

Introduction to R Commander

1. Get R and Rcmdr to run
2. Familiarize yourself with Rcmdr
3. Look over Rcmdr metadata (Fox, 2005)
4. Start doing stats / plots with Rcmdr

Tasks

1. Clear Workspace and History.
2. Install and run the Rcmdr package.
3. Manually enter and edit data files
(LecturerData and OnlyLecturer)
4. Import RData file in three formats:
lxs, txt, csv
5. Save data file

Installing Rcmdr Package

1. Install Rcmdr package
2. Load Rcmdr package
3. A new GUI window will pop-up; which will add functionality to R via menu-driven functions

NOTE: You can install packages using the R console or R Studio. I show you how to do the installation both ways, but you will only need to do it once.

HINT: I suggest using R Studio.

Question 1a

Figure out how to import the xls file: **RData1.xls**

Explain how you managed to import this file.

Report the R commands you used

Question 1b

Figure out how to import the txt file: **RData2.txt**

Explain how you managed to import this file.

Report the R commands you used

Question 1c

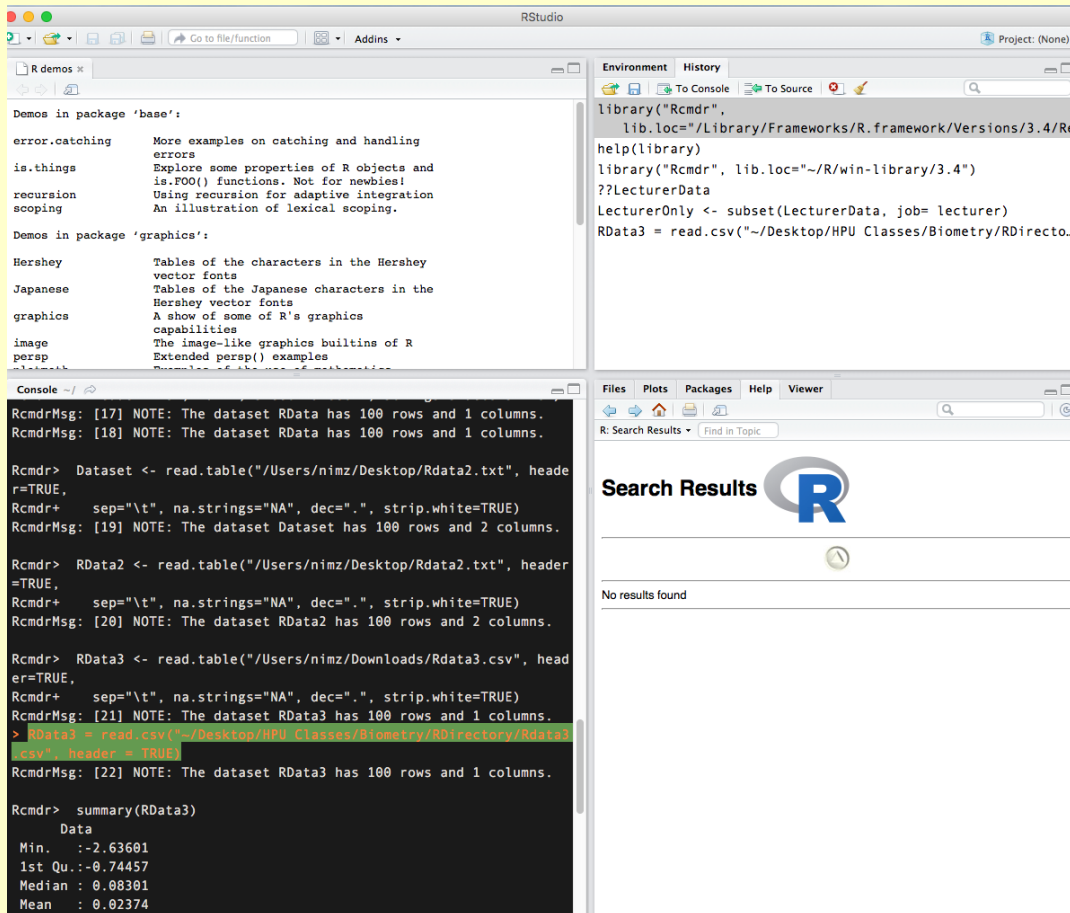
Figure out how to import this csv file: **RData3.csv**

Explain how you managed to import this file.

Report the R commands you used

Answers

Utilized command for csv from R textbook for mac. Initially imported same way as .txt and excel files and got same response from R, so it seems like the other way worked too



The screenshot shows the RStudio environment with the following content:

```
Demos in package 'base':
error.catching More examples on catching and handling errors
is.things Explore some properties of R objects and is.FOO() functions. Not for newbies!
recursion Using recursion for adaptive integration
scoping An illustration of lexical scoping.

Demos in package 'graphics':
Hershey Tables of the characters in the Hershey vector fonts
Japanese Tables of the Japanese characters in the Hershey vector fonts
graphics A show of some of R's graphics capabilities
image The image-like graphics builtins of R
persp Extended persp() examples

Environment History
library("Rcmdr",
lib.loc="/Library/Frameworks/R.framework/Versions/3.4/Re
help(library)
library("Rcmdr", lib.loc="/R/win-library/3.4")
??LecturerData
LecturerOnly <- subset(LecturerData, job= lecturer)
RData3 = read.csv("~/Desktop/HPU Classes/Biometry/RDirecto...

Files Plots Packages Help Viewer
R: Search Results Find in Topic
Search Results R
No results found

RcmdrMsg: [17] NOTE: The dataset RData has 100 rows and 1 columns.
RcmdrMsg: [18] NOTE: The dataset RData has 100 rows and 1 columns.

Rcmdr> Dataset <- read.table("/Users/nimz/Desktop/Rdata2.txt", header=TRUE,
Rcmdr+ sep="\t", na.strings="NA", dec=".", strip.white=TRUE)
RcmdrMsg: [19] NOTE: The dataset Dataset has 100 rows and 2 columns.

Rcmdr> RData2 <- read.table("/Users/nimz/Desktop/Rdata2.txt", header=TRUE,
Rcmdr+ sep="\t", na.strings="NA", dec=".", strip.white=TRUE)
RcmdrMsg: [20] NOTE: The dataset RData2 has 100 rows and 2 columns.

Rcmdr> RData3 <- read.table("/Users/nimz/Downloads/Rdata3.csv", header=TRUE,
Rcmdr+ sep="\t", na.strings="NA", dec=".", strip.white=TRUE)
RcmdrMsg: [21] NOTE: The dataset RData3 has 100 rows and 1 columns.
> RData3 = read.csv("~/Desktop/HPU Classes/Biometry/RDirectory/Rdata3.csv", header=TRUE)
RcmdrMsg: [22] NOTE: The dataset RData3 has 100 rows and 1 columns.

Rcmdr> summary(RData3)
Data
Min. : -2.63601
1st Qu.: -0.74457
Median : 0.08301
Mean : 0.02374
```



Answers

I opened the “readr” package first and used this command to import the csv file (assigned the file to the name “RData3”):

```
library(readr)  
RData3 <- read_csv("C:/Users/Angelica  
Moua/Desktop/RData3_.csv")
```


Answers

The command for R to be able to read a csv file is:

```
Rdata3 = read.csv("Rdata3.csv")
```

```
> Rdata5 = read.csv("Rdata3.csv")
```

```
Warning in file(file, "rt") : cannot open file  
'Rdata3.csv': No such file or directory Error in  
file(file, "rt") : cannot open the connection
```

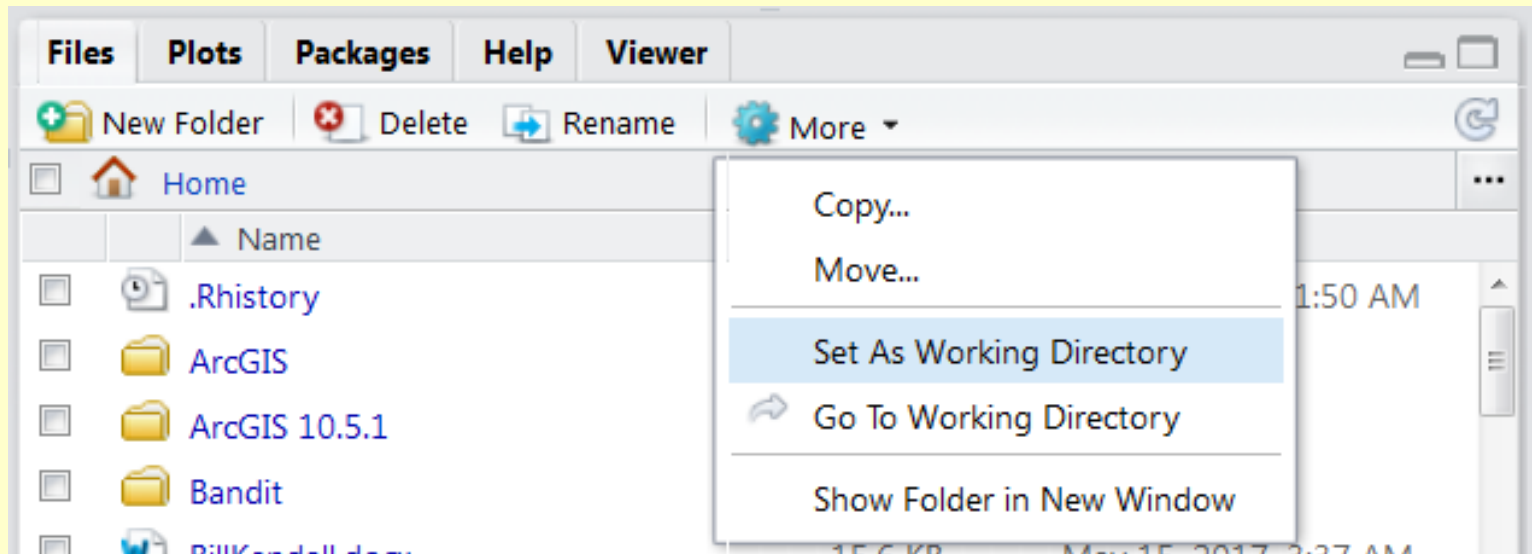
Remember:

R is always pointed at a **directory** on your computer. You can find out which **directory** by running the **getwd** (get working directory) function; this function has no arguments. To **change** your **working directory**, use **setwd** and specify the path to the desired **folder**.

Troubleshooting

Checking Working Directory: `getwd()`

Setting Working Directory: `setwd()`



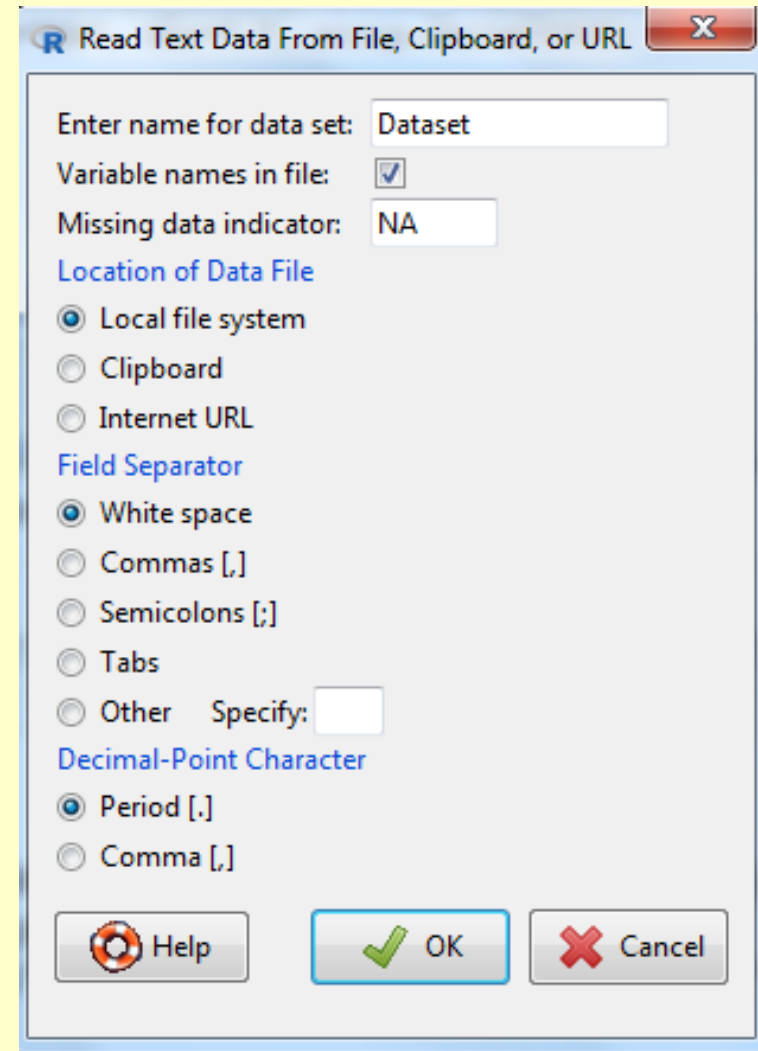
```
> setwd("~/R/working")  
> Rdata5 = read.csv("Rdata3.csv")
```

Answers

I was able to import the RData3.csv file by going to:

Data -> Import Data -> from text file, clipboard, or URL...

