

Introduction

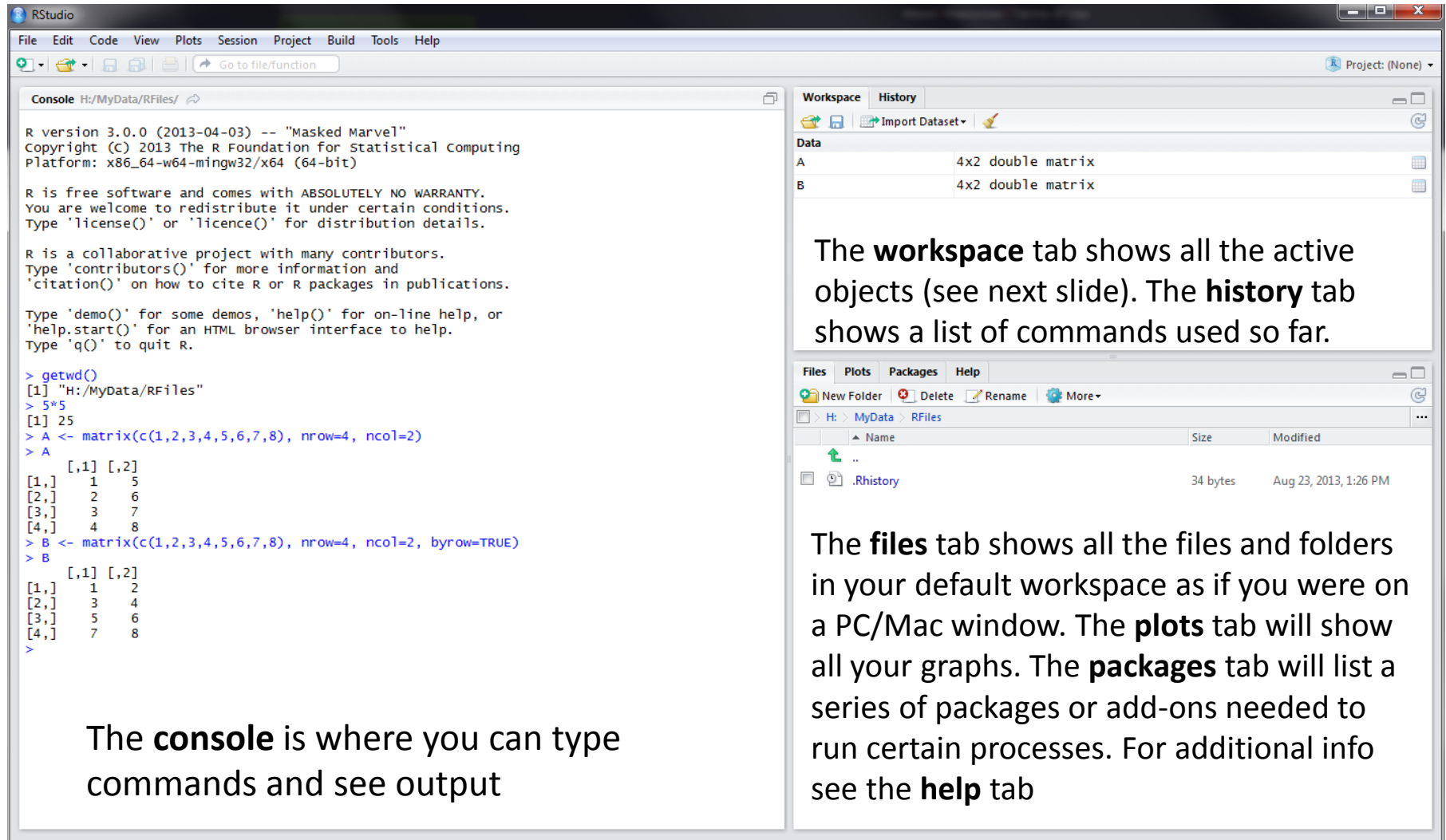
RStudio allows the user to run R in a more user-friendly environment. It is open-source (i.e. free) and available at <http://www.rstudio.com/>

For R related tutorials and/or resources see the following links:

<http://dss.princeton.edu/training/>

<http://libguides.princeton.edu/dss>

RStudio screen



The screenshot displays the RStudio application window. The top menu bar includes File, Edit, Code, View, Plots, Session, Project, Build, Tools, and Help. Below the menu is a toolbar with icons for file operations and a search bar labeled 'Go to file/function'. The main interface is divided into four panes: Console, Workspace, History, and Files. The Console pane on the left shows the R version (3.0.0), copyright information, and a series of commands and their outputs, including creating two matrices, A and B. The Workspace pane on the right shows the active objects A and B, both 4x2 double matrices. The History pane shows a list of commands used. The Files pane at the bottom shows the file structure of the current workspace, including a folder named .Rhistory.

Console H:/MyData/RFiles/

```
R version 3.0.0 (2013-04-03) -- "Masked Marvel"
Copyright (C) 2013 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.

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.

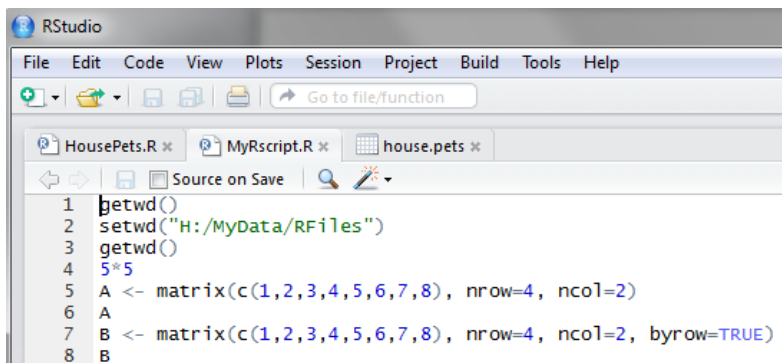
> getwd()
[1] "H:/MyData/RFiles"
> 5*5
[1] 25
> A <- matrix(c(1,2,3,4,5,6,7,8), nrow=4, ncol=2)
> A
      [,1] [,2]
[1,]    1    5
[2,]    2    6
[3,]    3    7
[4,]    4    8
> B <- matrix(c(1,2,3,4,5,6,7,8), nrow=4, ncol=2, byrow=TRUE)
> B
      [,1] [,2]
[1,]    1    2
[2,]    3    4
[3,]    5    6
[4,]    7    8
>
```

The **workspace** tab shows all the active objects (see next slide). The **history** tab shows a list of commands used so far.

