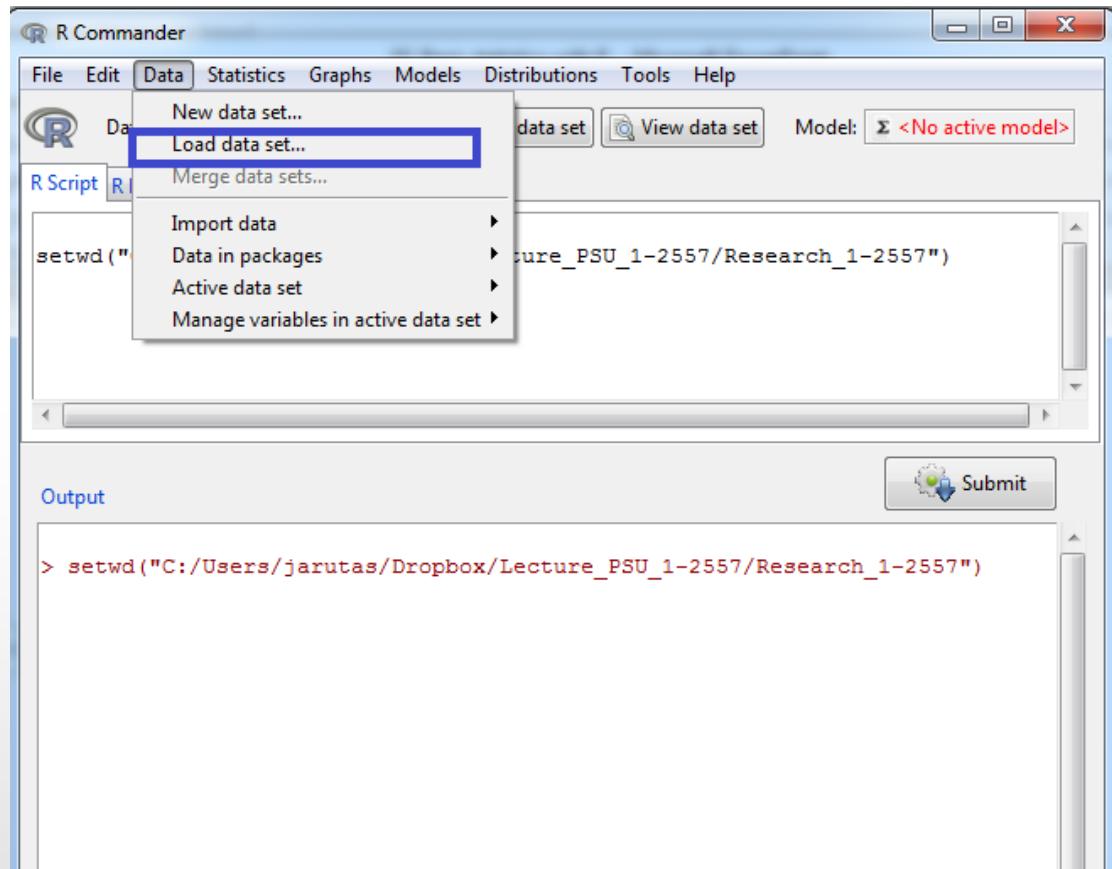
- Change directory
 - Find the folder where you placed your data file





Starting with R

- Load data
 - Select dataset





Starting with R

- Save workspace as
 - Give a name to your file



Let's get started...

- Change directory
 - Find the folder where you placed your data file
- Import data
 - Give it the name:
Arachnophobia
- Save workspace as
 - Give a name to your file

The screenshot shows the R Commander interface. The 'Data' menu is open, displaying options such as 'New data set...', 'Load data set...', 'Merge data sets...', 'Import data' (with sub-options for text files, SPSS, SAS, Minitab, and STATA), 'Data in packages', 'Active data set', and 'Manage variables in active data set'. Below the menu, the 'Output' window displays R code and its results:

```
data: anxiety_picture and anxiety_real
t = -2.4725, df = 11, p-value = 0.03098
alternative hypothesis: true difference in means is not eq
95 percent confidence interval:
-13.2312185 -0.7687815
sample estimates:
mean of the differences
-7

> library(shiny) nrow=15)
```



