INTRODUCTION TO R

Konstantinos Kounetas

School of Business Administration
Department of Economics

Master of Science in Applied Economic Analysis

SPSS =

muggle

SPSS and SAS users are like muggles. They are limited in their ability to change their environment. They have to rely on algorithms that have been developed for them. The way they approach a problem is constrained by how SAS/SPSS employed programmers thought to approach them. And they have to pay money to use these constraining algorithms.

=

wizard

R users are like wizards. They can rely on functions (spells) that have been developed for them by statistical researchers, but they can also create their own. They don't have to pay for the use of them, and once experienced enough (like Dumbledore), they are almost unlimited in their ability to change their environment.

Some history

S was developed at Bell Labs, starting in the 1970s

R was created in the 1990s by Ross Ihaka and Robert Gentleman
R was based on S, with code written in C
S largely was used to make good graphs – not an easy thing in 1975. R, like S, is quite good for graphing. For lots of examples, ṡee http://rgraphgallery.blogspot.com/
or http://www.r-graph-gallery.com

See ggplot2-cheatsheet-2.0.pdf

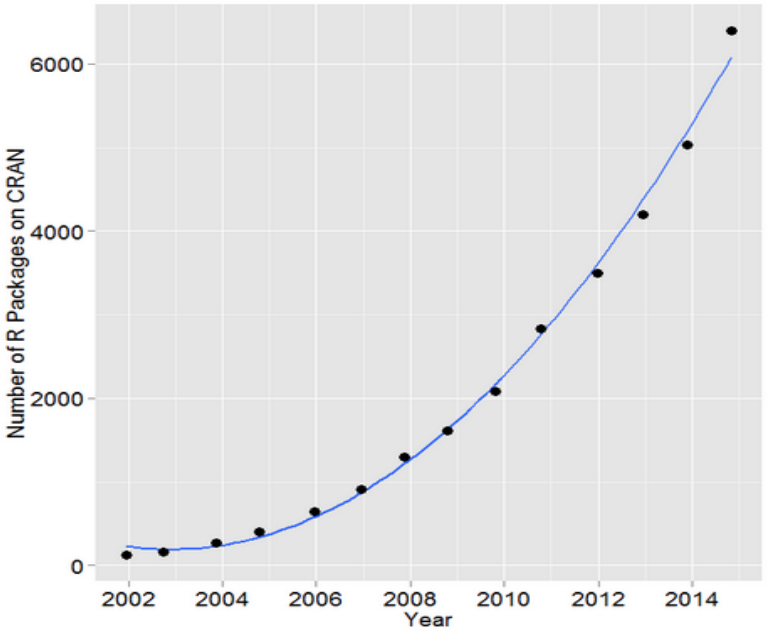


Figure 9. Number of R packages available on its main distribution site for the last version released in each year.

Outline

- Introduction:
  - Historical development
  - S, Splus
  - Capability
  - Statistical Analysis
- References
- Calculator
- Data Type
- Resources
- Simulation and Statistical Tables
  - Probability distributions
- Programming
  - Grouping, loops and conditional execution
  - Function
- Reading and writing data from files
- Modeling
  - Regression
  - ANOVA
- Data Analysis on Association
  - Lottery
  - Geyser
- Smoothing

R, S and S-plus

- S: an interactive environment for data analysis developed at Bell Laboratories since 1976
  - 1988 - S2: RA Becker, JM Chambers, A Wilks
  - 1992 - S3: JM Chambers, TJ Hastie
  - 1998 - S4: JM Chambers
- Exclusively licensed by *AT&T/Lucent* to *Insightful Corporation*, Seattle WA. Product name: “S-plus”.
- Implementation languages C, Fortran.
- See: http://cm.bell-labs.com/cm/ms/departments/sia/S/history.html
- 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 ca. 15 people with access to common CVS archive.

Introduction

- R is “GNU S” — A language and environment for data manipulation, calculation and graphical display.
  - R is similar to the award-winning S system, which was developed at Bell Laboratories by John Chambers et al.
  - a suite of operators for calculations on arrays, in particular matrices,
  - a large, coherent, integrated collection of intermediate tools for interactive data analysis,
  - graphical facilities for data analysis and display either directly at the computer or on hardcopy
  - a well developed programming language which includes conditionals, loops, user defined recursive functions and input and output facilities.
- The core of R is an interpreted computer language.
  - It allows branching and looping as well as modular programming using functions.
  - Most of the user-visible functions in R are written in R, calling upon a smaller set of internal primitives.
  - It is possible for the user to interface to procedures written in C, C++ or FORTRAN languages for efficiency, and also to write additional primitives.