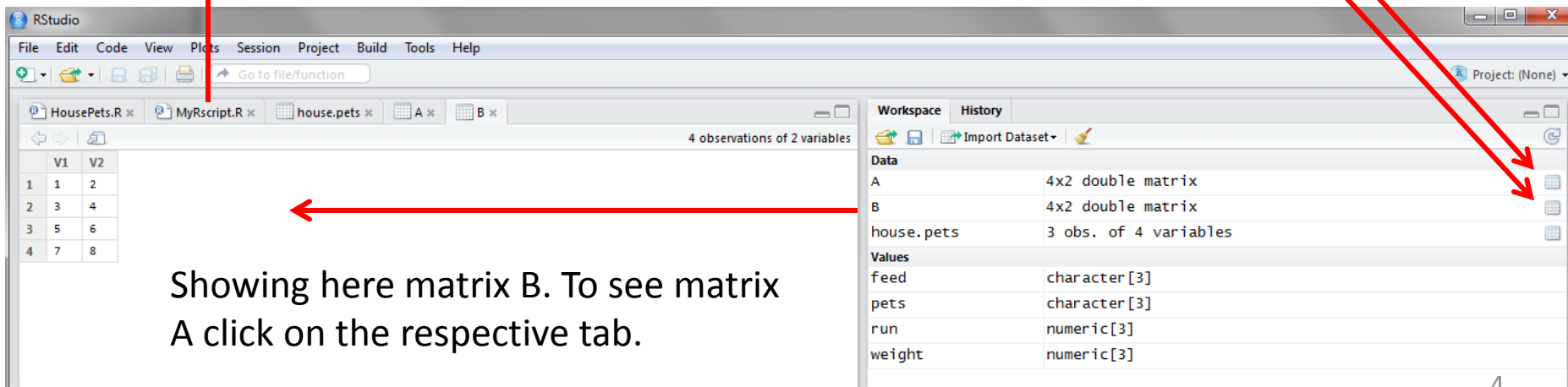
The **files** tab shows all the files and folders in your default workspace as if you were on a PC/Mac window. The **plots** tab will show all your graphs. The **packages** tab will list a series of packages or add-ons needed to run certain processes. For additional info see the **help** tab

Workspace tab (1)

The workspace tab stores any object, value, function or anything you create during your R session. In the example below, if you click on the dotted squares you can see the data on a screen to the left.



```
1 betwd()
2 setwd("H:/MyData/RFiles")
3 getwd()
4 5*5
5 A <- matrix(c(1,2,3,4,5,6,7,8), nrow=4, ncol=2)
6 A
7 B <- matrix(c(1,2,3,4,5,6,7,8), nrow=4, ncol=2, byrow=TRUE)
8 B
```



The screenshot shows the RStudio interface with the workspace and data viewer tabs active. The workspace tab lists the objects A, B, and house.pets. The data viewer tab shows the data for matrix B.

4 observations of 2 variables

	V1	V2
1	1	2
2	3	4
3	5	6
4	7	8

Showing here matrix B. To see matrix A click on the respective tab.

Workspace History

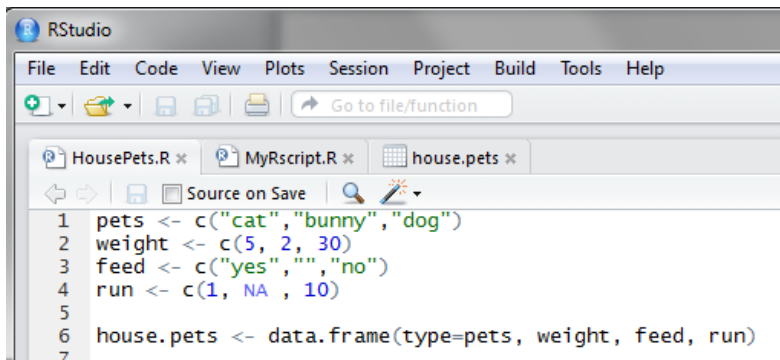
Object	Type
A	4x2 double matrix
B	4x2 double matrix
house.pets	3 obs. of 4 variables

Values

Variable	Type
feed	character[3]
pets	character[3]
run	numeric[3]
weight	numeric[3]

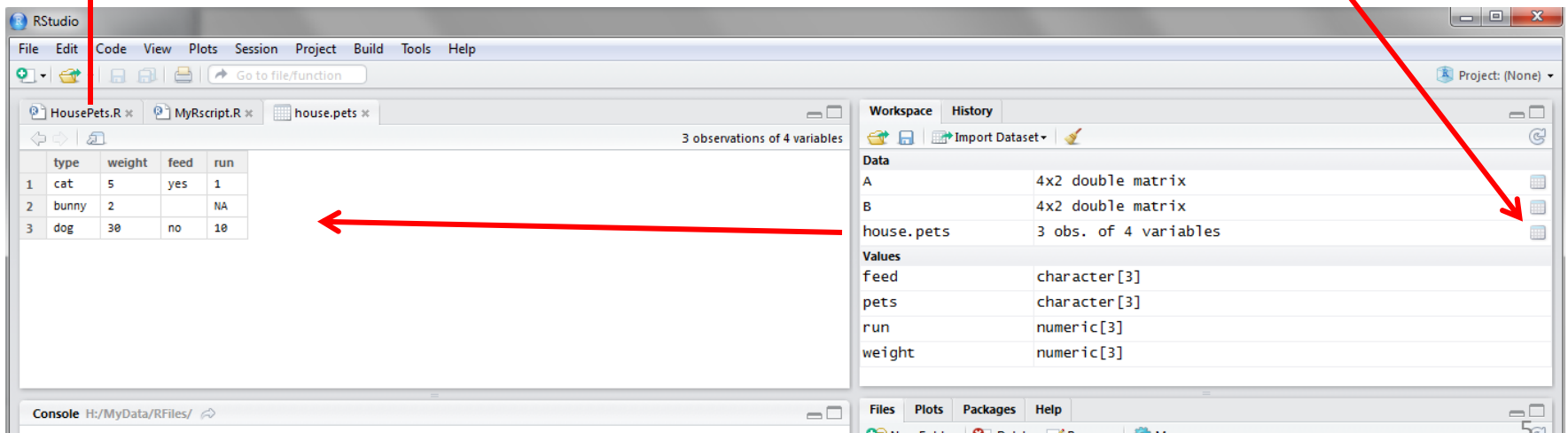
Workspace tab (2)