I selected “commas [,]” as the field separator, left the rest as the default and hit OK.



```
> RData4 <-  
  read.table("C:/Assignment#2/Rdata2_3.csv",  
            header=TRUE, sep="," , na.strings="NA", dec=".",  
            strip.white=TRUE)
```

Answers

```
> RData4 <-  
  read.table("C:/Assignment#2/Rdata2_3.csv",  
            header=TRUE, sep="," , na.strings="NA", dec=".",  
            strip.white=TRUE)
```

`read.table(file)` reads a file in table format and creates a data frame from it; the default separator `sep=""` is any whitespace;

use `header=TRUE` to read the first line as a header of column names; use `as.is=TRUE` to prevent character vectors from being converted to factors; use `comment.char=""` to prevent `"#"` from being interpreted as a comment; use `skip=n` to skip `n` lines before reading data; see the help for options on row naming, NA treatment, and others `read.csv("filename",header=TRUE)` id. but with defaults set for reading comma-delimited files

<https://cran.r-project.org/doc/contrib/Short-refcard.pdf>

Answers

I searched in the Short-refcard.pdf for csv data and found a Command. I put the following in R console and it worked!

`read.csv("filename",header=TRUE)` id. but with defaults set for reading comma-delimited files

```
> RData5 <- read.csv("Rdata3.csv",header=TRUE)
```

<https://cran.r-project.org/doc/contrib/Short-refcard.pdf>

Question 2: Subsetting Data

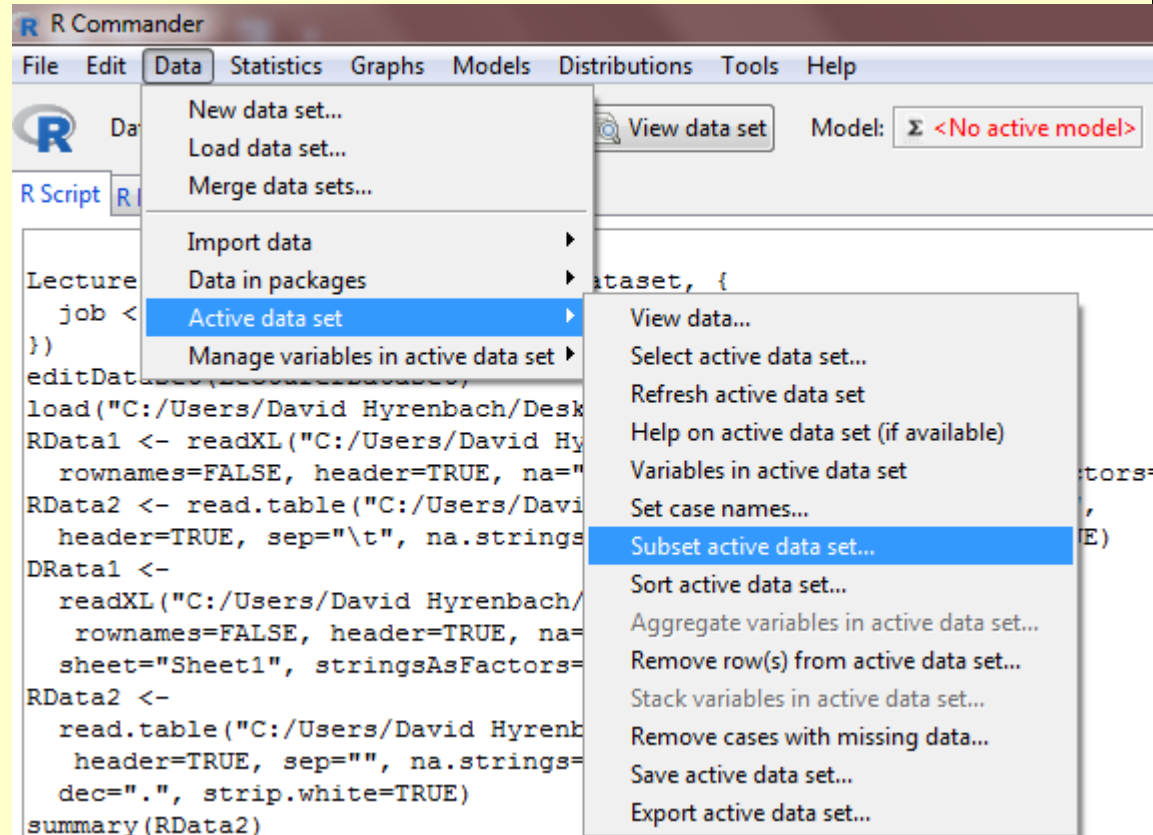
Make `LecturerData` the active file

Extract records where "job" is Lecturer, and set aside records where "job" is Researcher.

Use `subset` command:

`Data / Active Dataset / Subset Active Dataset`

Save the new dataframe you created. Call it: `LecturerOnly`



Answers

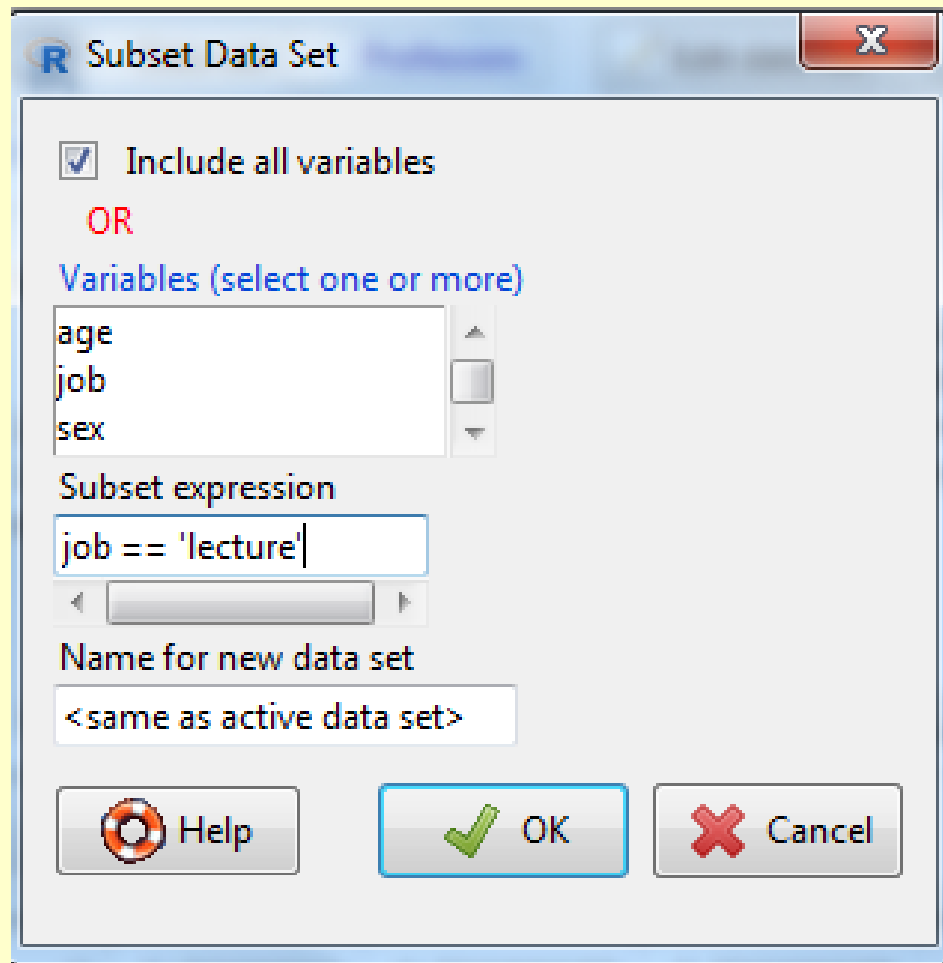
I searched in the Short-refcard.pdf for csv data and found a Command. I put the following in the R console and it worked!

subset(x, ...) returns a selection of x with respect to criteria (... , typically comparisons: x\$V1 < 10); if x is a data frame, the option select gives the variables to be kept or dropped using a minus sign

```
lecturers  
<- subset(professors, subset=job == 'lecturer')
```

<https://cran.r-project.org/doc/contrib/Short-refcard.pdf>

Answers



<https://www.youtube.com/watch?v=ImRQIgvjJ2Y>

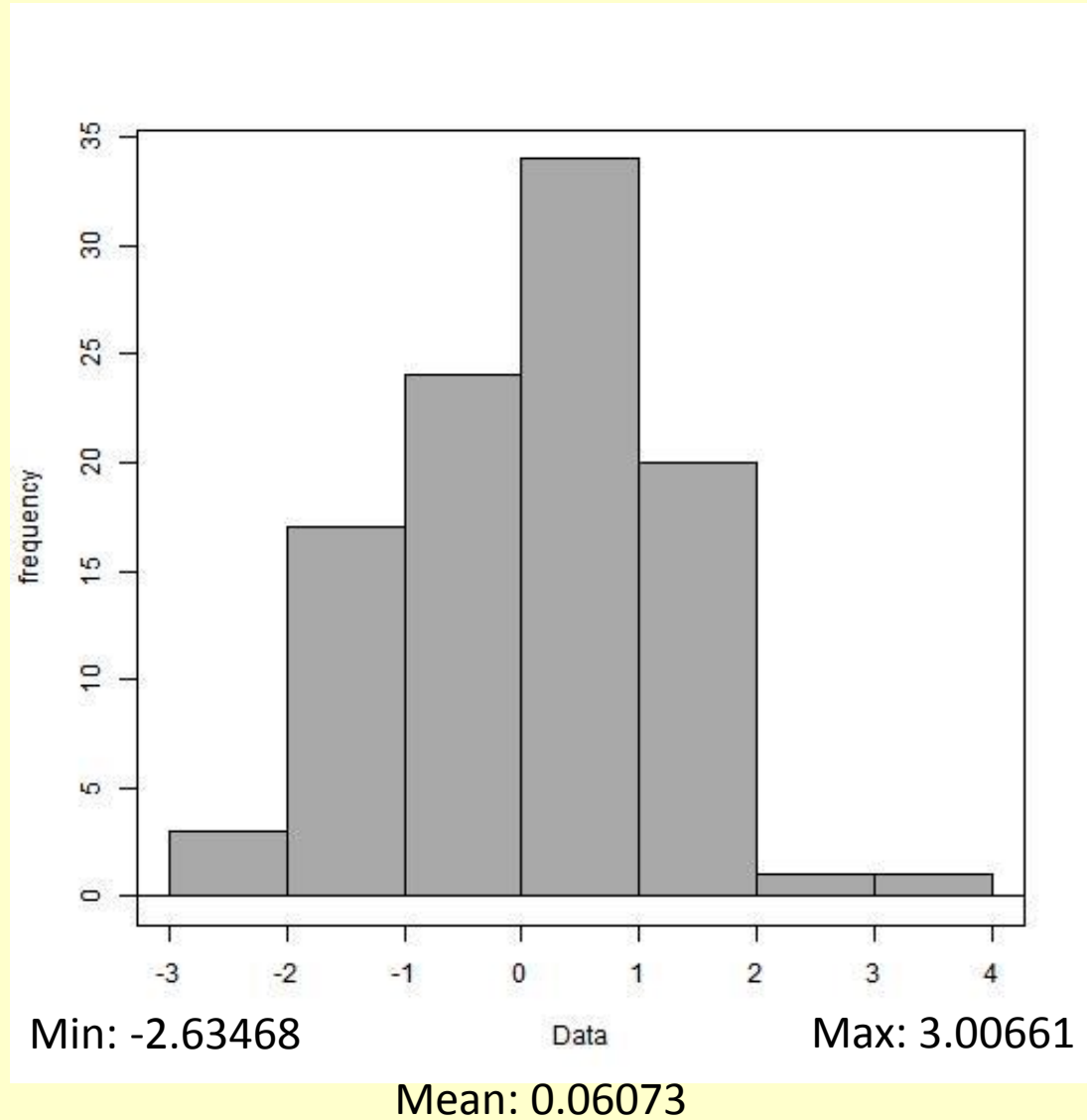
Question 3

Enter the following information, from the three datasets:
RData1, RData2, RData3

Summary Statistics	RData 1	RData2	RData3
Min	-2.63468	-2.259221	-2.63601
1 st Quartile	-0.84449	-0.592752	-0.74457
Median	0.12613	-0.062743	0.08301
Mean	0.06073	-0.004115	0.02374
3 rd Quartile	0.80002	0.532207	0.67735
Max	3.00661	2.076910	3.51744