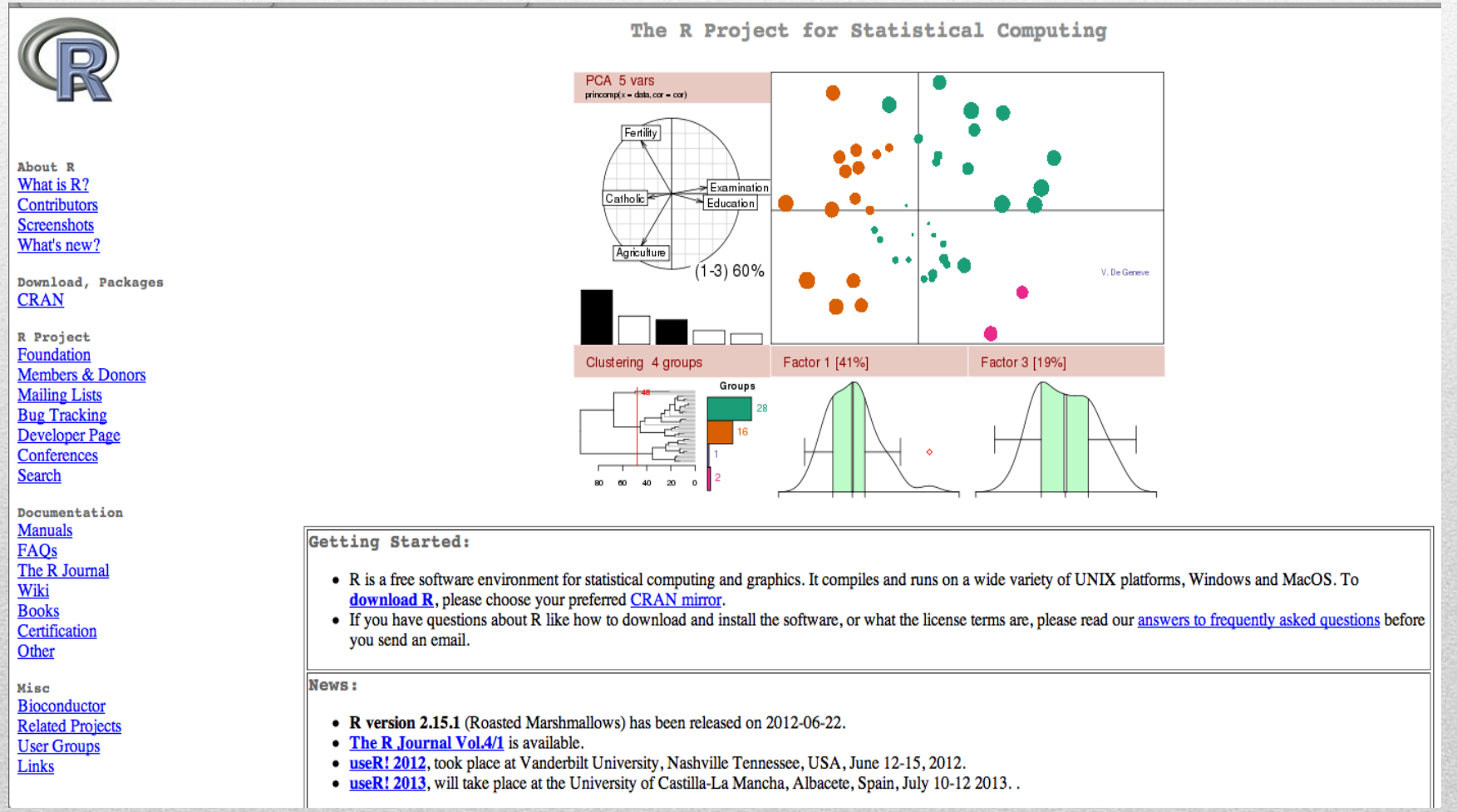
What R does and does not

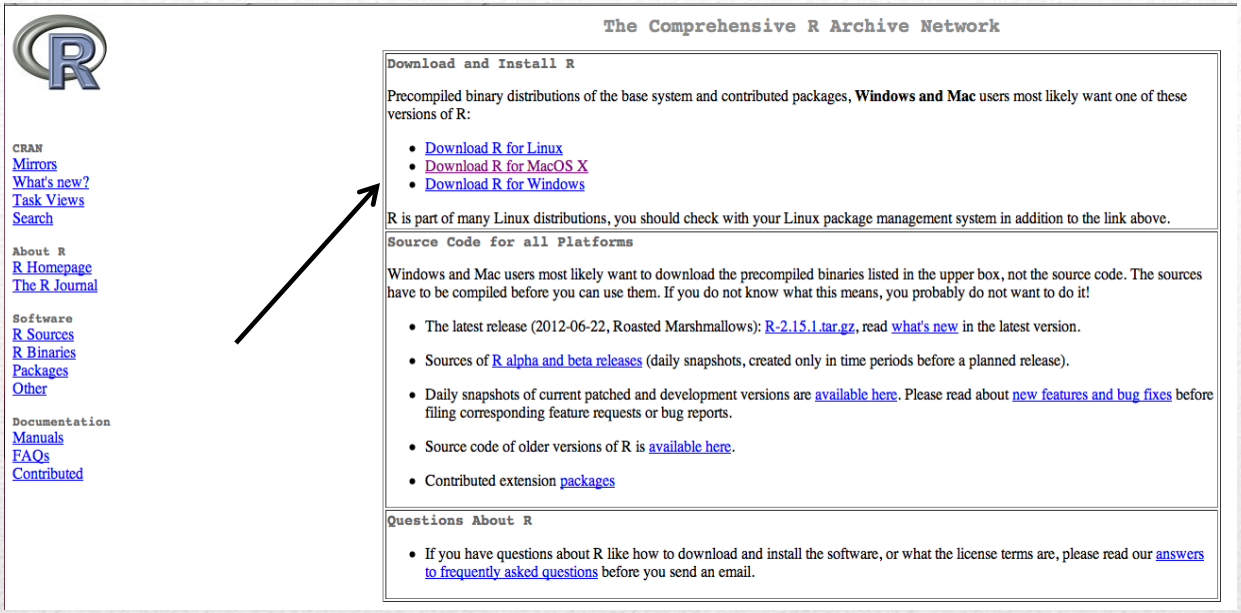
- data handling and storage: numeric, textual
- matrix algebra
- hash tables and regular expressions
- high-level data analytic and statistical functions
- classes (“OO”)
- graphics
- programming language: loops, branching, subroutines