Here is another example on how the workspace looks like when more objects are added. Notice that the data frame `house.pets` is formed from different individual values or vectors.



```
1 pets <- c("cat","bunny","dog")
2 weight <- c(5, 2, 30)
3 feed <- c("yes","", "no")
4 run <- c(1, NA, 10)
5
6 house.pets <- data.frame(type=pets, weight, feed, run)
7
```

Click on the dotted square to look at the dataset in a spreadsheet form.



The RStudio interface shows the workspace tab with the following data:

	type	weight	feed	run
1	cat	5	yes	1
2	bunny	2		NA
3	dog	30	no	10

The Workspace tab on the right shows the following objects:

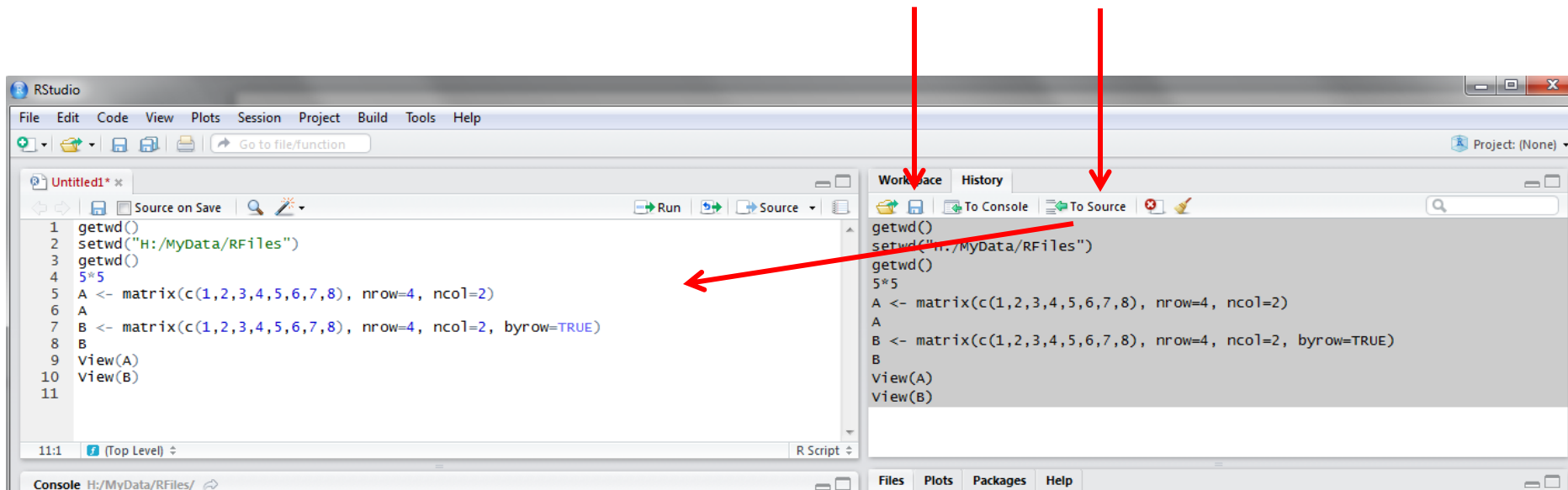
Object	Type
A	4x2 double matrix
B	4x2 double matrix
house.pets	3 obs. of 4 variables
feed	character [3]
pets	character [3]
run	numeric [3]
weight	numeric [3]

Red arrows indicate the flow from the code editor to the spreadsheet view and from the workspace tab to the spreadsheet view.

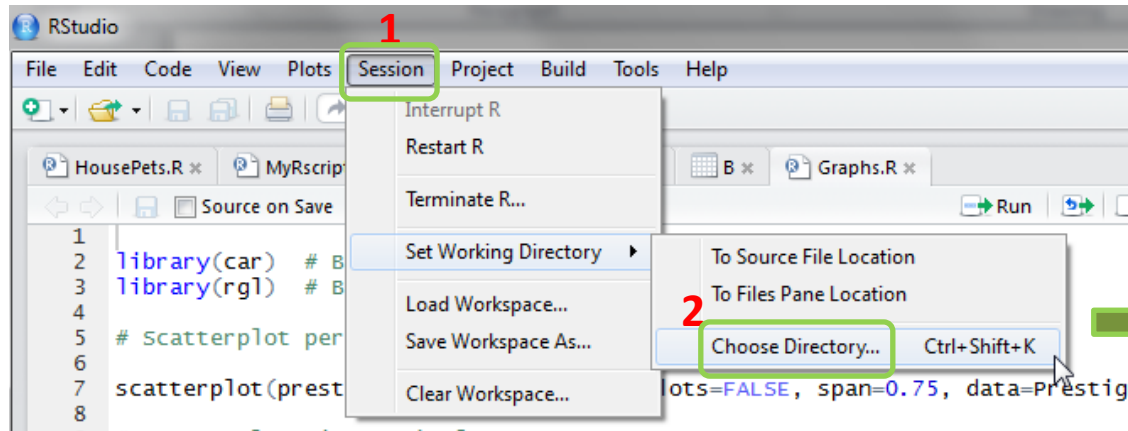
History tab

The history tab keeps a record of all previous commands. It helps when testing and running processes. Here you can either **save** the whole list or you can **select** the commands you want and send them to an R script to keep track of your work.

In this example, we select all and click on the “To Source” icon, a window on the left will open with the list of commands. Make sure to save the ‘untitled1’ file as an *.R script.