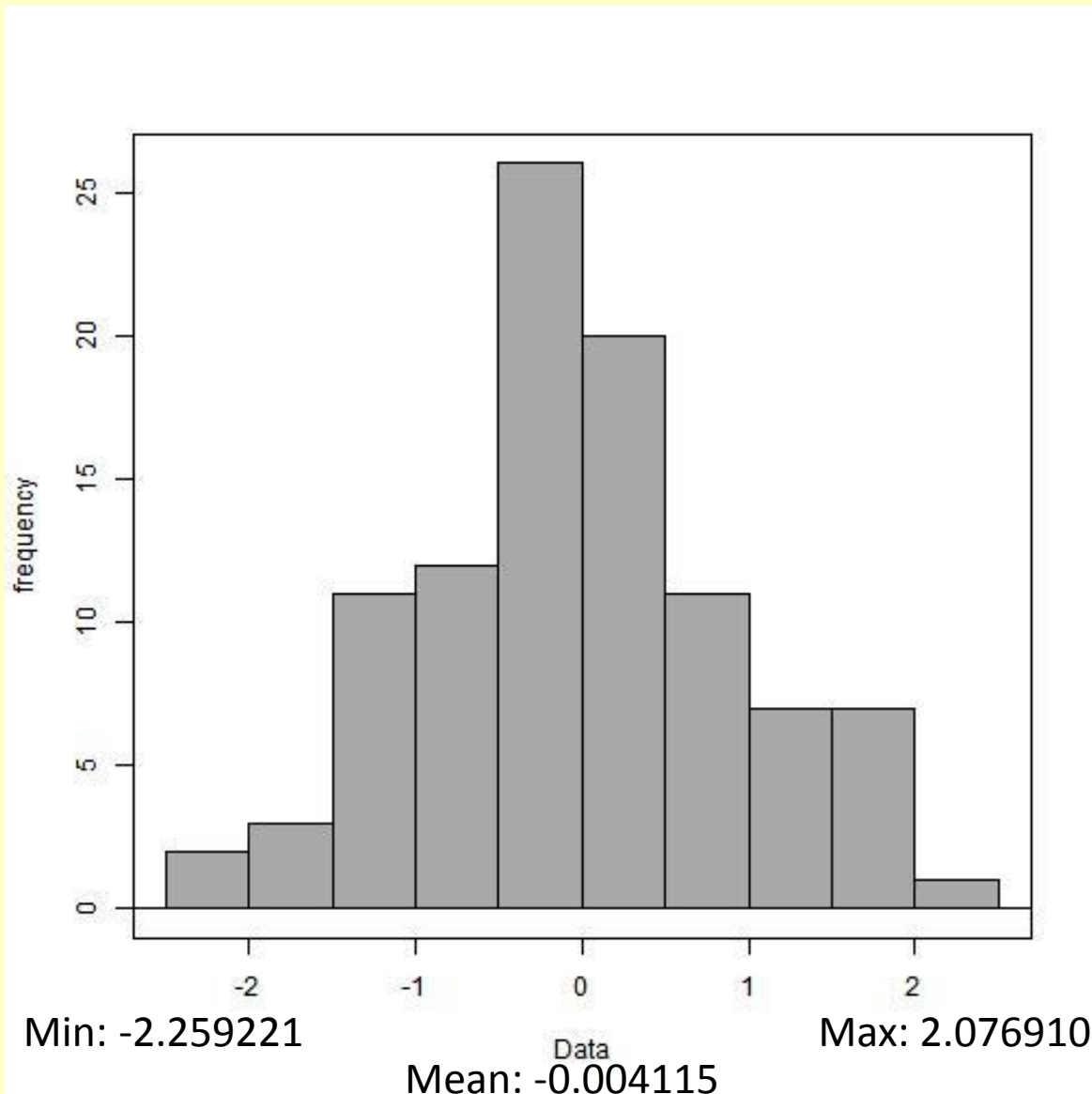
Answers

RData1



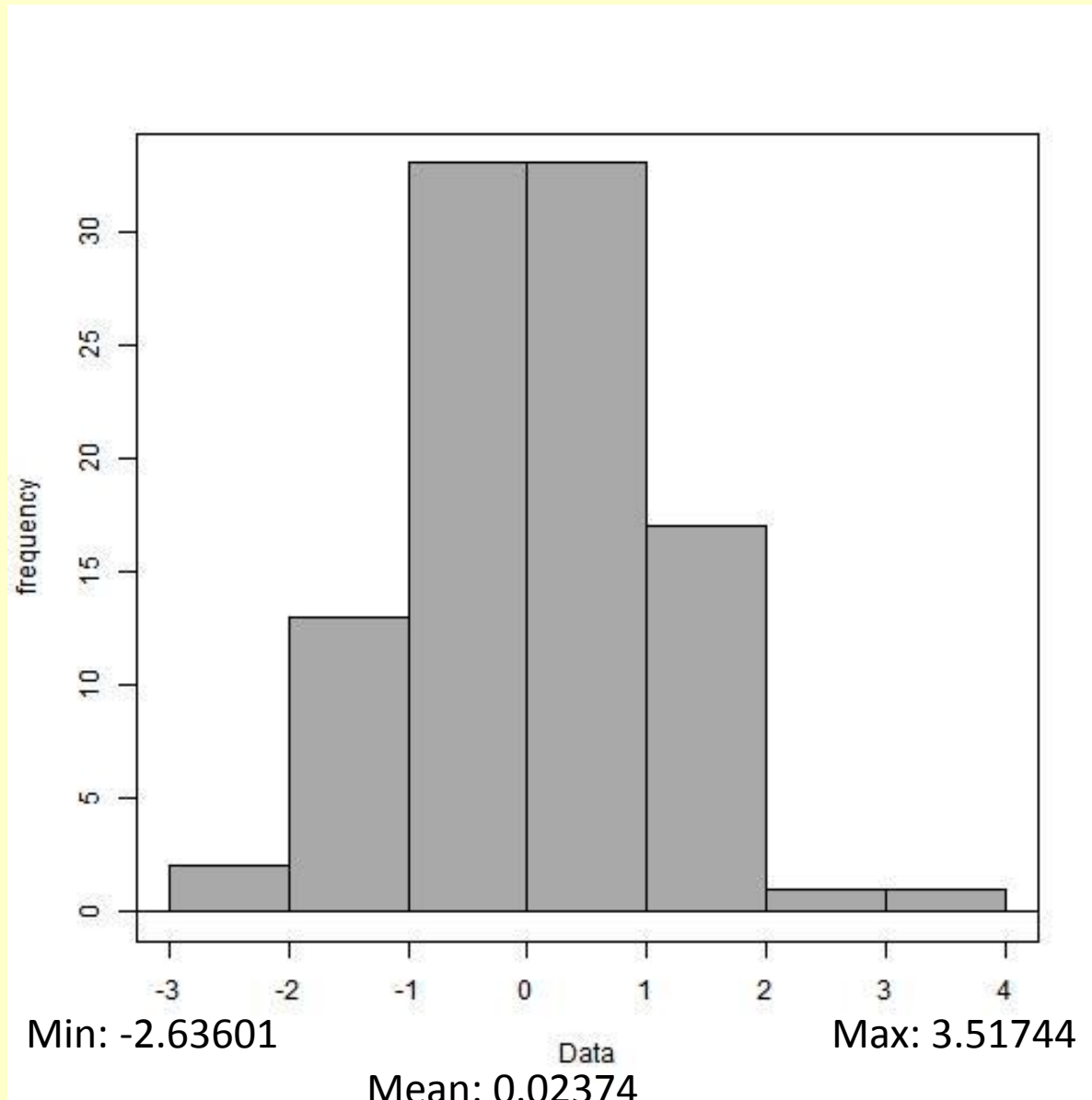
Answers

RData2



Answers

RData3



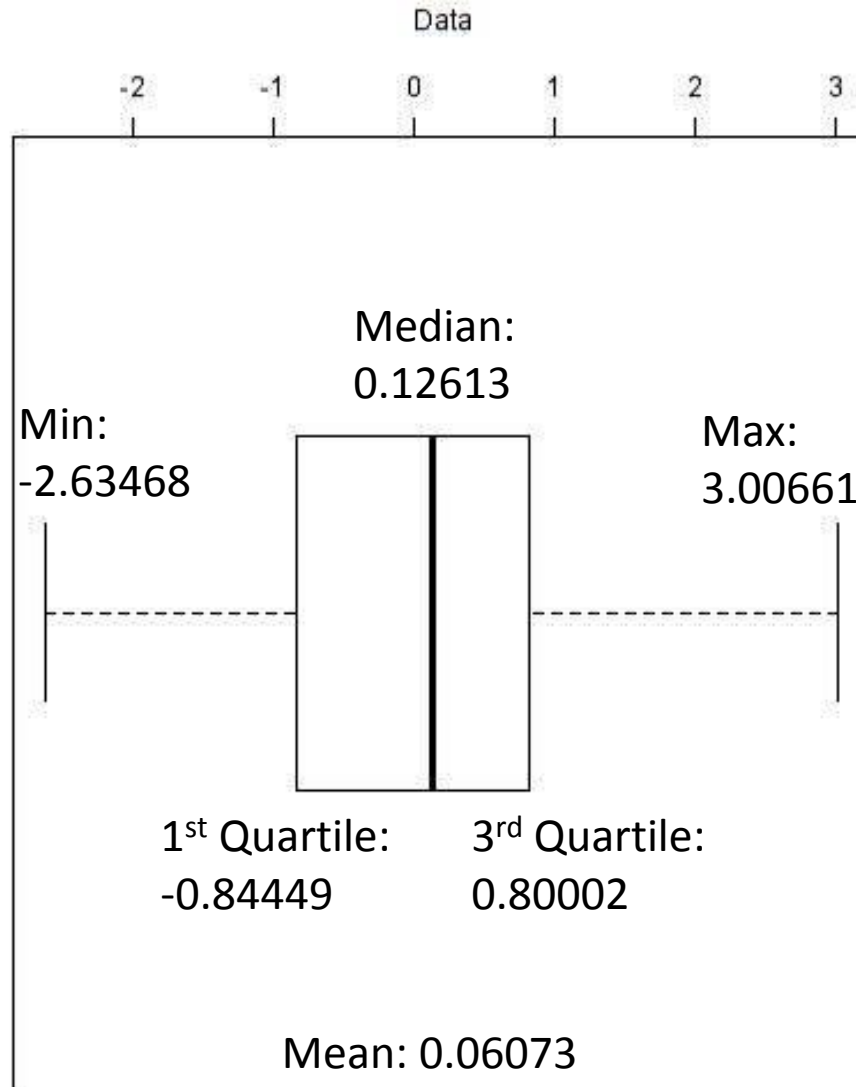
Questions / Answers

6. Create, save and paste three box plots: one for each of the three datasets: RData1, RData2, RData3.

Paste each plot in a separate slide and make sure you report the summary statistics you calculated in question 4.