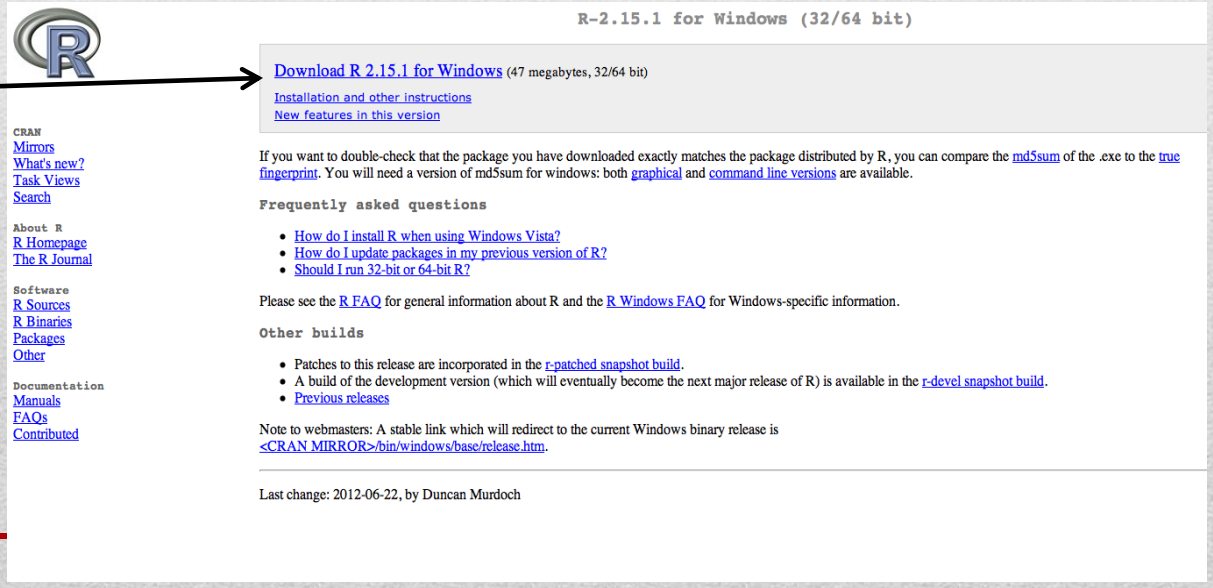
- is not a database, but connects to DBMSs
- has no graphical user interfaces, but connects to Java, TclTk
- language interpreter can be very slow, but allows to call own C/C++ code
- no spreadsheet view of data, but connects to Excel/MsOffice
- no professional / commercial support