Changing the working directory



If you have different projects you can change the working directory for that session, see above. Or you can type:

```
# Shows the working directory (wd)
```

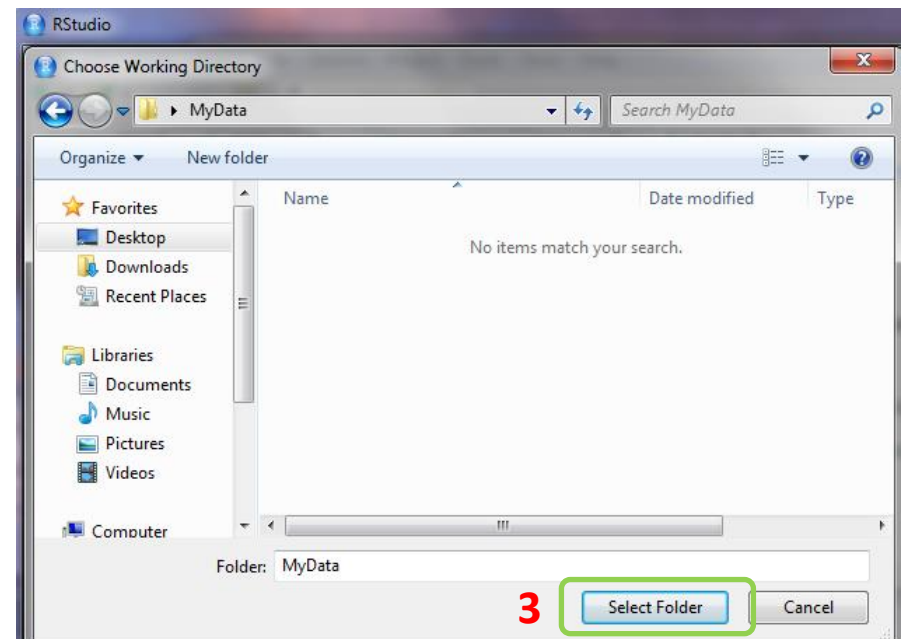
```
getwd()
```

```
# Changes the wd
```

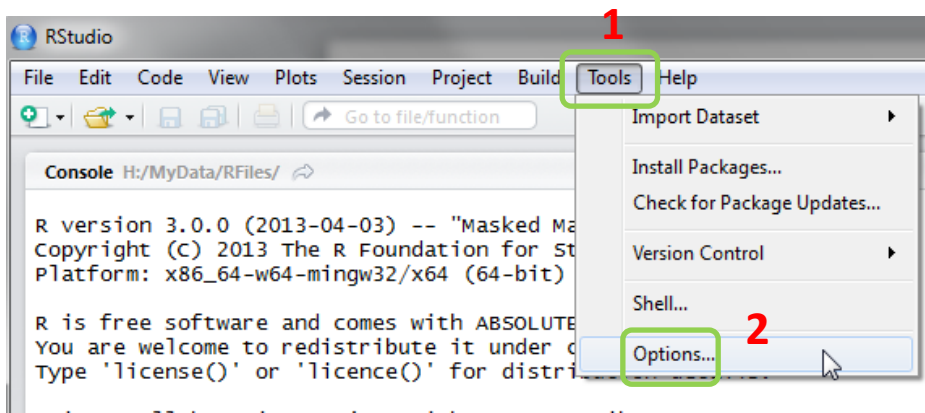
```
setwd("C:/myfolder/data")
```

More info see the following document:

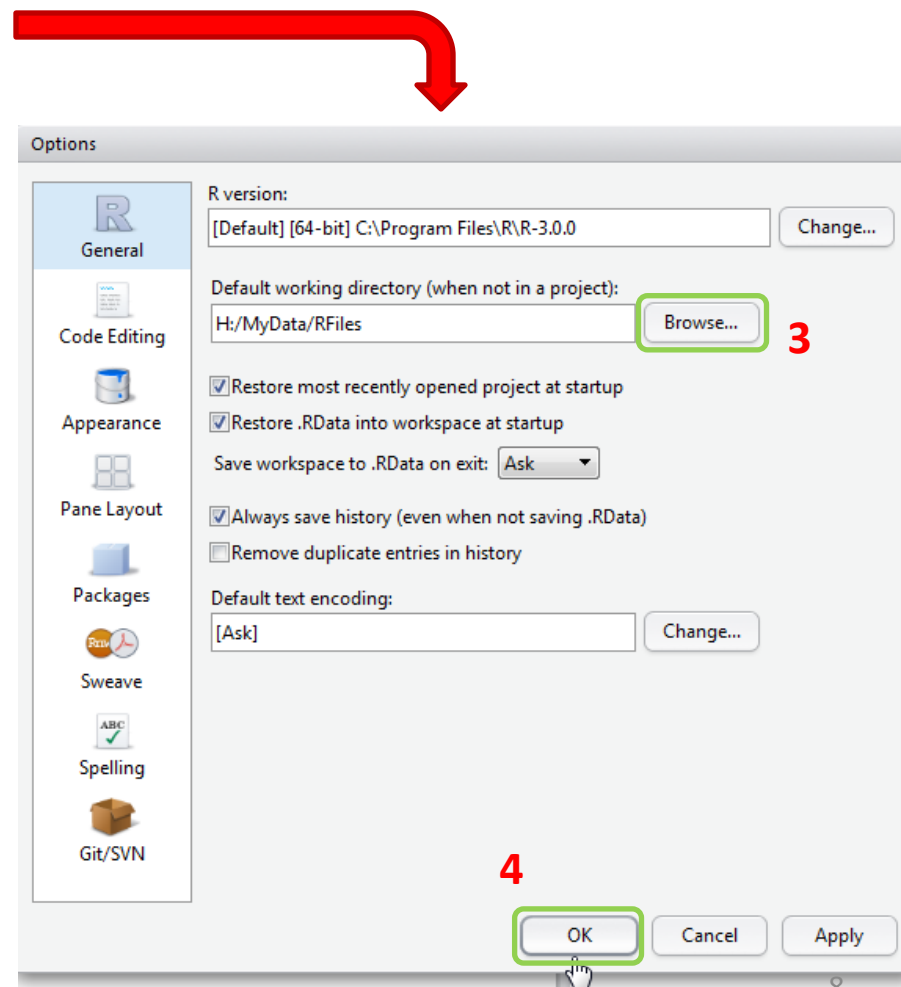
<http://dss.princeton.edu/training/RStata.pdf>