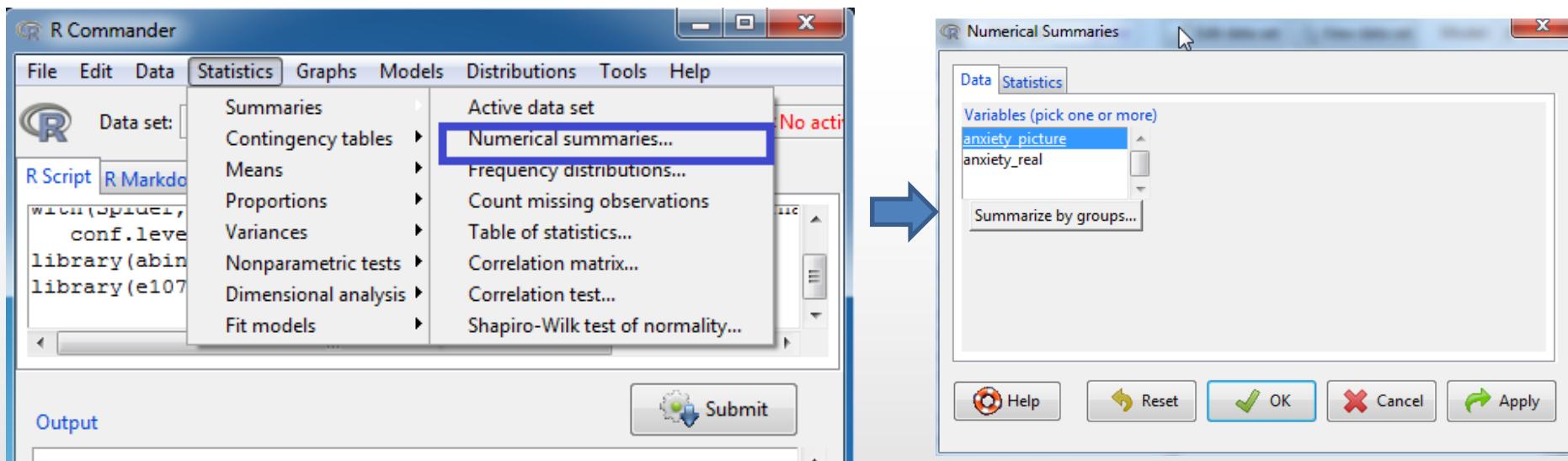
Data Types

- **Vectors**
 - including continuous variables
- **Factors**
 - Nominal/ categorical
- Matrices, arrays and data frames
- Lists

<http://www.statmethods.net/input/datatypes.html>

Summarize variable

- Calculate mean, median and standard deviation
 - Numerical summaries





File Edit Data Statistics Graphs Models Distributions Tools Help

Data set: Spider Edit data set View data set Model: <No active model>

R Script R Markdown

```
library(e1071, pos=15)
numSummary(Spider[, "anxiety_picture"],
  statistics=c("mean", "sd", "IQR", "quantiles"),
  quantiles=c(0, .25, .5, .75, 1))
```