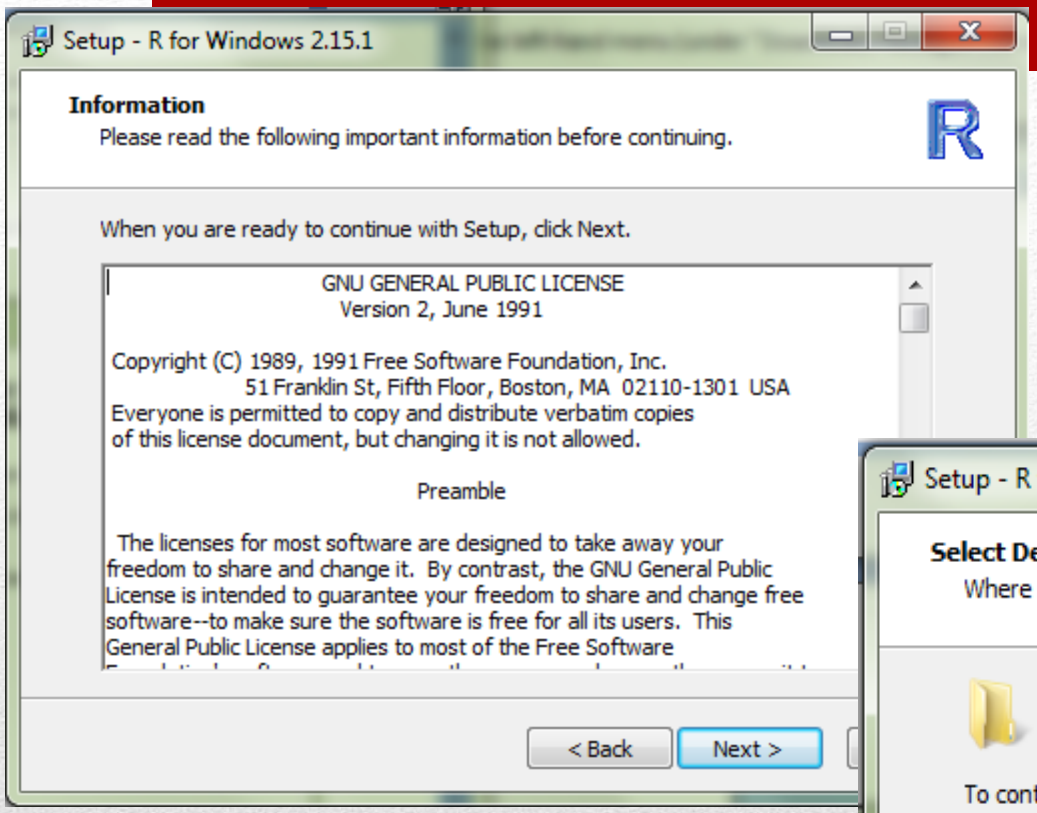
Getting Started-Installing R

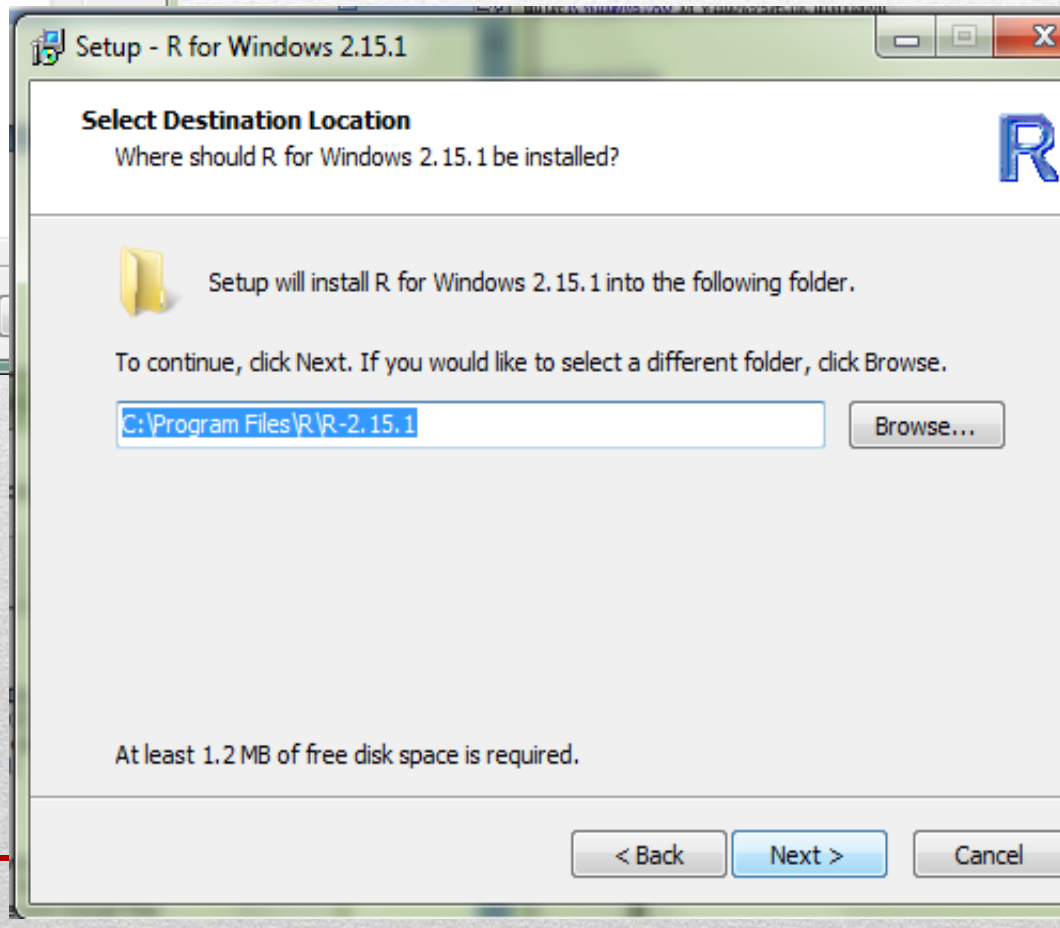
To install R on your MAC or PC you first need to go to http://www.r-project.org/.