Setting a default working directory



Every time you open RStudio, it goes to a default directory. You can change the default to a folder where you have your datafiles so you do not have to do it every time. In the menu go to Tools->Options

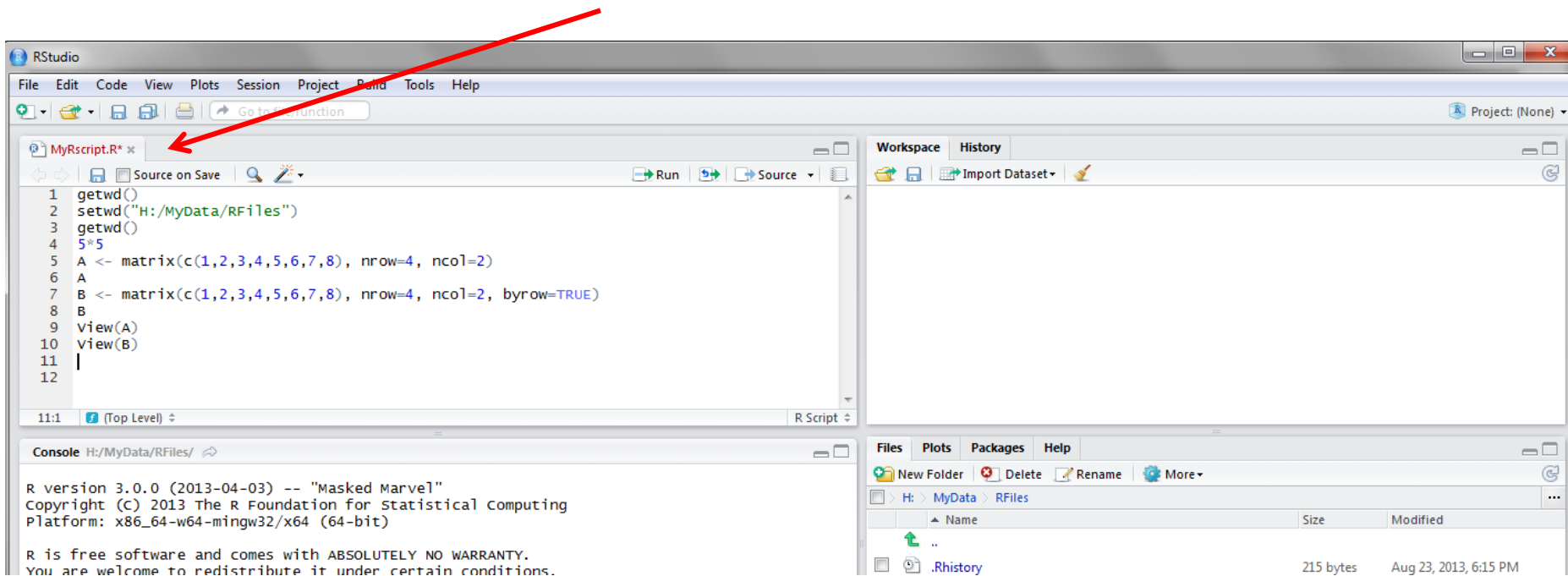


R script (1)

The usual Rstudio screen has four windows:

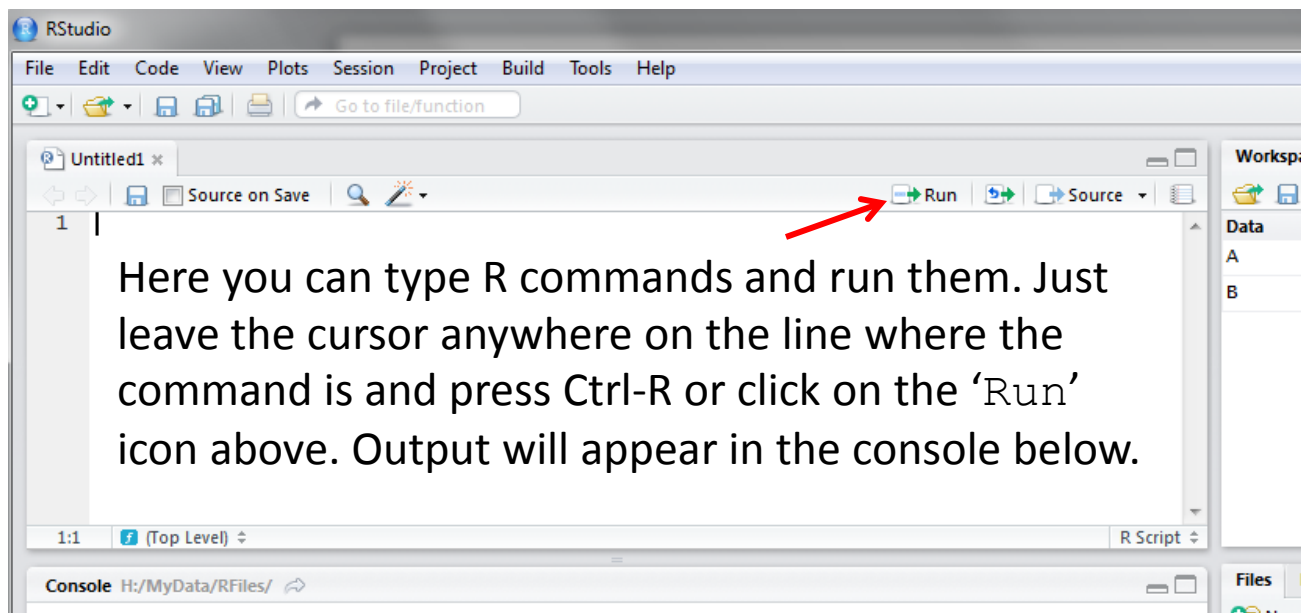
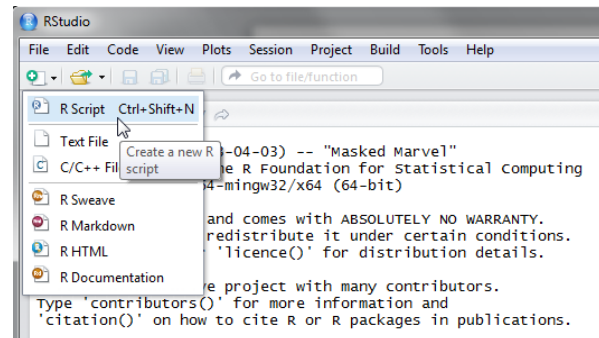
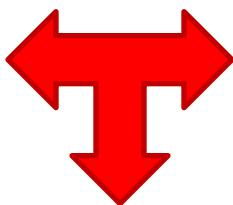
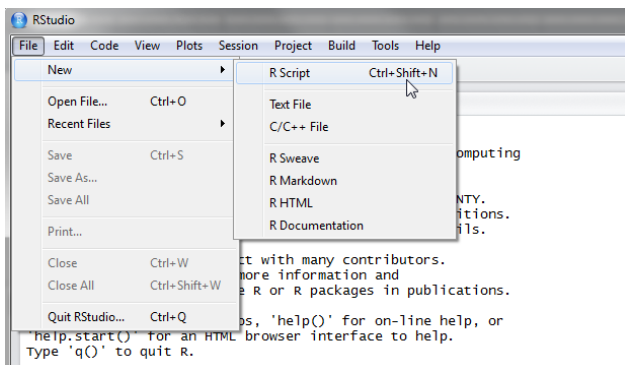
1. Console.
2. Workspace and history.
3. Files, plots, packages and help.
4. The R script(s) and data view.