Output Submit

```
Sample estimates:
mean of the differences
-7
```

```
> library(abind, pos=15)

> library(e1071, pos=16)

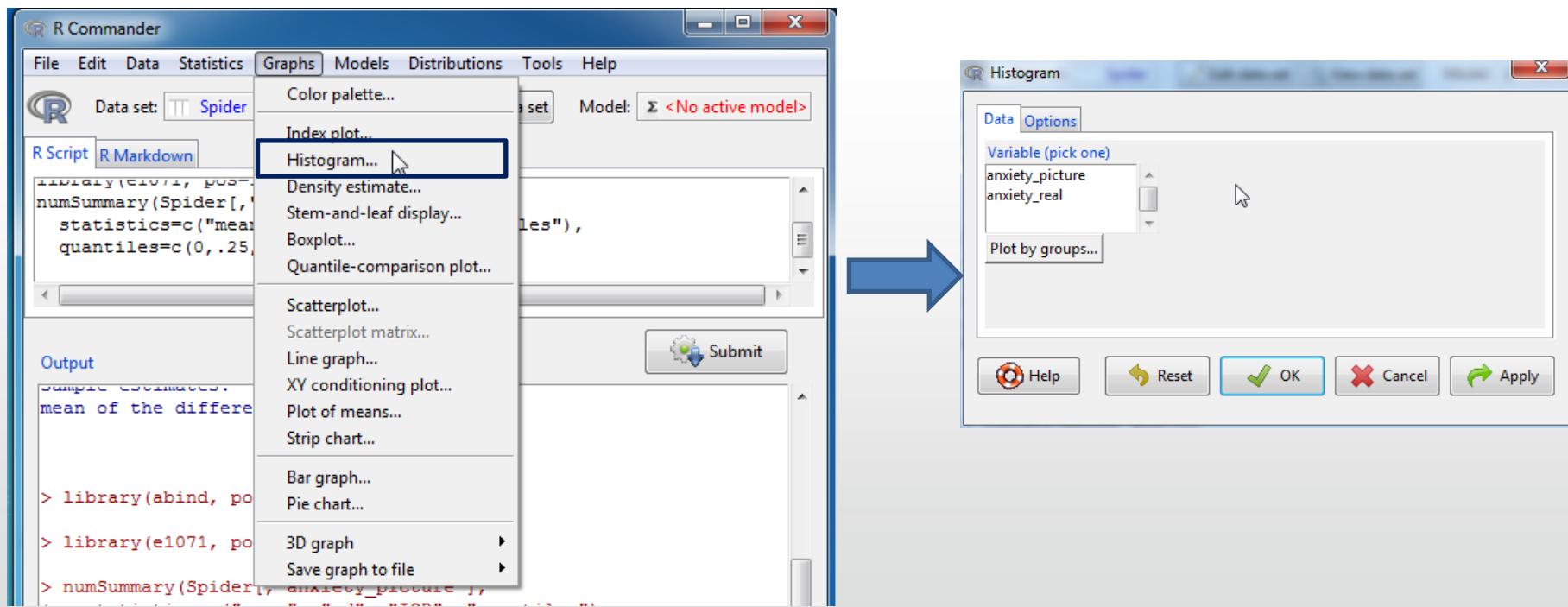
> numSummary(Spider[, "anxiety_picture"],
+   statistics=c("mean", "sd", "IQR", "quantiles"),
+   quantiles=c(0, .25, .5, .75, 1))
  mean      sd    IQR 0% 25% 50% 75% 100% n
  40 9.293204 12.5 25 33.75 40 46.25 55 12
```