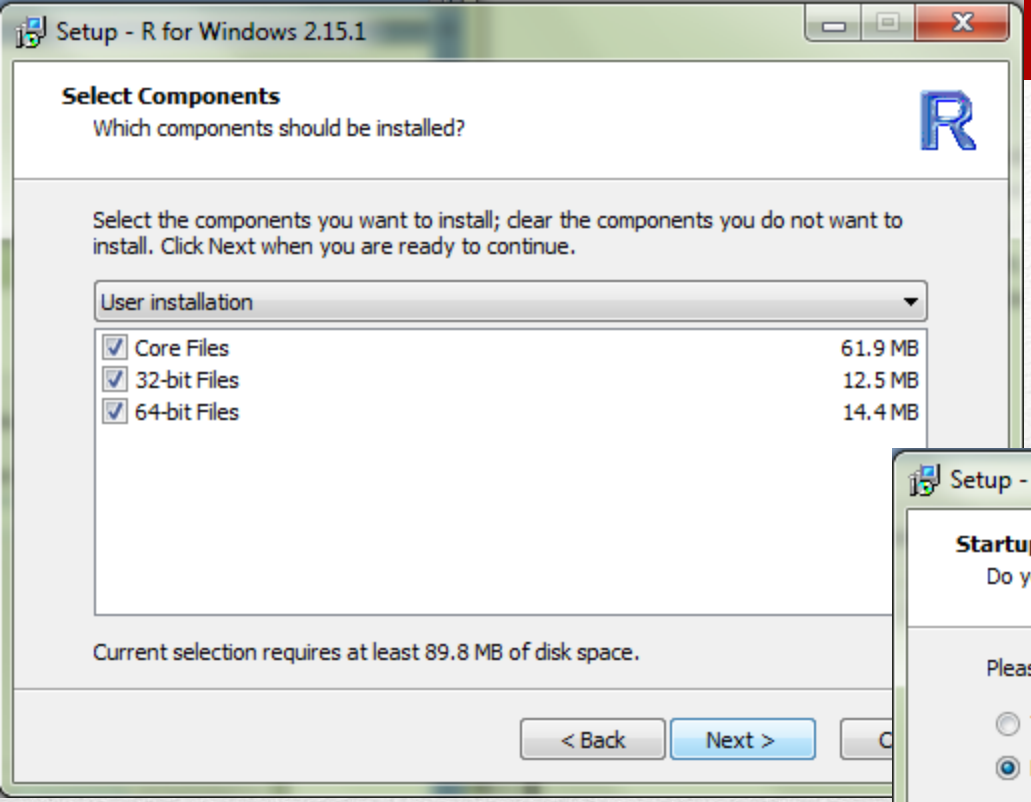


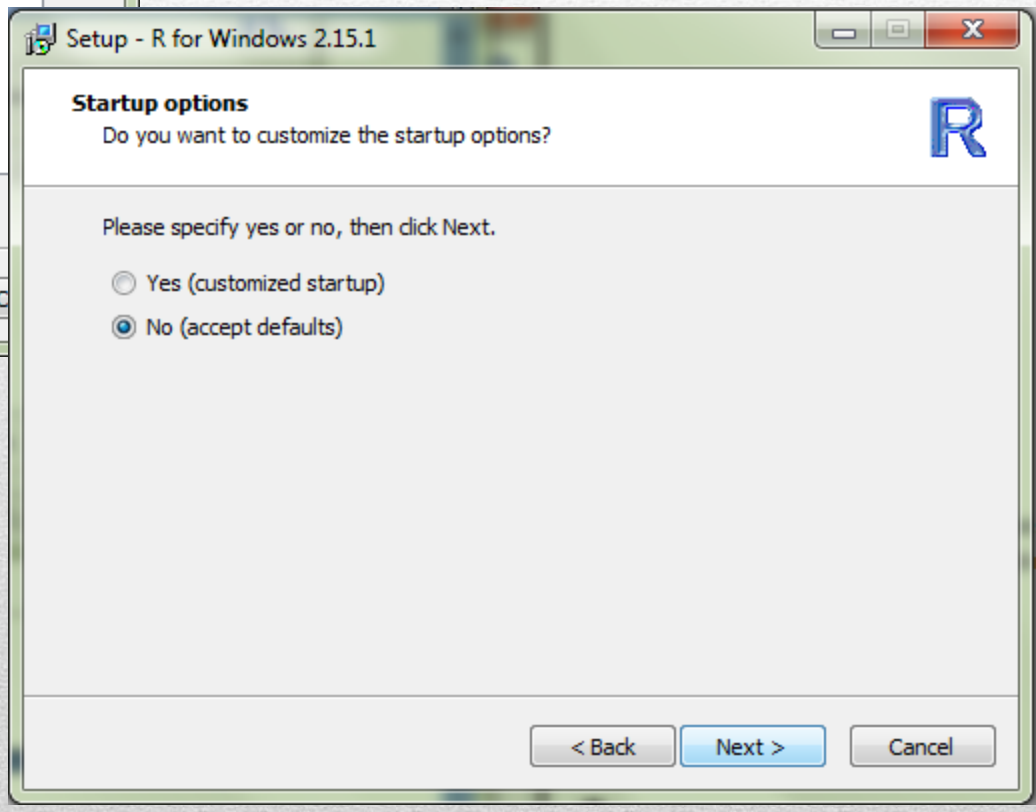

The Comprehensive R Archive Network
CRAN
Mirrors
What's new?
Task Views
Search
About R
R Homepage
The R Journal
Software
R Sources
R Binaries
Packages
Other
Documentation
Manuals
FAQs
Contributed
Download and Install R
Precompiled binary distributions of the base system and contributed packages, Windows and Mac users most likely want one of these versions of R:
Download R for Linux
Download R for MacOS X
Download R for Windows
R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.
Source Code for all Platforms
Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!
The latest release (2012-06-22, Roasted Marshmallows): R-2.15.1.tar.gz, read what's new in the latest version.
Sources of R alpha and beta releases (daily snapshots, created only in time periods before a planned release).
Daily snapshots of current patched and development versions are available here. Please read about new features and bug fixes before filing corresponding feature requests or bug reports.
Source code of older versions of R is available here.
Contributed extension packages
Questions About R
If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.