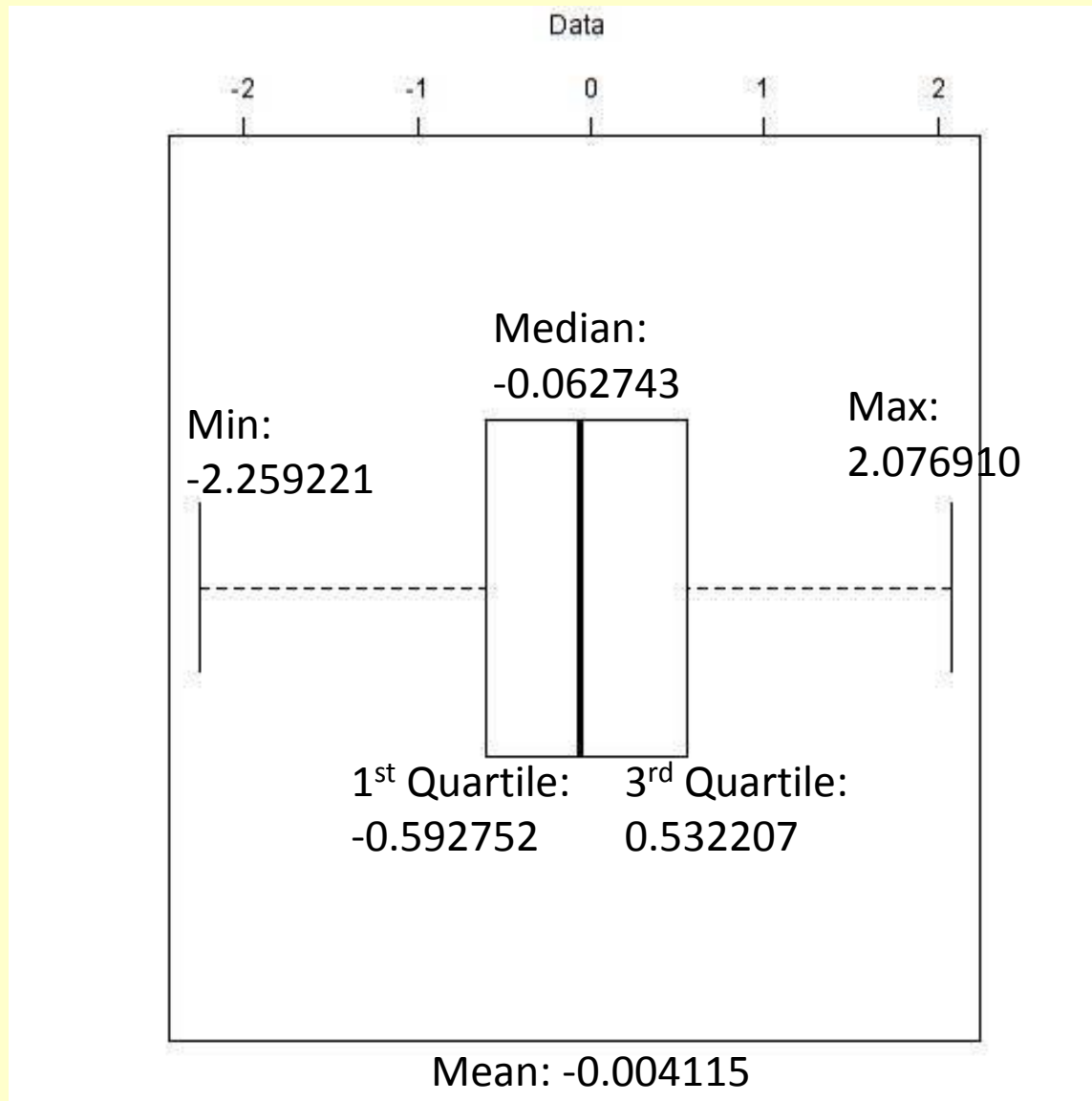
Answers

RData1



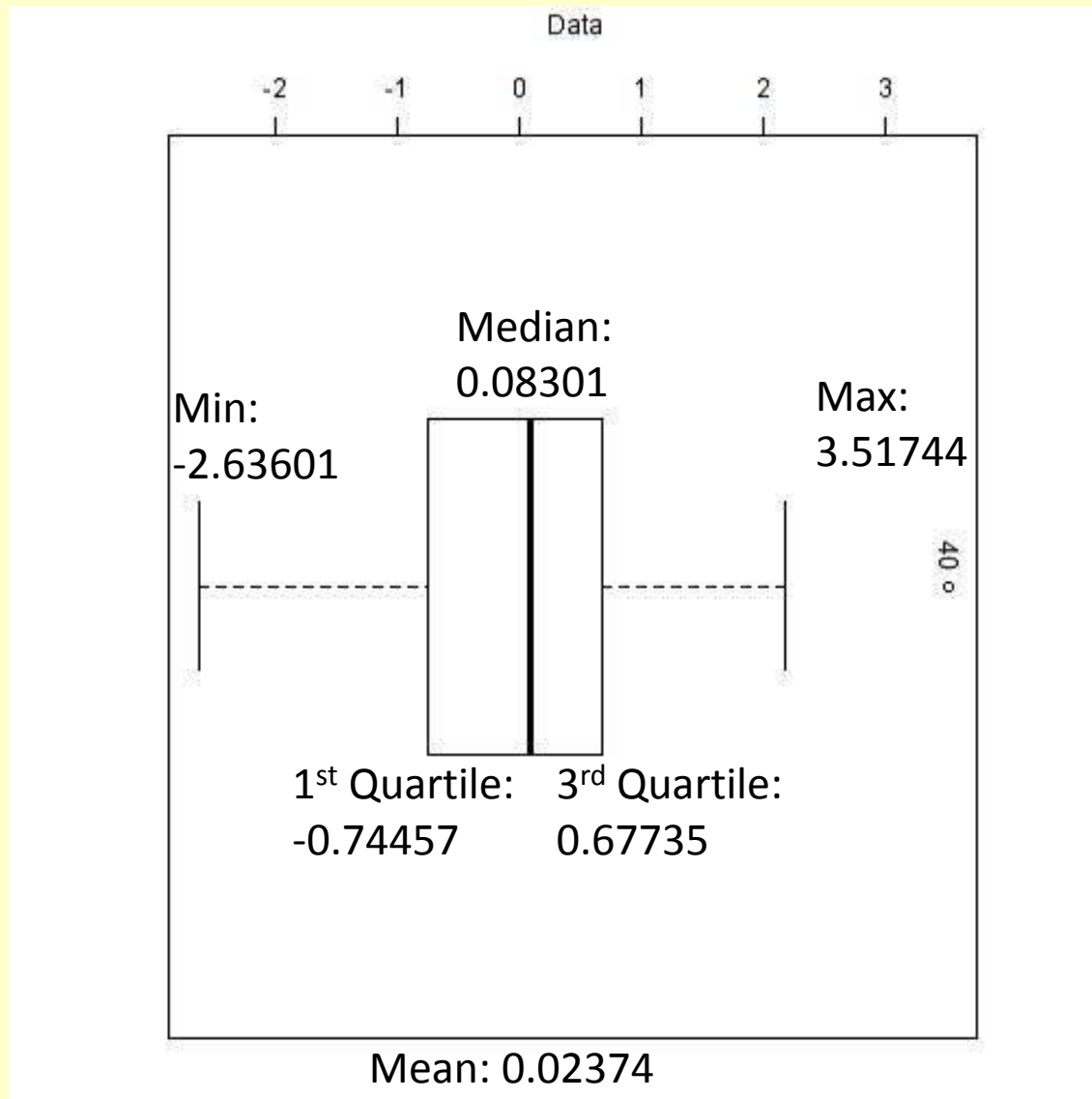
Answers

RData2

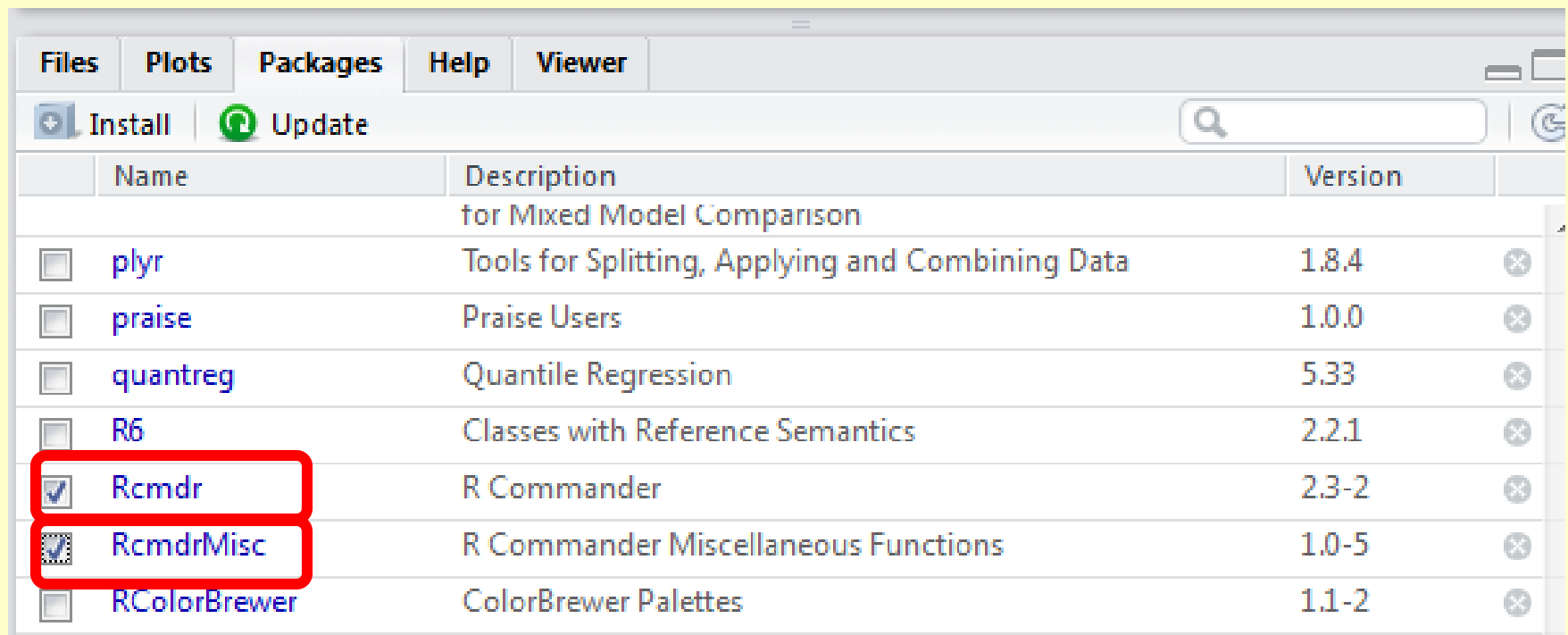


Answers

RData3



Installing "Rcmdr" Package in R Studio



The screenshot shows the R Studio interface with the 'Packages' pane open. The 'Install' button is highlighted. A table lists several packages, with 'Rcmdr' and 'RcmdrMisc' selected (checked boxes) and highlighted with a red rectangle. The table columns are Name, Description, and Version.

	Name	Description	Version	
		for Mixed Model Comparison		
<input type="checkbox"/>	plyr	Tools for Splitting, Applying and Combining Data	1.8.4	⊗
<input type="checkbox"/>	praise	Praise Users	1.0.0	⊗
<input type="checkbox"/>	quantreg	Quantile Regression	5.33	⊗
<input type="checkbox"/>	R6	Classes with Reference Semantics	2.2.1	⊗
<input checked="" type="checkbox"/>	Rcmdr	R Commander	2.3-2	⊗
<input checked="" type="checkbox"/>	RcmdrMisc	R Commander Miscellaneous Functions	1.0-5	⊗
<input type="checkbox"/>	RColorBrewer	ColorBrewer Palettes	1.1-2	⊗

```
> library("Rcmdr", lib.loc="~/R/win-library/3.4")
```

```
Loading required package: splines
```

```
Loading required package: RcmdrMisc
```

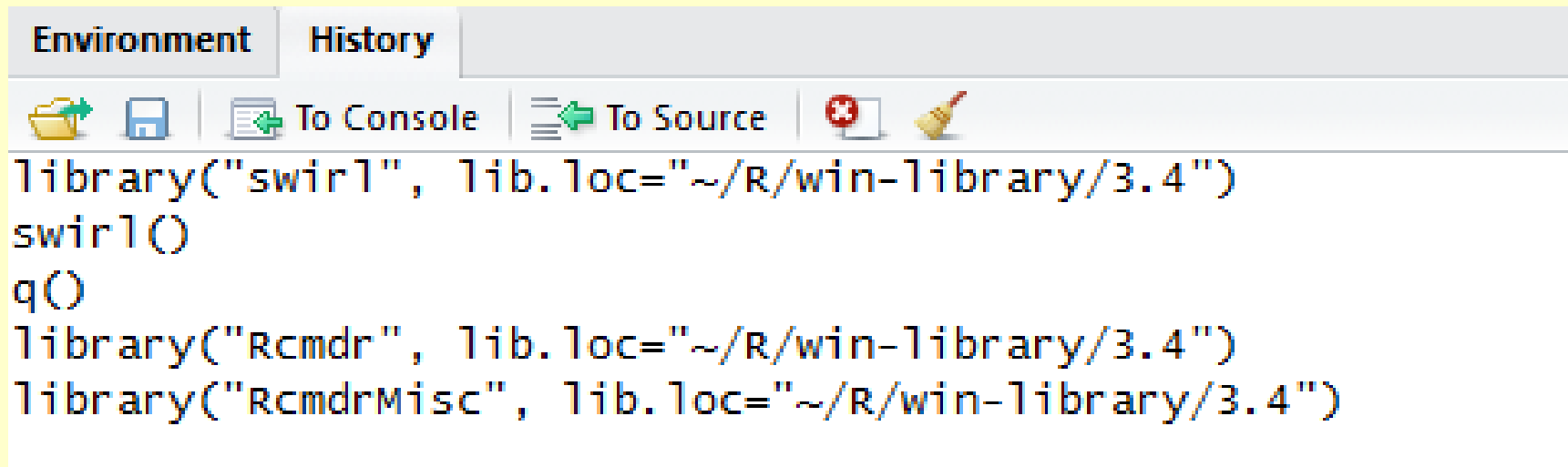
```
Loading required package: car
```

```
Loading required package: sandwich
```

```
RcmdrMsg: [1] NOTE: R Commander Version 2.3-2: Mon Sep 04 13:08:46 2017
```

```
Rcmdr version 2.3-2
```

Check the History Window



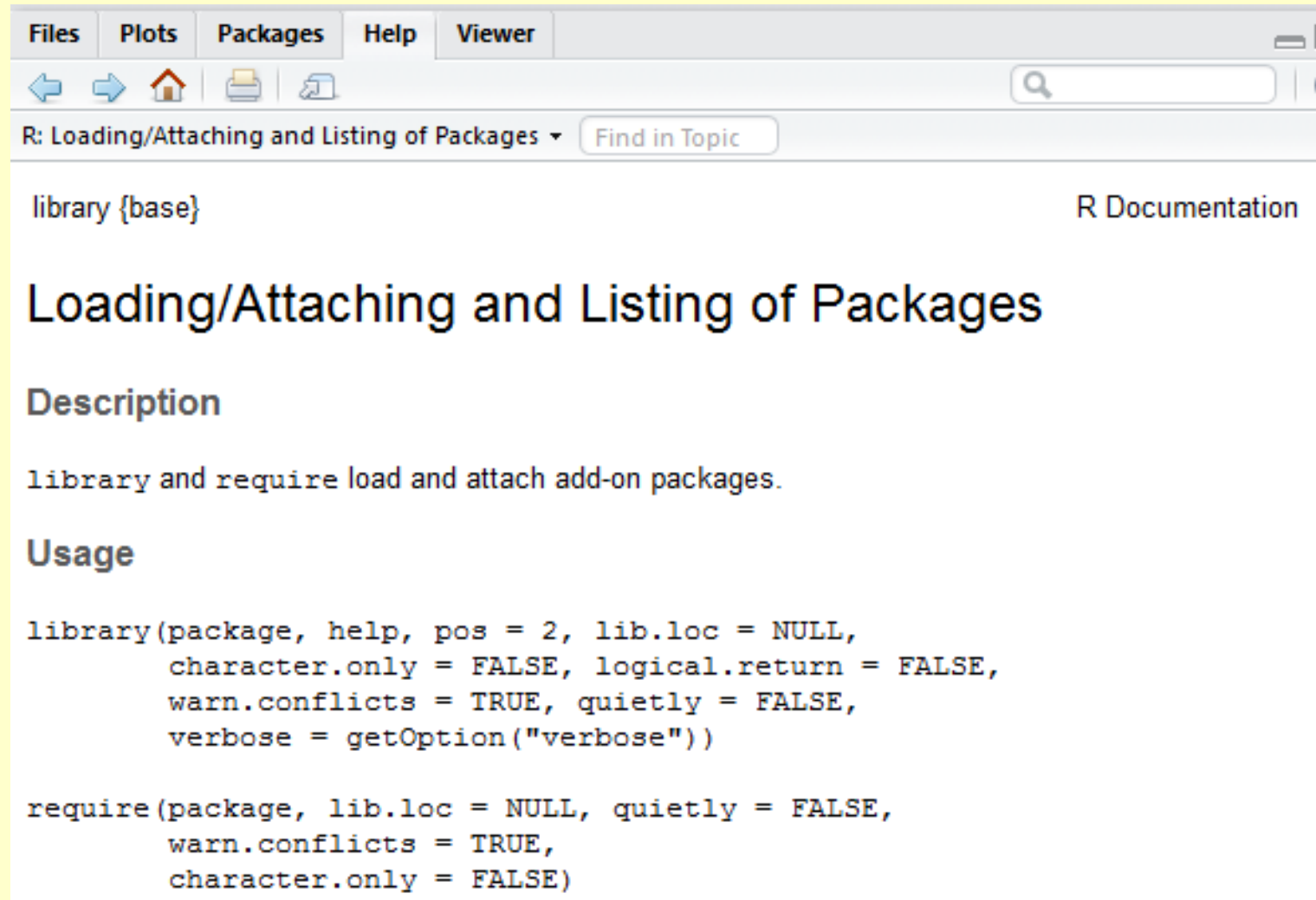
The screenshot shows the RStudio interface with the 'History' window active. The window title bar includes 'Environment' and 'History'. Below the title bar is a toolbar with icons for 'To Console', 'To Source', and a brush icon. The history list contains the following commands:

```
library("swirl", lib.loc="~/R/win-library/3.4")
swirl()
q()
library("Rcmdr", lib.loc="~/R/win-library/3.4")
library("RcmdrMisc", lib.loc="~/R/win-library/3.4")
```