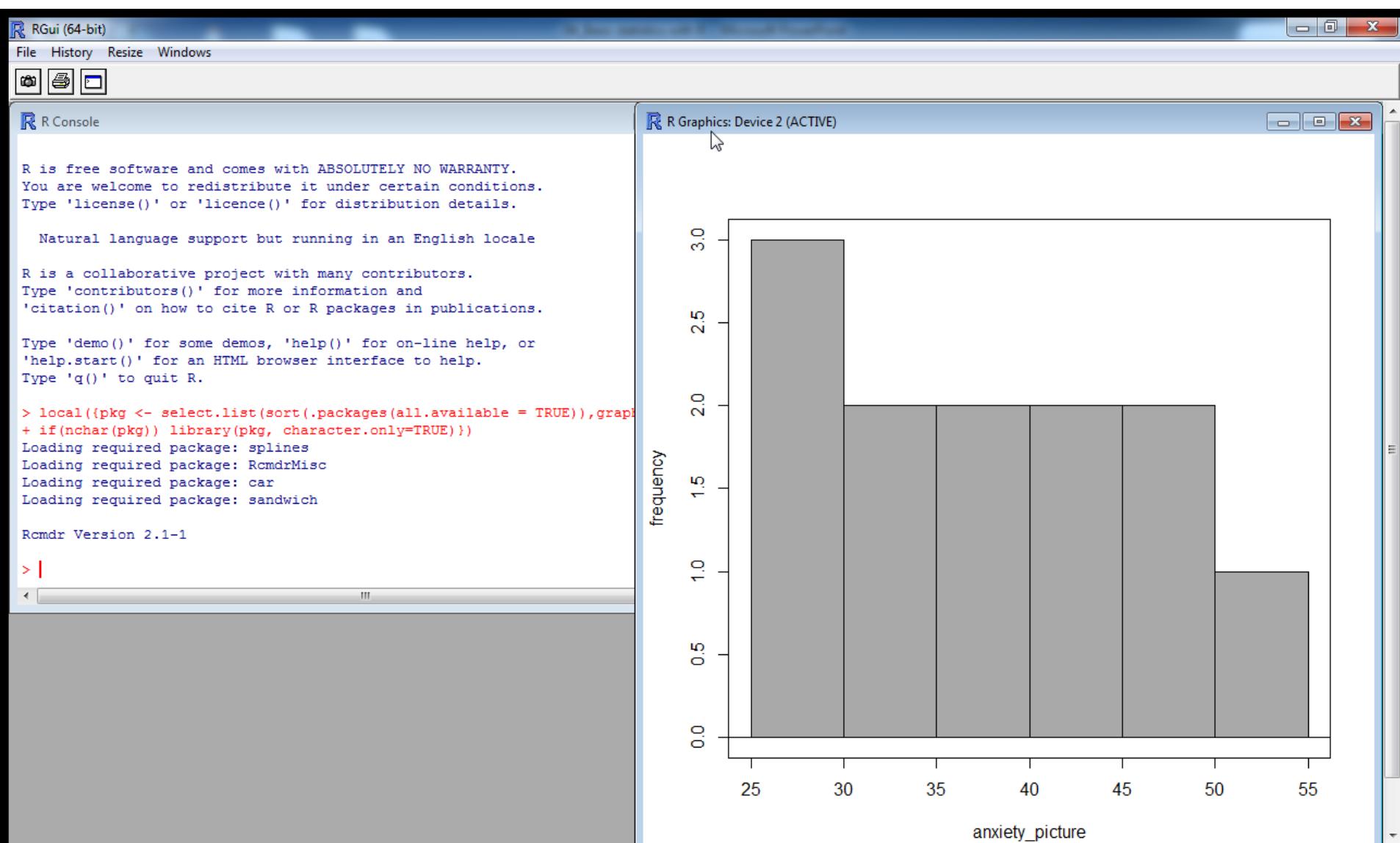
Messages

```
with the single-document interface (SDI); see ?Commander.
[3] NOTE: The dataset Spider has 12 rows and 3 columns.
```

Make a histogram of the response

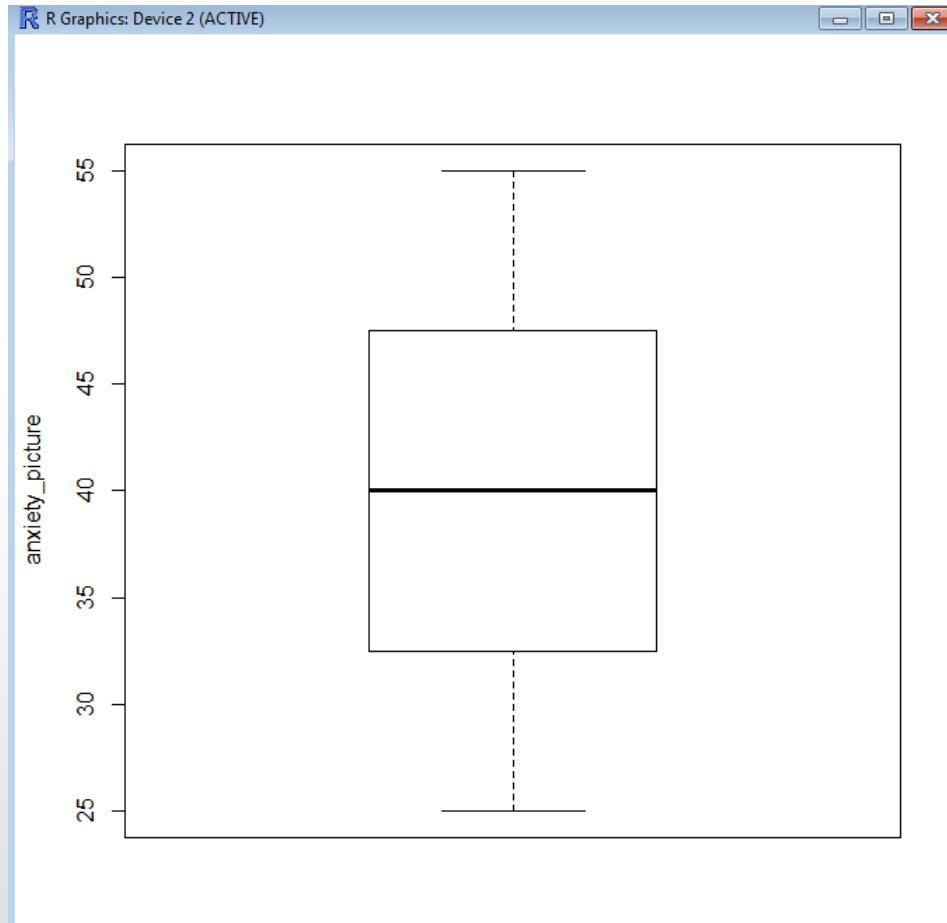
- The histograms will be printed in the R window (not inside R commander)
 - Right click on the graph and you can copy it as metafile to paste it into a document, print it or save it