The R script is where you keep a record of your work. For Stata users this would be like the do-file, for SPSS users is like the syntax and for SAS users the SAS program.



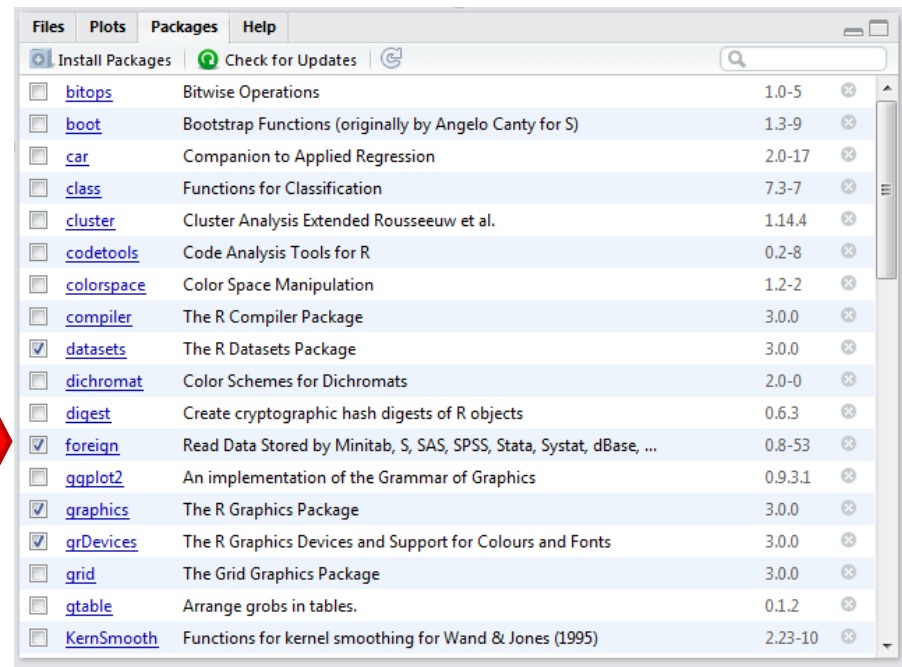
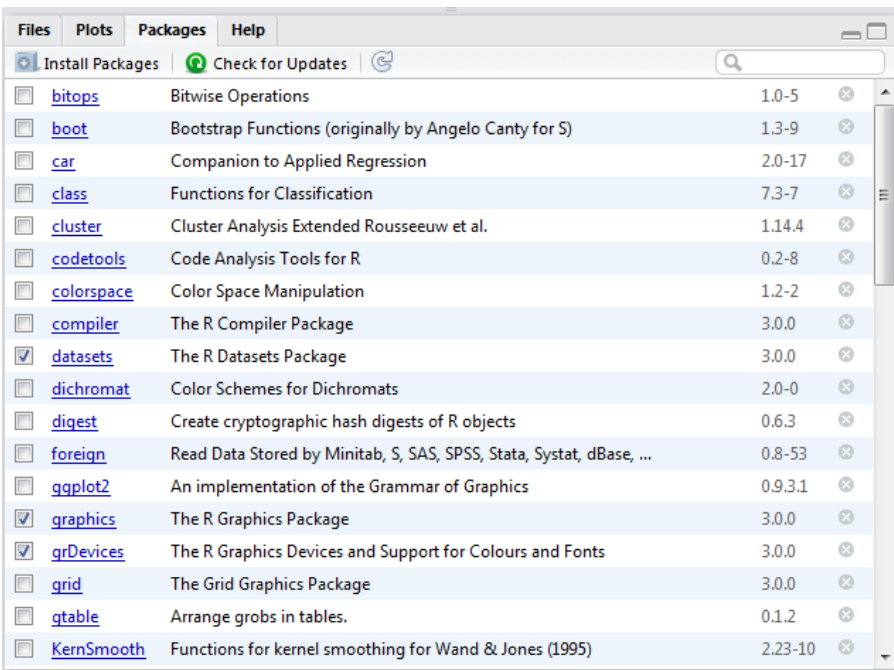
R script (2)

To create a new R script you can either go to `File -> New -> R Script`, or click on the icon with the “+” sign and select “R Script”, or simply press `Ctrl+Shift+N`. Make sure to save the script.