> library("Rcmdr", lib.loc="~/R/win-library/3.4")

Check the Function Library

> help(library)



The screenshot shows the R help window for the `library` function. The window title is "R: Loading/Attaching and Listing of Packages". The main heading is "Loading/Attaching and Listing of Packages". The description states: "library and require load and attach add-on packages." The usage section shows the following code:

```
library(package, help, pos = 2, lib.loc = NULL,  
        character.only = FALSE, logical.return = FALSE,  
        warn.conflicts = TRUE, quietly = FALSE,  
        verbose = getOption("verbose"))  
  
require(package, lib.loc = NULL, quietly = FALSE,  
        warn.conflicts = TRUE,  
        character.only = FALSE)
```

<http://www.rcommander.com/>

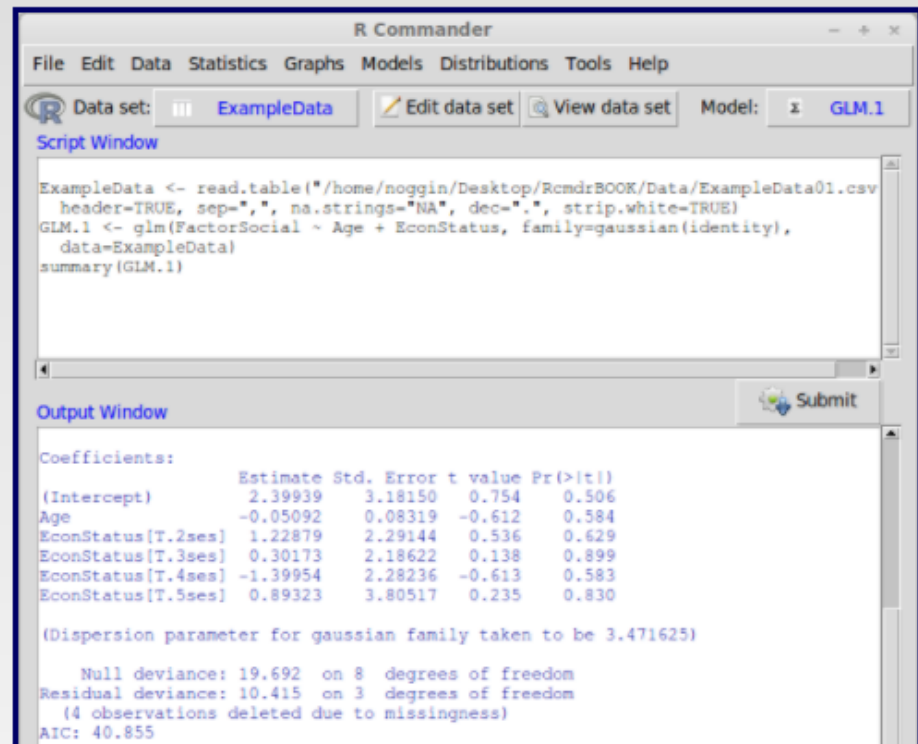
<http://socserv.mcmaster.ca/jfox/Misc/Rcmdr/>

Rcommander a graphical interface for R

R commander (Rcmdr)

R provides a powerful and comprehensive system for analysing data and when used in conjunction with the R-commander (a graphical user interface, commonly known as Rcmdr) it also provides one that is easy and intuitive to use. Basically, R provides the engine that carries out the analyses and Rcmdr provides a convenient way for users to input commands. The Rcmdr program enables analysts to access a selection of commonly-used R commands using a simple interface that should be familiar to most computer users. It also serves the important role of helping users to implement R commands and develop their knowledge and expertise in using the command line --- an important skill for those wishing to exploit the full power of the program.

Information about installing R can be found on the web at the R homepage <http://www.r-project.org/> which provides lots of information about the R project and also directs users to one of the CRAN sites (the Comprehensive R Archive Network) that have been set up on many servers across the world in order for users to download the software. CRAN provides all files necessary to install R on a number of different computing platforms (Linux, MacOS X and Windows) along with detailed information about installation and also offers manuals and contributed documentation in a number of languages and for a number of specific disciplines.



The screenshot shows the R Commander application window. The title bar reads "R Commander". The menu bar includes "File", "Edit", "Data", "Statistics", "Graphs", "Models", "Distributions", "Tools", and "Help". The "Data set:" field shows "ExampleData" with a dropdown arrow. The "Model:" field shows "GLM.1" with a dropdown arrow. The "Script Window" contains the following R code:

```
ExampleData <- read.table("/home/noggin/Desktop/RcmdrBOOK/Data/ExampleData01.csv",
  header=TRUE, sep=";", na.strings="NA", dec=".", strip.white=TRUE)
GLM.1 <- glm(FactorSocial ~ Age + EconStatus, family=gaussian(identity),
  data=ExampleData)
summary(GLM.1)
```

The "Output Window" displays the results of the GLM fit:

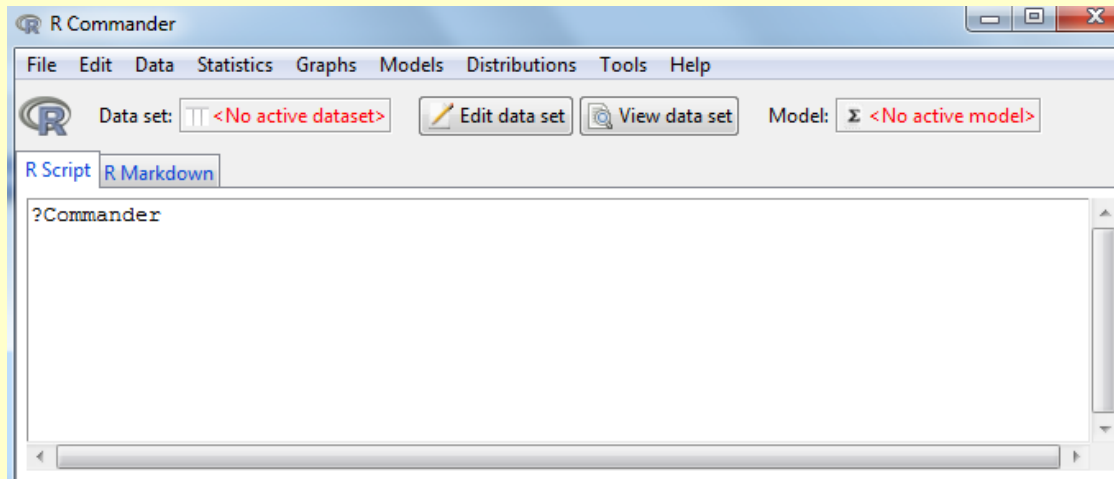
```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.39939    3.18150   0.754   0.506
Age          -0.05092    0.08319  -0.612   0.584
EconStatus[T.2ses] 1.22879    2.29144   0.536   0.629
EconStatus[T.3ses] 0.30173    2.18622   0.138   0.899
EconStatus[T.4ses] -1.39954    2.28236  -0.613   0.583
EconStatus[T.5ses] 0.89323    3.80517   0.235   0.830

(Dispersion parameter for gaussian family taken to be 3.471625)

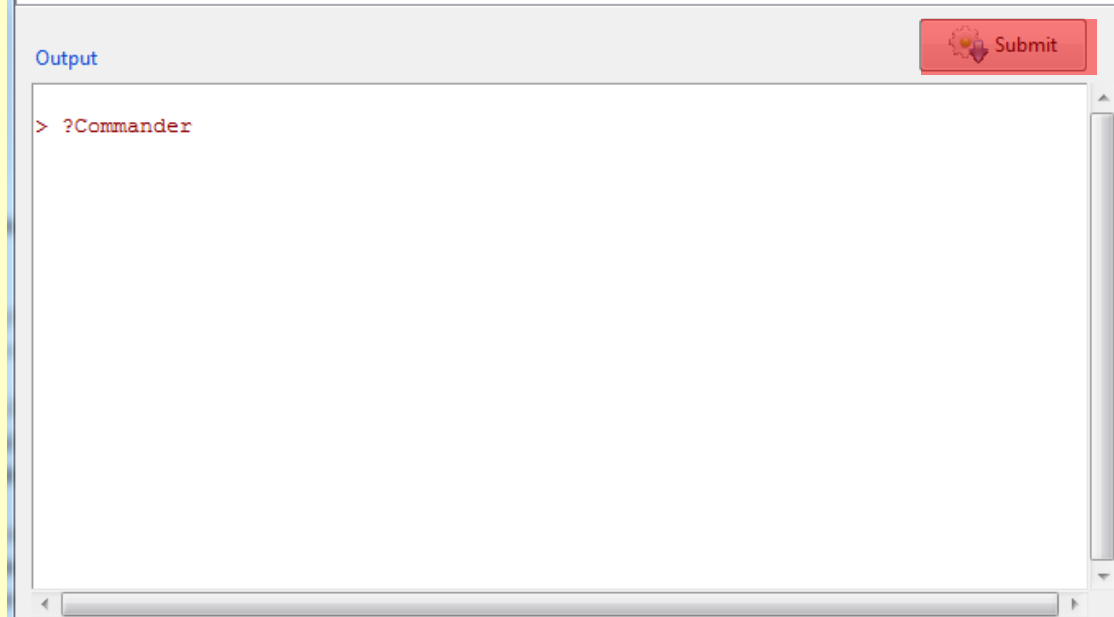
Null deviance: 19.692 on 8 degrees of freedom
Residual deviance: 10.415 on 3 degrees of freedom
(4 observations deleted due to missingness)
AIC: 40.855
```

Running Rcmdr Package

Console Window

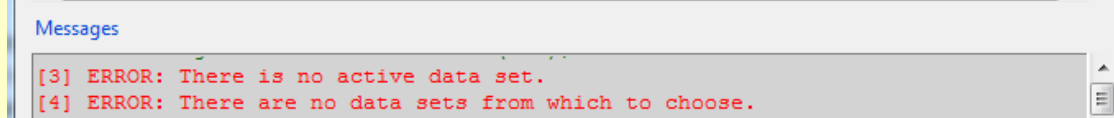


Output Window

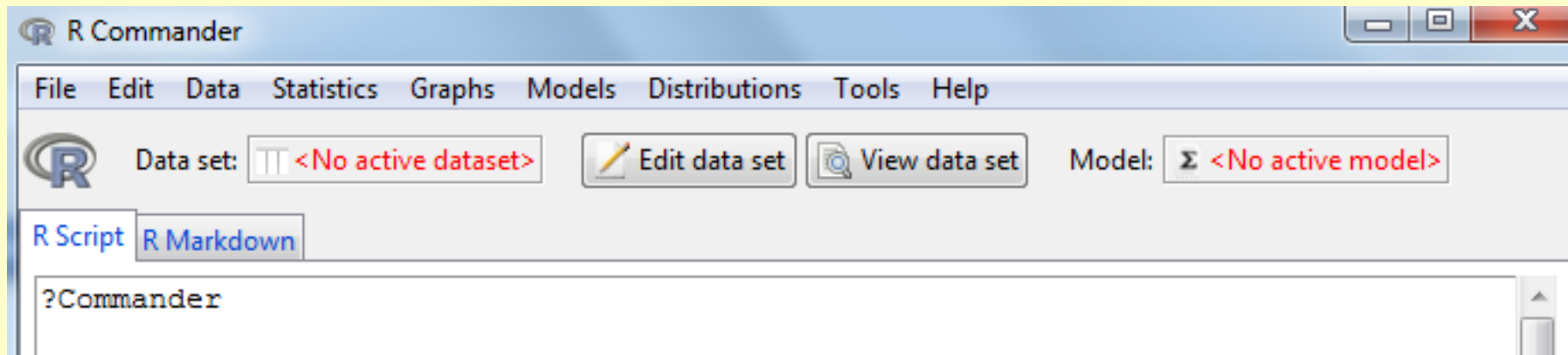


Submit Button ("ENTER")