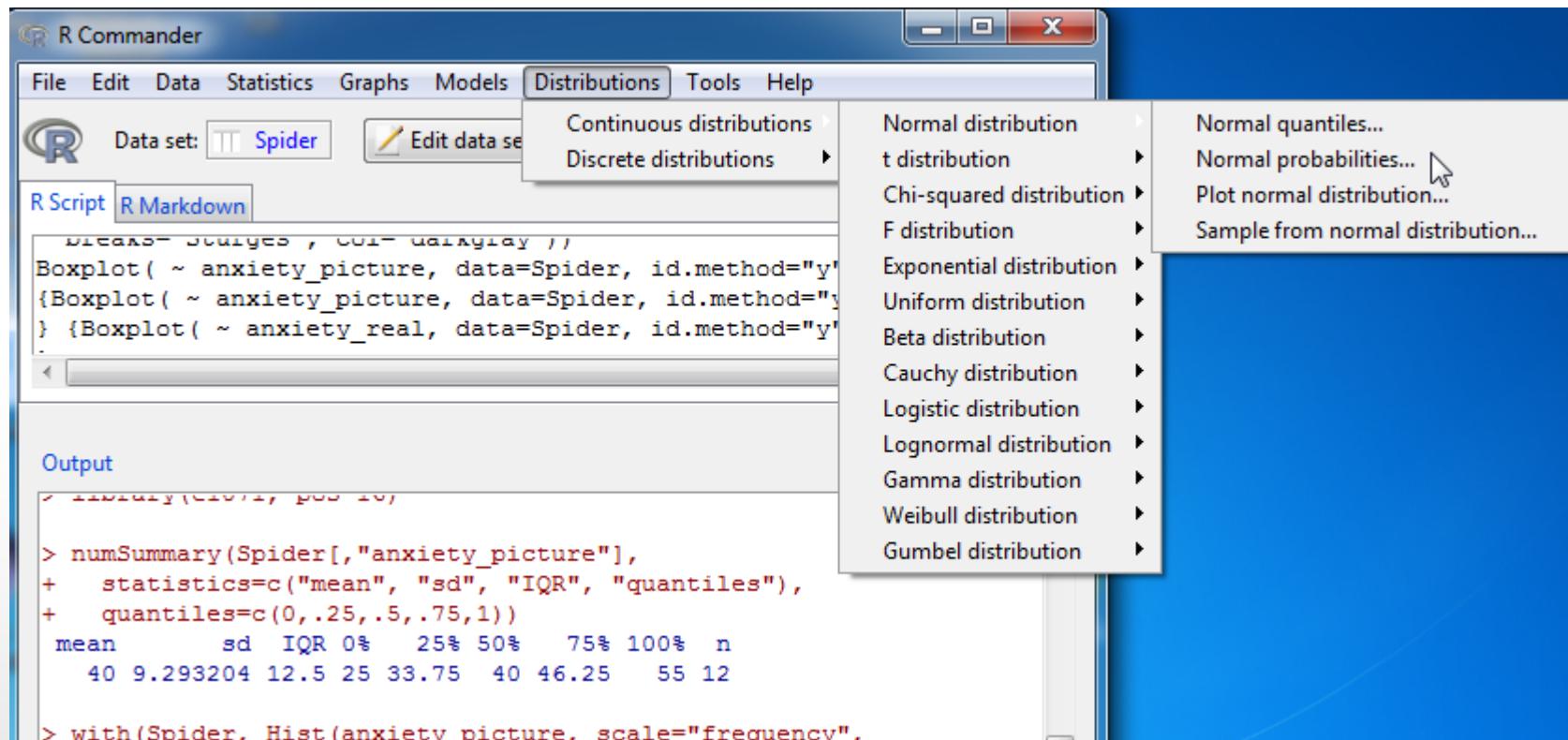


Box plot

- Box plot

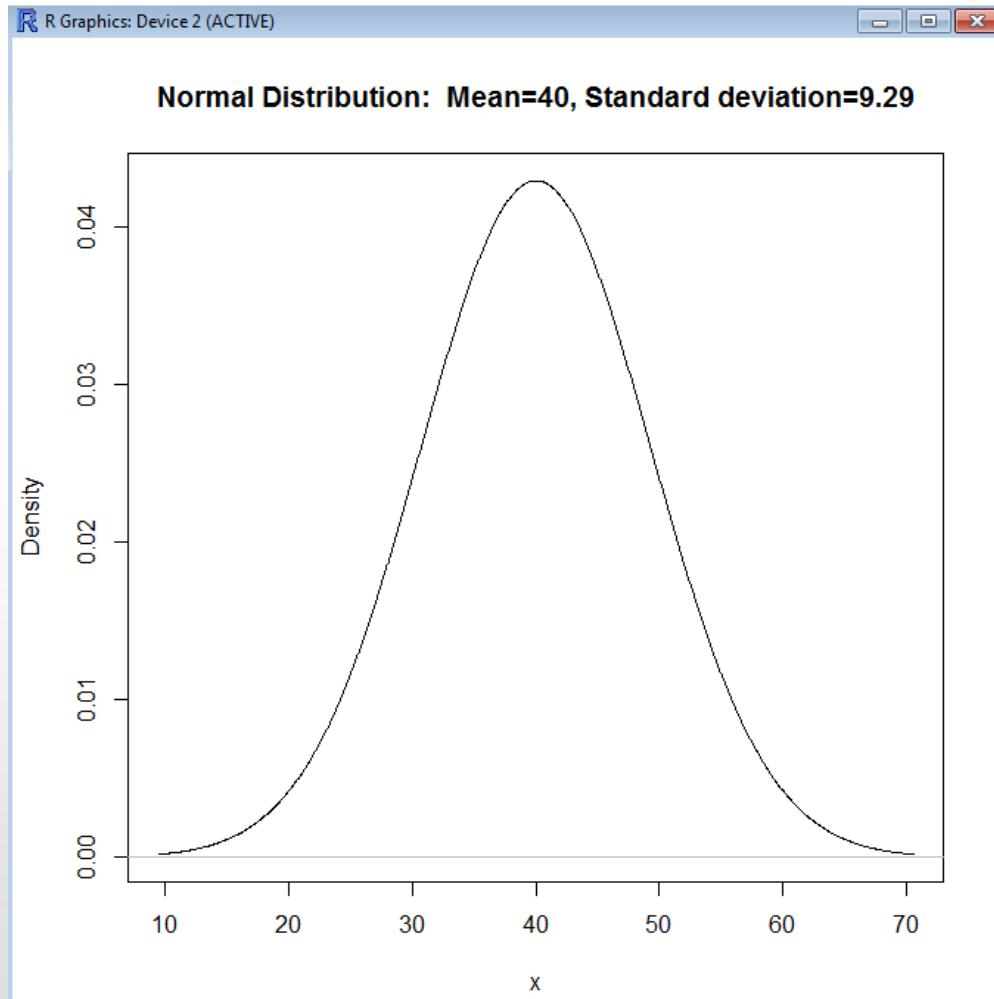


Check the assumption





Check the assumption (cont)





t-test (Paired sample)

R Commander

File Edit Data Statistics Graphs Models Distributions Tools Help

Data set: View data set Model: Σ <No active model>

R Script R Markdown

```
xlab="x",
main=paste
})
```

Summaries
Contingency tables
Means
Proportions
Variances
Nonparametric tests
Dimensional analysis
Fit models

Single-sample t-test...
Independent samples t-test...
Paired t-test...
One-way ANOVA...
Multi-way ANOVA...