Packages tab

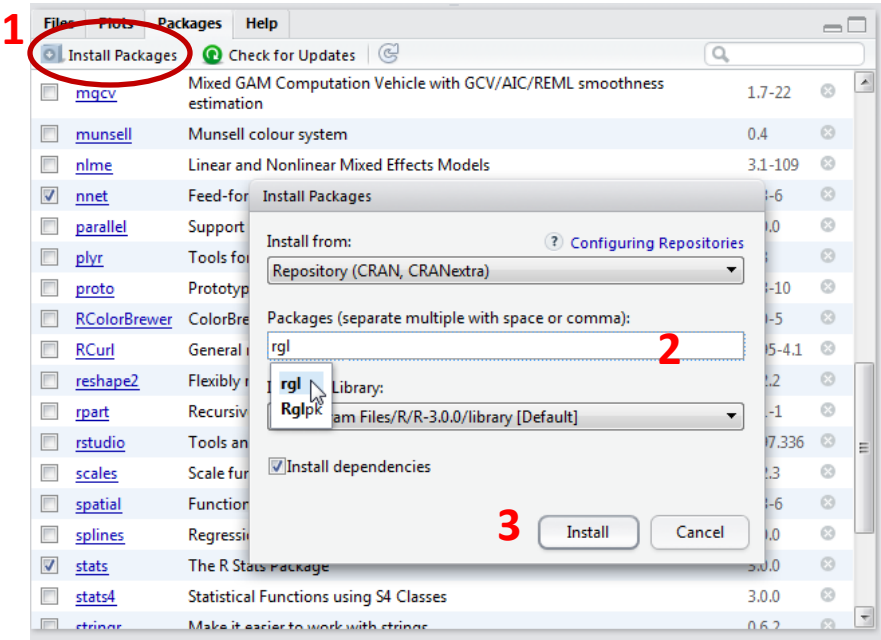
The package tab shows the list of add-ons included in the installation of RStudio. If checked, the package is loaded into R, if not, any command related to that package won't work, you will need select it. You can also install other add-ons by clicking on the 'Install Packages' icon. Another way to activate a package is by typing, for example, `library(foreign)`. This will automatically check the `--foreign` package (it helps bring data from proprietary formats like Stata, SAS or SPSS).



Installing a package

<input type="checkbox"/>	RCurl	General network (HTTP/FTP/...) client interface for R	1.95-4.1	✕
<input type="checkbox"/>	reshape2	Flexibly reshape data: a reboot of the reshape package.	1.2.2	✕
<input type="checkbox"/>	rpart	Recursive Partitioning	4.1-1	✕

Before



We are going to install the package – `rgl` (useful to plot 3D images). It does not come with the original R install.

Click on “Install Packages”, write the name in the pop-up window and click on “Install”.

After

<input type="checkbox"/>	RCurl	General network (HTTP/FTP/...) client interface for R	1.95-4.1	✕
<input type="checkbox"/>	reshape2	Flexibly reshape data: a reboot of the reshape package.	1.2.2	✕
<input type="checkbox"/>	rgl	3D visualization device system (OpenGL)	0.93.952	✕
<input type="checkbox"/>	rpart	Recursive Partitioning	4.1-1 12	✕

Plots tab (1)

RStudio interface showing the **Plots** tab. The script in the editor includes the following code:

```
1 library(car) # By John Fox and Sanford Weisberg
2 library(rgl) # By Daniel Adler and Duncan Murdoch
3
4 # Scatterplot per group
5
6 scatterplot(prestige ~ income|type, boxplots=FALSE, span=0.75, data=Prestige)
7
8 # Scatterplots in matrix form
9
10 scatterplotMatrix(~ prestige + income + education, span=0.7, data=Prestige)
11
12 # 3D graph, scatter3d is from the --car package. It will open in a separate window.
13
14 scatter3d(prestige ~ income + education, id.n=3, data=Duncan)
15
```

The console shows the execution of the command on line 7:

```
> scatterplot(prestige~income|type, boxplots=FALSE, span=0.75, data=Prestige)
>
```

The **Plots** tab displays a scatterplot of **prestige** (y-axis) versus **income** (x-axis), faceted by **type** (bc, prof, wc). The plot shows three distinct groups of data points, each with a fitted smoothing line. A red arrow points from the text on the left to the plot.

Plots tab (2)

The screenshot shows the RStudio interface with the Plots tab selected. The console on the left displays the following R code:

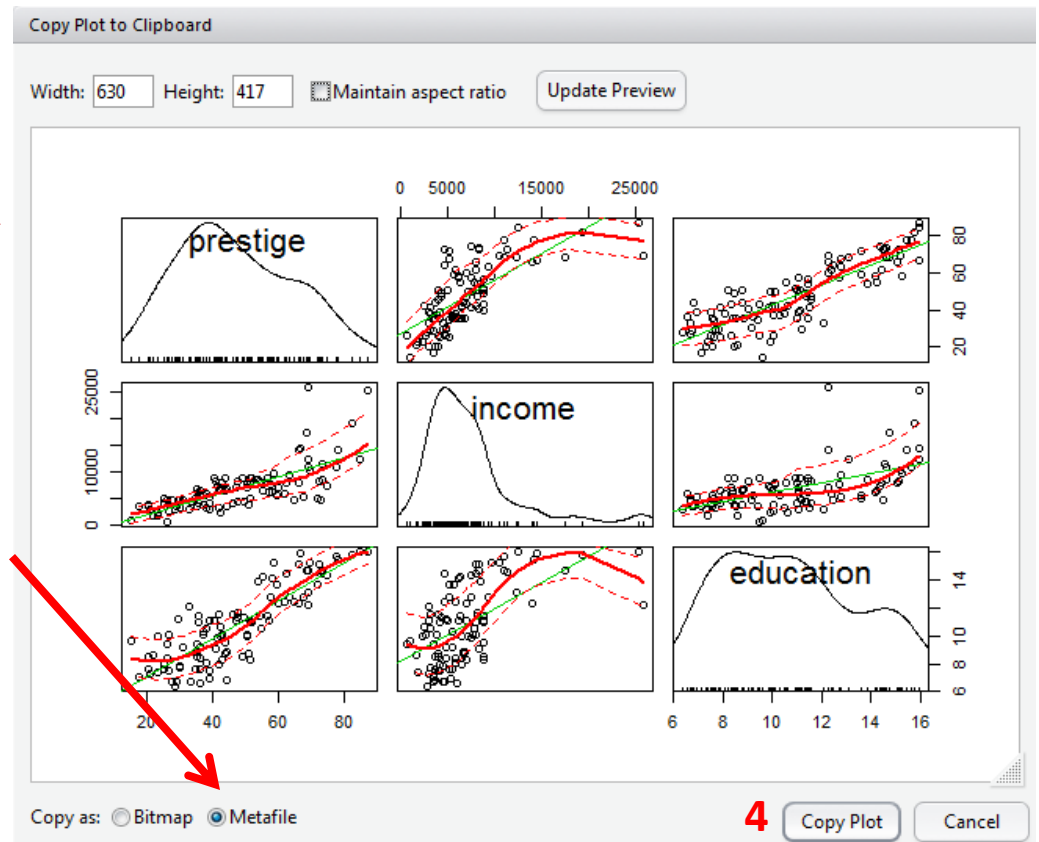
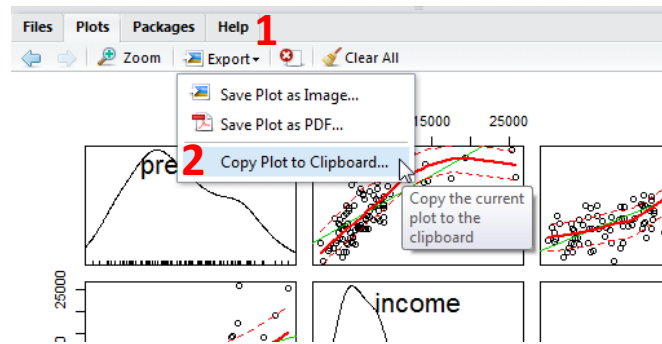
```
1 library(car) # By John Fox and Sanford Weisberg
2 library(rgl) # By Daniel Adler and Duncan Murdoch
3
4 # Scatterplot per group
5
6 scatterplot(prestige ~ income|type, boxplots=FALSE, span=0.75, data=Prestige)
7
8 # Scatterplots in matrix form
9
10 scatterplotMatrix(~ prestige + income + education, span=0.7, data=Prestige)
11
12 # 3D graph, scatter3d is from the --car package. It will open in a separate window.
13
14 scatter3d(prestige ~ income + education, id.n=3, data=Duncan)
```