Messages



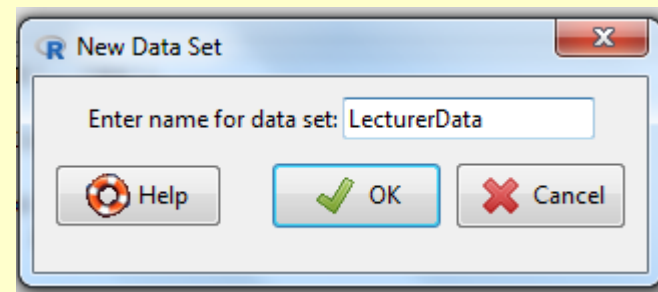
Creating a Dataframe in Rcmdr



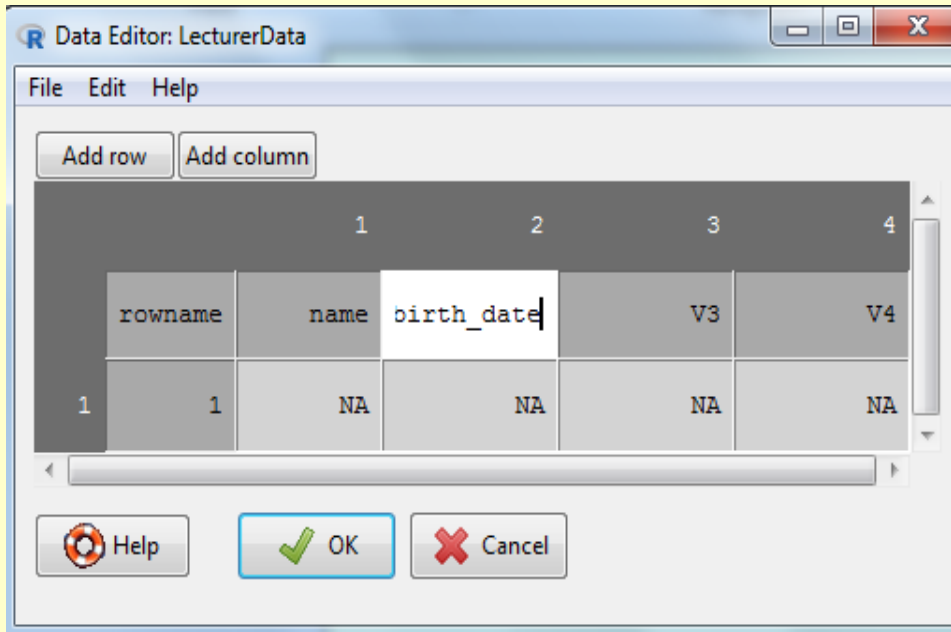
Rcmdr has a built-in spreadsheet tool that allows the user to enter and manipulate data.

For this exercise, you will create a dataframe called "LecturerData" similar to the one developed in lecture 3.

Go to the top Rcmdr menu and use:
Data / New Data Set



Adding Data to a Dataframe in Rcmdr



Create a dataset with four columns and four rows by adding rows / columns to dataframe

Name columns

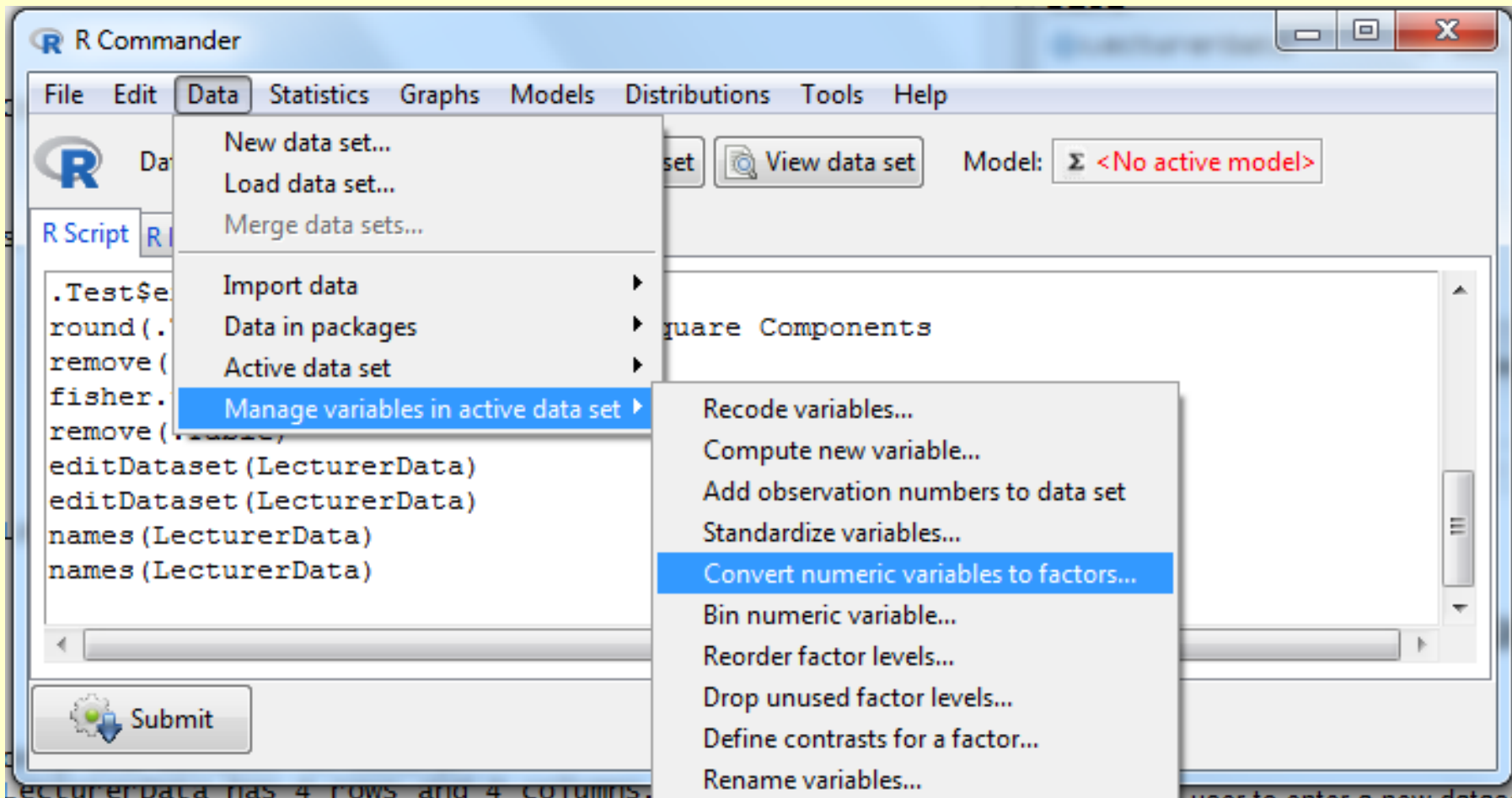
Enter data shown below:

REMEMBER THESE RULES:

- Data from the same entity go in the same row
- Different variables are placed on different columns

	name	age	salary	job
1	bob	25	10	1
2	mary	26	11	1
3	phil	27	12	2
4	john	28	10	2

Creating Coding Variables in Rcmdr



Data /

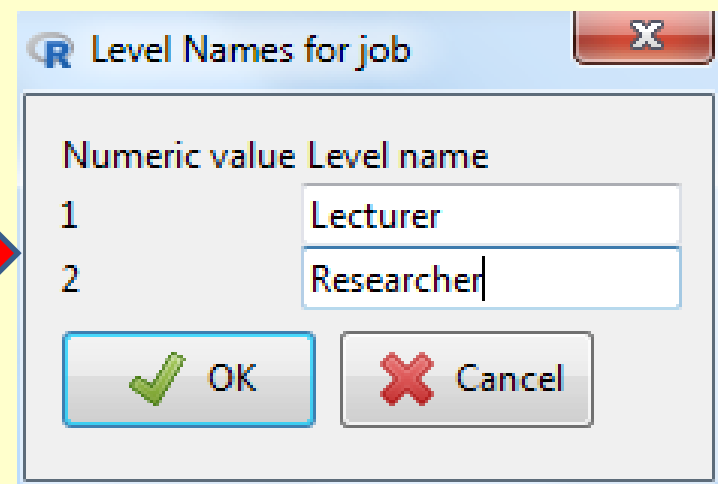
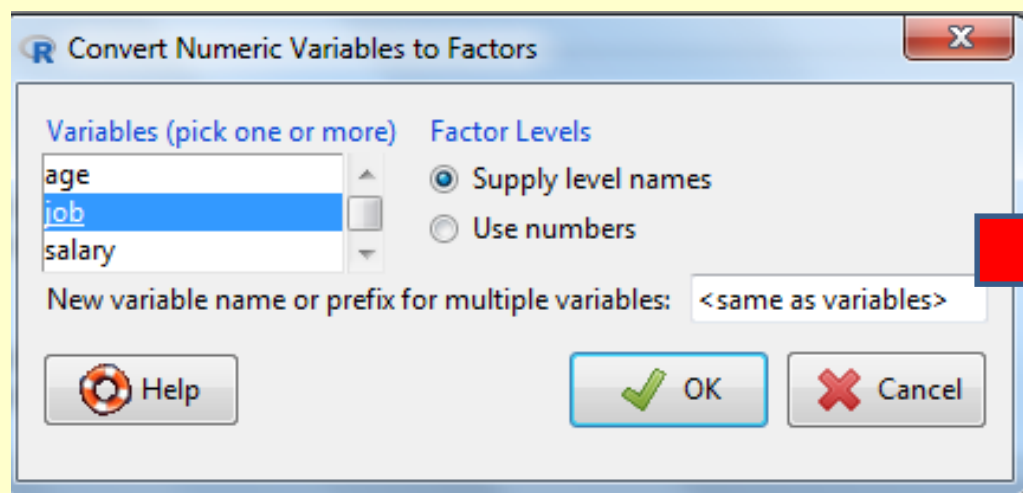
Manage Variables in Active Dataset /

Convert Numeric Variables to Factors

Creating Coding Variables in Rcmdr

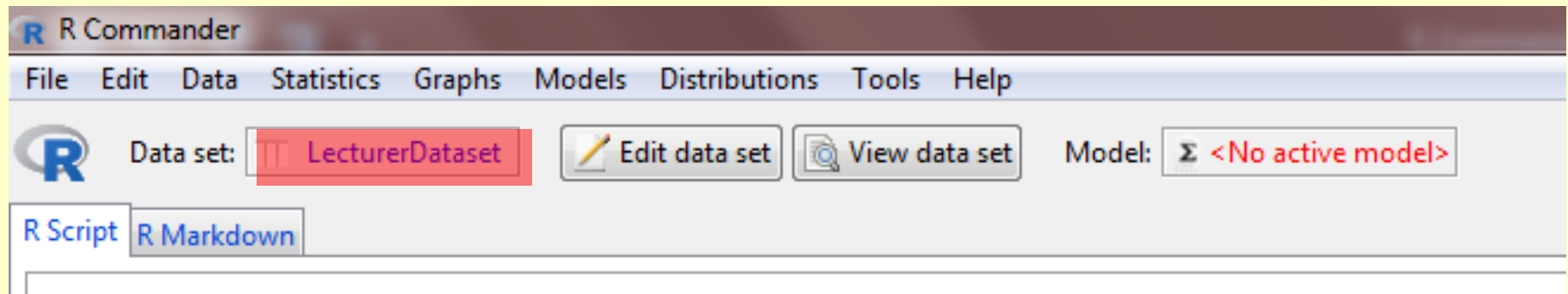
Note: Because we want to analyze the categorical data (1 is lecturer / 2 is researcher) as a factor in an analysis, we coded these fields with integers.

However, we want the computer to understand that these two integers are used as identifiers of categorical data: 1 means lecturer & 2 means researcher



Overwrite the "Jobs" Variable

Editing a Dataset in Rcmdr



Selecting and Editing Datasets:

Activate dataset by selecting it from scrolling list

Use menu buttons to view and edit the selected dataset

The screenshot shows a window titled 'LecturerDataset' with a table of data. The table has four columns: 'name', 'age', 'salary', and 'job'. The rows are numbered 1 to 4.

	name	age	salary	job
1	bob	25	10	lecturer
2	mary	26	11	lecturer
3	phil	27	12	researcher
4	john	28	10	researcher

The screenshot shows a window titled 'Data Editor: LecturerDataset' with a table of data. The table has five columns: 'rowname', 'name', 'age', 'salary', and 'job'. The rows are numbered 1 to 4. Below the table are three buttons: 'Help', 'OK', and 'Cancel'.

	rowname	name	age	salary	job
1	1	bob	25	10	lecturer
2	2	mary	26	11	lecturer
3	3	phil	27	12	researcher
4	4	john	28	10	researcher

Subsetting Data with Rcmdr

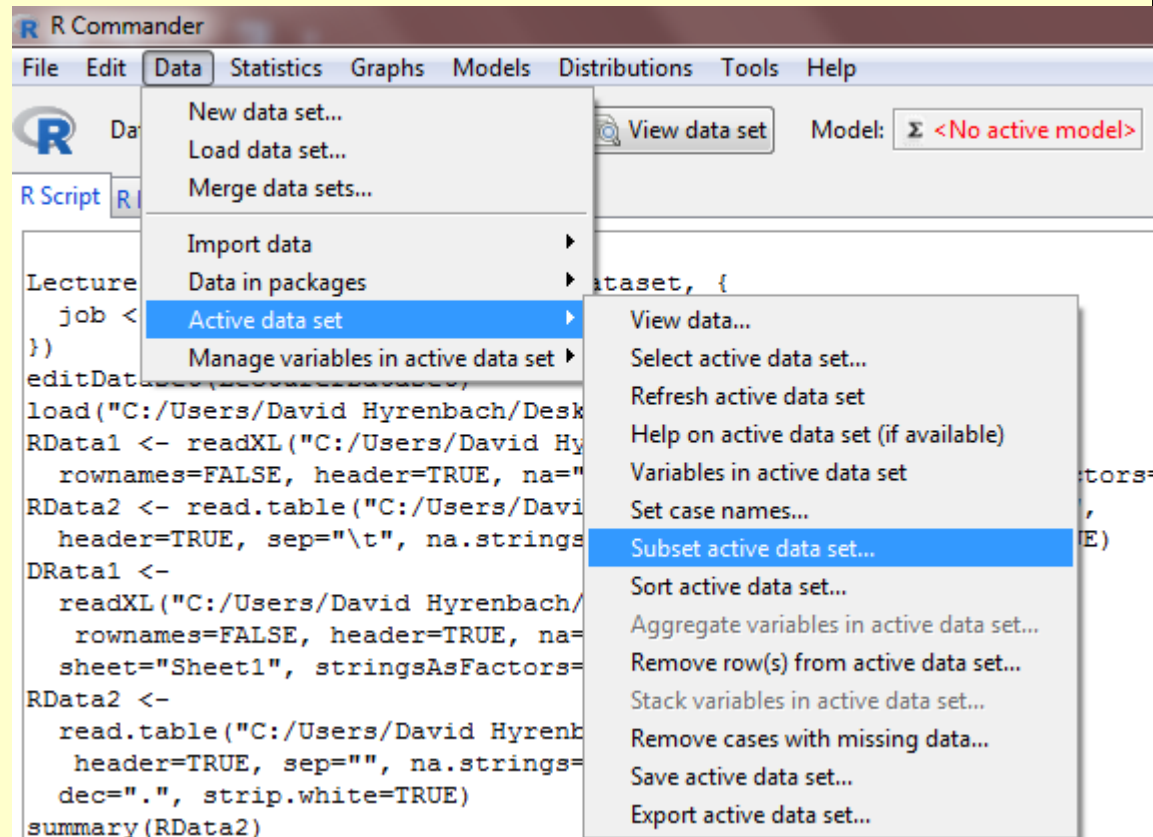
Make `LecturerData` the active file

Extract records where "job" is Lecturer, and set aside records where "job" is Researcher.