Output

```
> with(Spider, Hist(anxiety_picture, scale="frequency",
+   breaks="Sturges", col="darkgray"))

> Boxplot( ~ anxiety_picture, data=Spider, id.method="y")

> Boxplot( ~ anxiety_picture, data=Spider, id.method="y")
...
...
```

Submit



Paired t-Test

Data Options

First variable (pick one) Second variable (pick one)

anxiety_picture anxiety_picture
anxiety_real anxiety_real

Help Reset OK Cancel Apply



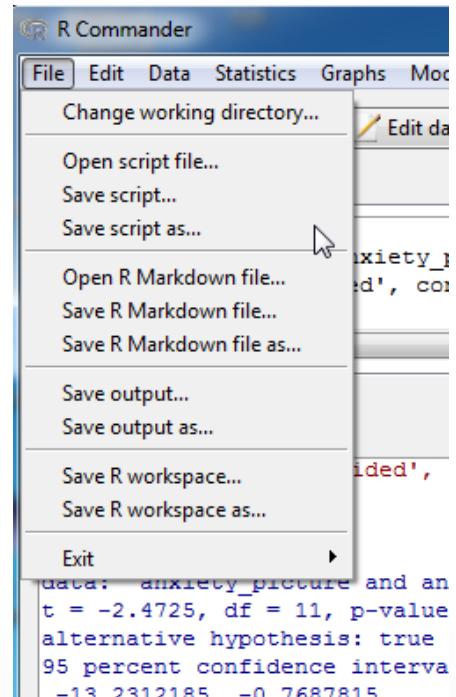
t-test (Paired sample): result

Output Submit

```
+   alternative='two.sided', conf.level=.95, paired=TRUE)))  
  
Paired t-test  
  
data: anxiety_picture and anxiety_real  
t = -2.4725, df = 11, p-value = 0.03098  
alternative hypothesis: true difference in means is not equal to 0  
95 percent confidence interval:  
 -13.2312185 -0.7687815  
sample estimates:  
mean of the differences  
 -7
```

How to save?

- Save R workspace as...
 - This will save your data (in the R format)
- Save output as...
 - This will save your output
 - Another strategy is to cut and paste what you want to save
- Always save the commands
 - essential if you want to re-run the analyses later
 - WordPad is a better option than Word etc (does not autocorrect - change to upper case etc)



How to write and a command?

- Simply write the command in the script window, mark it and click 'Submit' or press Ctrl+R



Nice to know:

- When writing **comments** in the syntax, start with the following sign '#'
- If you are uncertain about a function, use google or help(name-of-function)

Are there differences in syntax between R and R commander?

- Commands that extend over more than one line should have the second and subsequent lines indented by one or more spaces or tabs; all lines of a multiline command must be submitted simultaneously for execution.
- Commands that include an assignment arrow (<-) will not generate printed output, even if such output would normally appear had the command been entered in the R Console [the command `print(x <- 10)`, for example]. On the other hand, assignments made with the equals sign (=) produce printed output even when they normally would not (e.g., `x = 10`).

Are there differences in syntax between R and R commander?

- Commands that produce normally invisible output will occasionally cause output to be printed in the output window. This behaviour can be modified by editing the entries of the log-exceptions.txt file in the R Commander's etc directory.
- Blocks of commands enclosed by braces, i.e., {}, are not handled properly unless each command is terminated with a semicolon (;). This is poor R style, and implies that the script window is of limited use as a programming editor. For serious R programming, it would be preferable to use the script editor provided by the Windows version of R itself, or – even better – a programming editor.



Further reading:

- The R Commander A Basic-Statistics Graphical User Interface to R - John Fox 2005.pdf
 - <http://www.jstatsoft.org/v14/i09/paper>
- Getting started with the R Commander: a basic-statistics graphical user interface to R
 - <http://socserv.mcmaster.ca/jfox/Getting-Started-with-the-Rcmdr.pdf>
- Quick-R: magnificent guide
 - <http://www.statmethods.net/>
- <http://cran.r-project.org/manuals.html>



R help forum:

- <http://r.789695.n4.nabble.com/>