The Plots tab on the right displays a 3x3 grid of plots for the variables prestige, income, and education. The diagonal elements are density plots for each variable. The off-diagonal elements are scatter plots showing the relationships between pairs of variables. Each scatter plot includes a solid red line for the linear fit and dashed green lines for the non-linear fits. A red arrow points from the console to the left arrow icon in the Plots tab toolbar, which is used to switch between the first and second plots in the sequence.

Here there is a second graph (see line 11 above). If you want to see the first one, click on the left-arrow icon.

Plots tab (3) – Graphs export

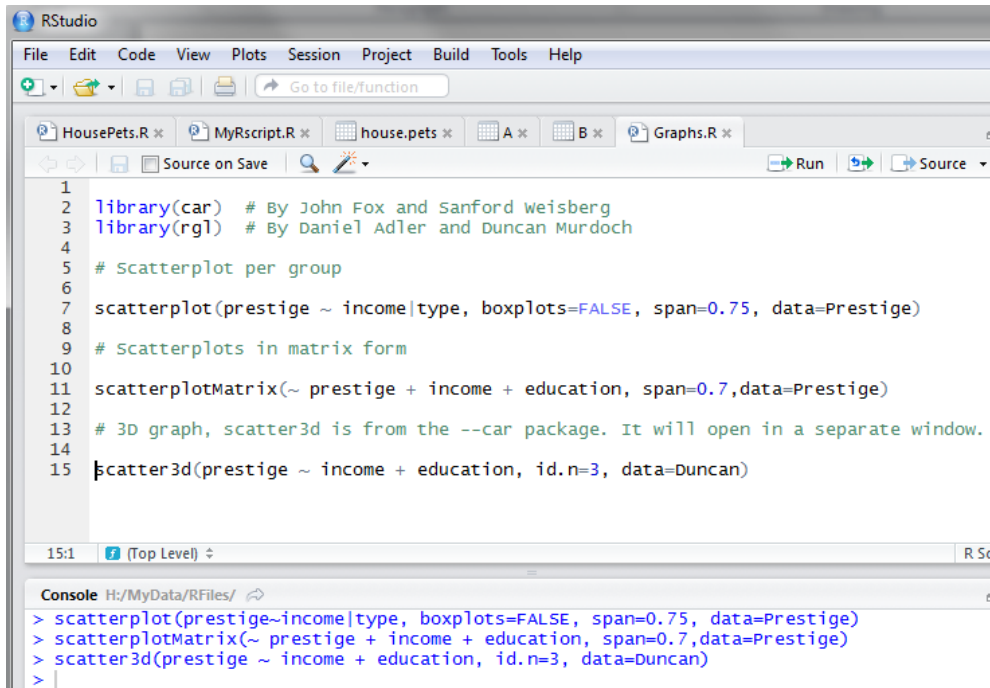
To extract the graph, click on “Export” where you can save the file as an image (PNG, JPG, etc.) or as PDF, these options are useful when you only want to share the graph or use it in a LaTeX document. Probably, the easiest way to export a graph is by copying it to the clipboard and then paste it directly into your Word document.



3 Make sure to select 'Metafile'

5 Paste it into your Word document

3D graphs



```
1 library(car) # By John Fox and Sanford Weisberg
2 library(rgl) # By Daniel Adler and Duncan Murdoch
3
4 # Scatterplot per group
5
6 scatterplot(prestige ~ income|type, boxplots=FALSE, span=0.75, data=Prestige)
7
8 # Scatterplots in matrix form
9
10 scatterplotMatrix(~ prestige + income + education, span=0.7, data=Prestige)
11
12 # 3D graph, scatter3d is from the --car package. It will open in a separate window.
13
14
15 scatter3d(prestige ~ income + education, id.n=3, data=Duncan)
```

Console H:/MyData/RFiles/

```
> scatterplot(prestige~income|type, boxplots=FALSE, span=0.75, data=Prestige)
> scatterplotMatrix(~ prestige + income + education, span=0.7, data=Prestige)
> scatter3d(prestige ~ income + education, id.n=3, data=Duncan)
```

3D graphs will display on a separate screen (see line 15 above). You won't be able to save it, but after moving it around, once you find the angle you want, you can screenshot it and paste it to you Word document.