Use `subset` command:

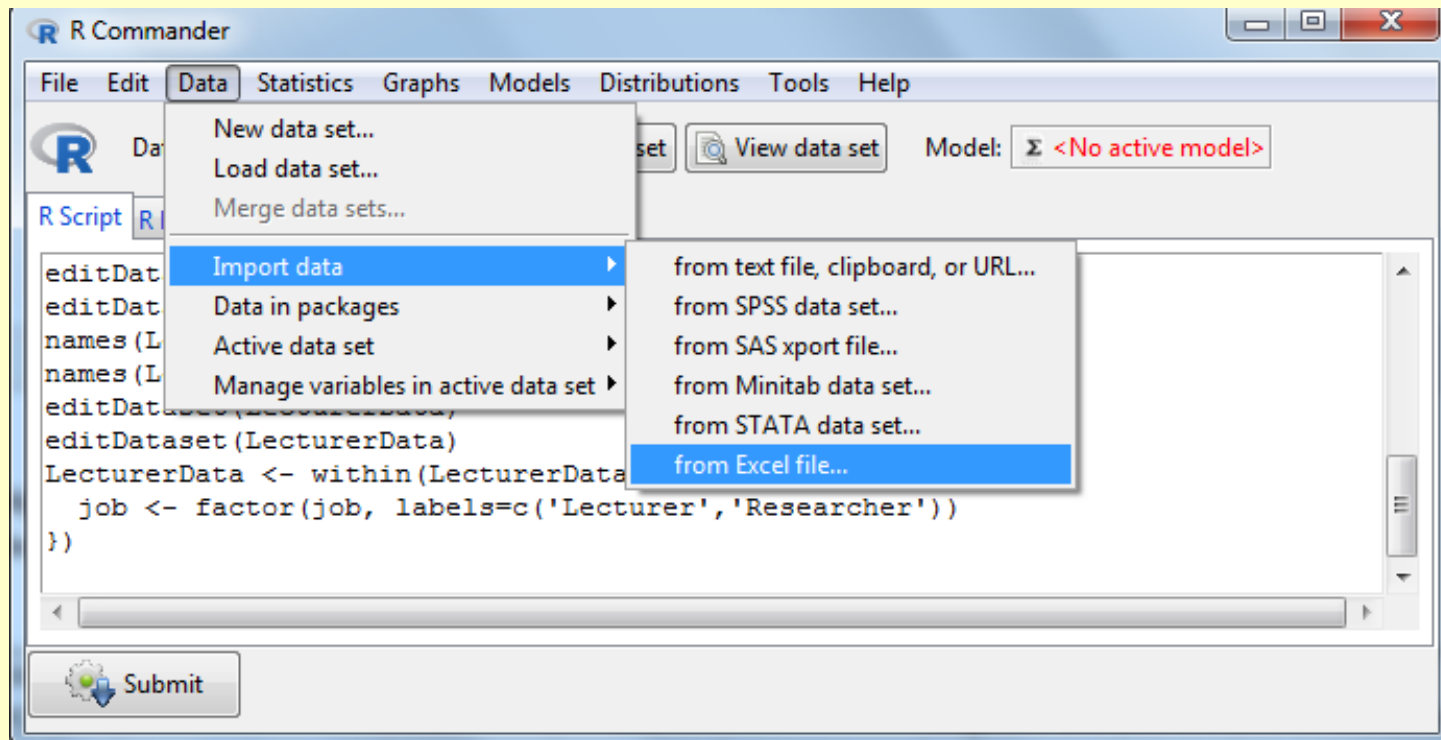
`Data / Active Dataset / Subset Active Dataset`

Save the new dataframe you created. Call it: `LecturerOnly`



Importing Data into Rcmdr

Note: Excel files ("xlsx", "csv" and ".txt") and SPSS files ("dat") can be easily imported into Rcmdr



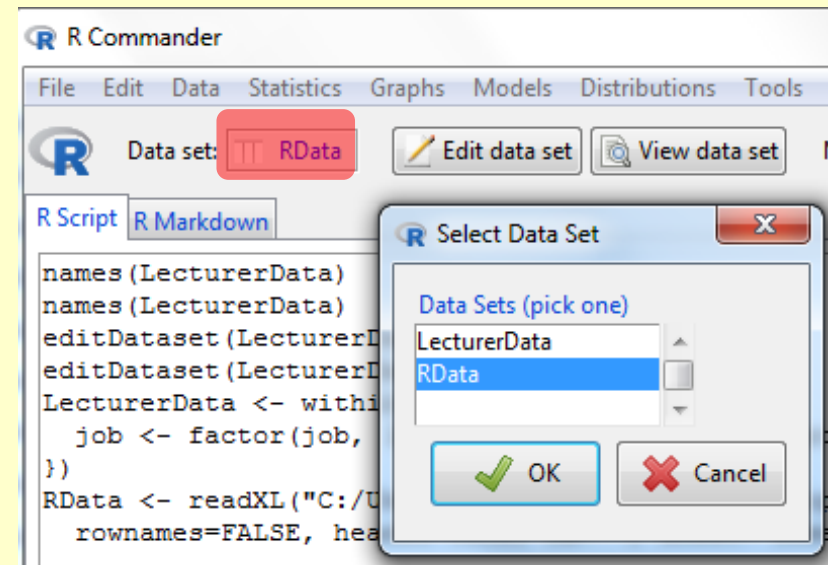
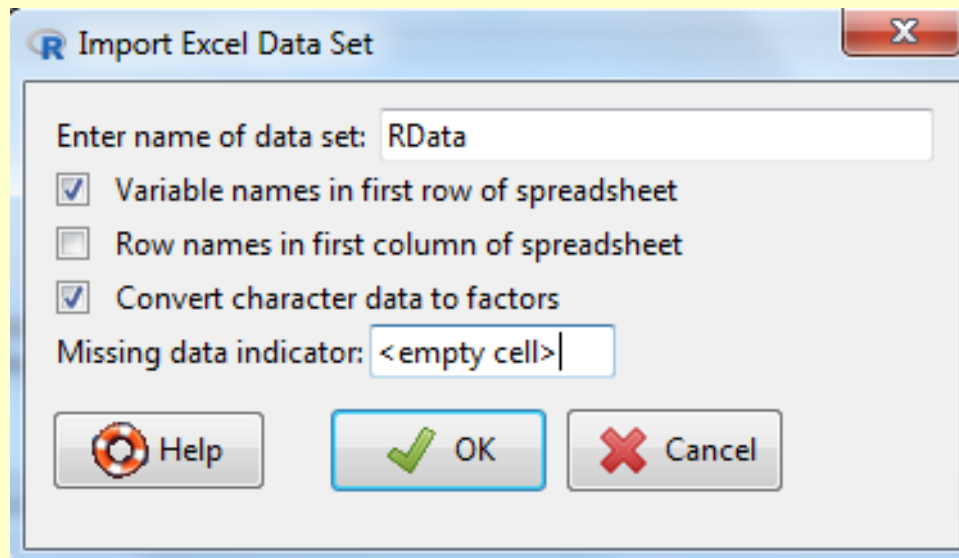
Data / Import Data / from Excel file...

Importing Data into Rcmdr

Import file **RData1.xlsx**

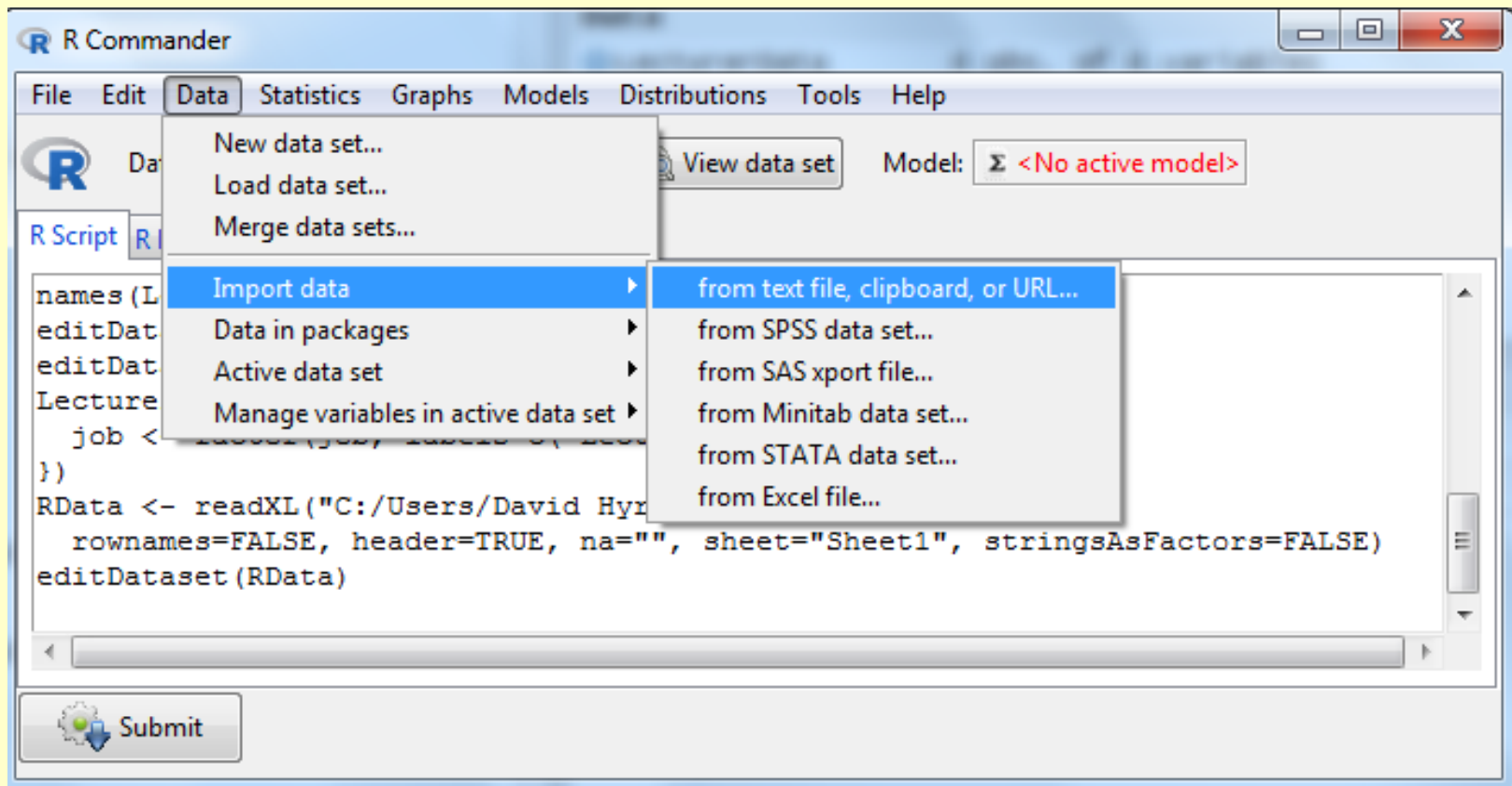
Enter the following information:

dataset name,
whether headers are present,
whether you want to convert character data into factors,
and whether there are any missing data (empty cells)



Importing Data into Rcmdr

Note: Importing ".txt" files from a variety of sources.



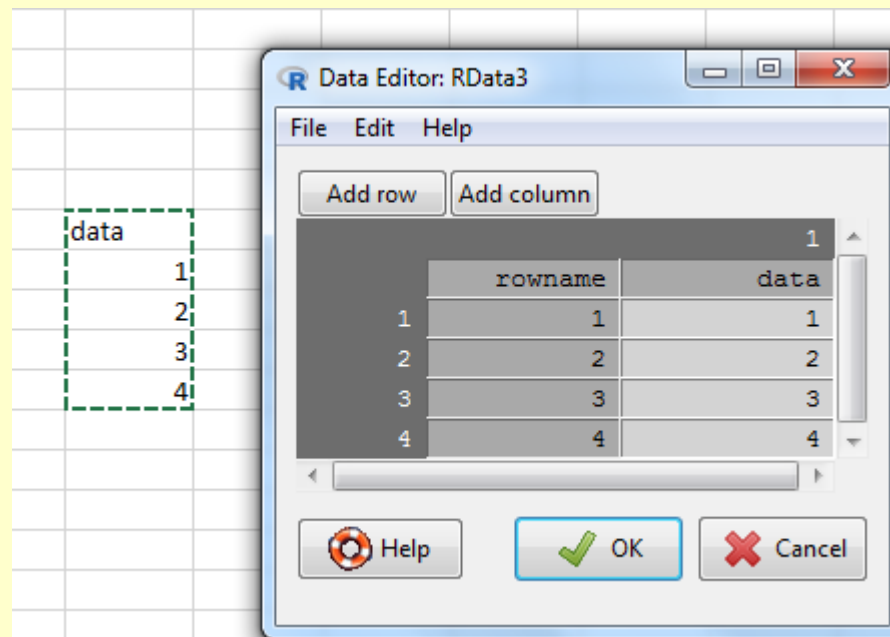
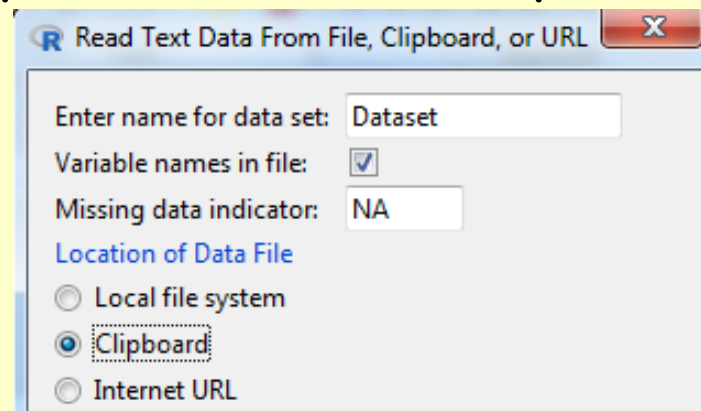
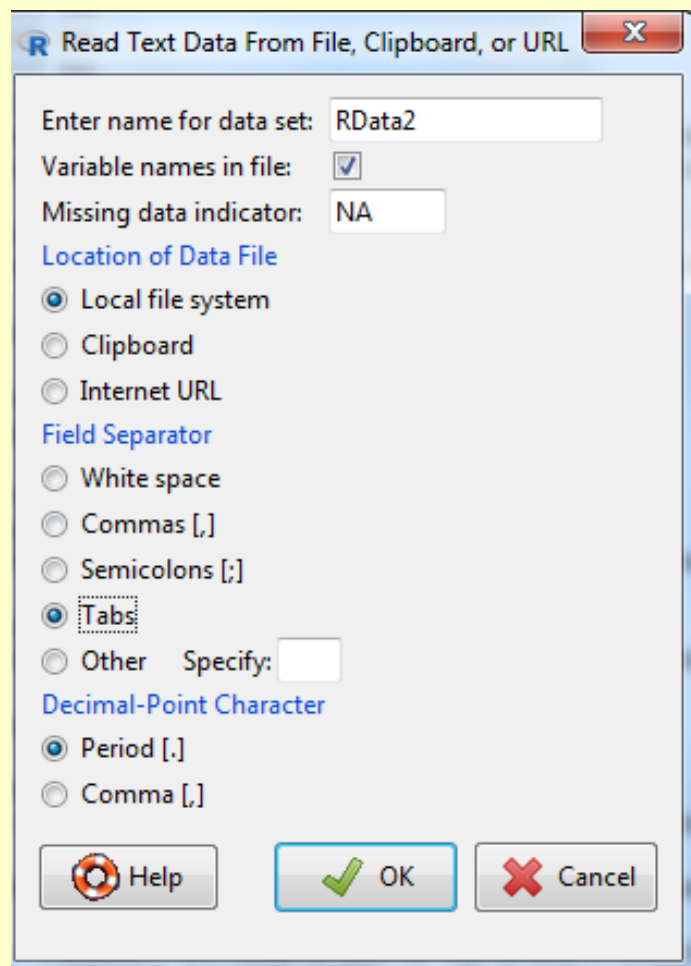
Data / Import Data/ from text file, clipboard, or URL...

Importing Data into Rcmdr

Import file **RData2.txt**

Import data from clipboard

Enter metadata:



Importing Data into Rcmdr

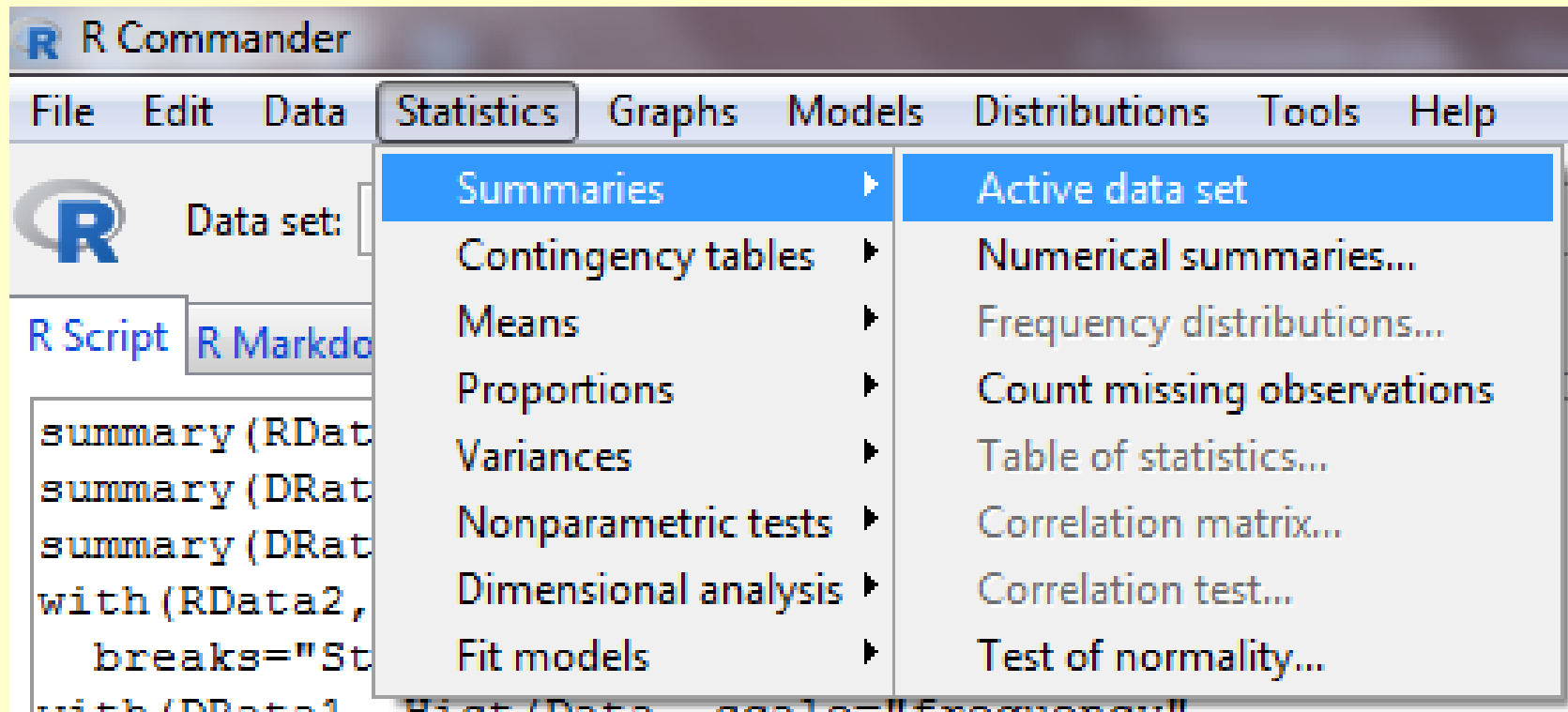
Import file `RData3.csv`

Figure out how to import this csv file

Hint: you will need to enter a line command

Summarizing Data with Rcmdr

- Summarize datasets using:
Statistics / Summaries / Active data set



NOTE: you need to select the active dataset and perform these calculations one at a time

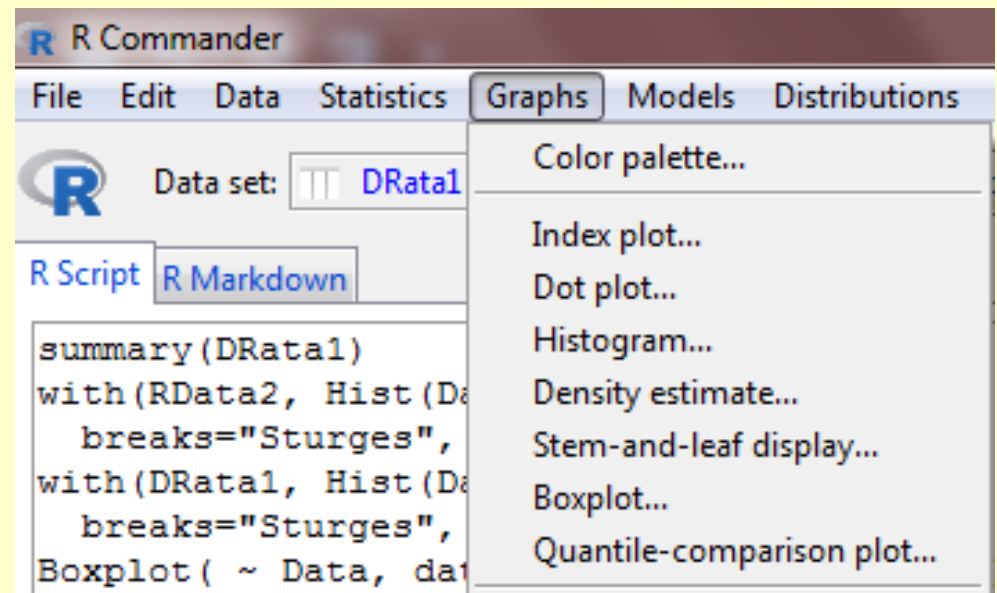
Sumamrizing Data with Rcmdr

- Create plots using:

Graphs /

Histograms

Box Plots



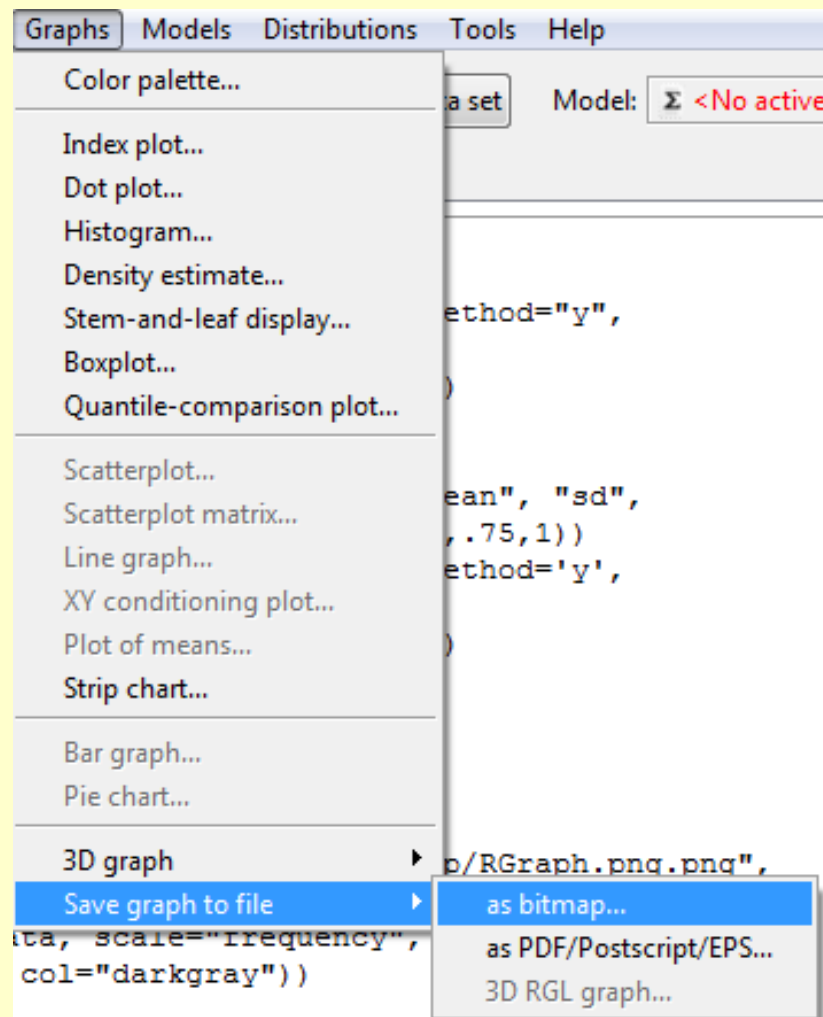
NOTE: you need to select the active dataset and perform these calculations one at a time

HINT: What types of plots can RCmdr make?
(Explore the other available plots)

Saving Plots Made with Rcmdr

- After you create a plot, can save it using:

Graphs /
Save graph to file /
as bitmap
as PDF / PS / EPS



Saving Plots Made with Rcmdr

NOTE: Bitmap is JPEG or PNG format files

You can select:

- Picture size
- Font size