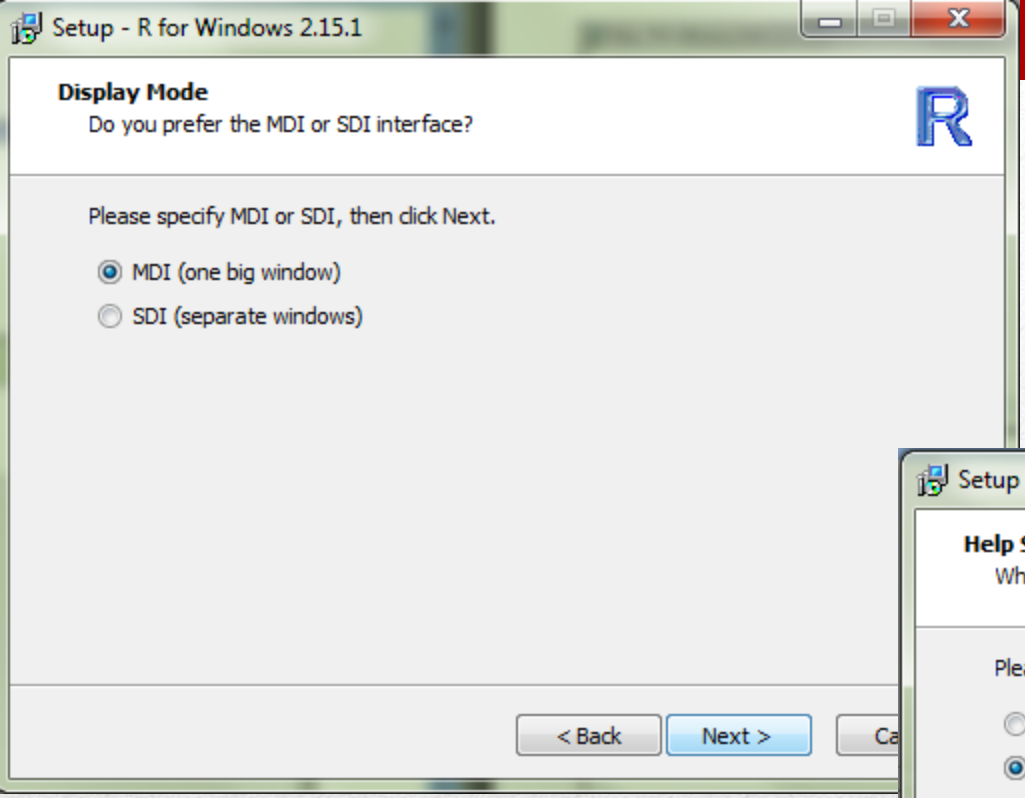

R-2.15.1 for Windows (32/64 bit)
Download R 2.15.1 for Windows (47 megabytes, 32/64 bit)
Installation and other instructions
New features in this version
CRAN
Mirrors
What's new?
Task Views
Search
About R
R Homepage
The R Journal
Software
R Sources
R Binaries
Packages
Other
Documentation
Manuals
FAQs
Contributed
If you want to double-check that the package you have downloaded exactly matches the package distributed by R, you can compare the md5sum of the .exe to the true fingerprint. You will need a version of md5sum for windows: both graphical and command line versions are available.
Frequently asked questions
How do I install R when using Windows Vista?
How do I update packages in my previous version of R?
Should I run 32-bit or 64-bit R?
Please see the R FAQ for general information about R and the R Windows FAQ for Windows-specific information.
Other builds
Patches to this release are incorporated in the r-patched snapshot build.
A build of the development version (which will eventually become the next major release of R) is available in the r-devel snapshot build.
Previous releases
Note to webmasters: A stable link which will redirect to the current Windows binary release is <CRAN MIRROR>/bin/windows/base/release.htm.
Last change: 2012-06-22, by Duncan Murdoch

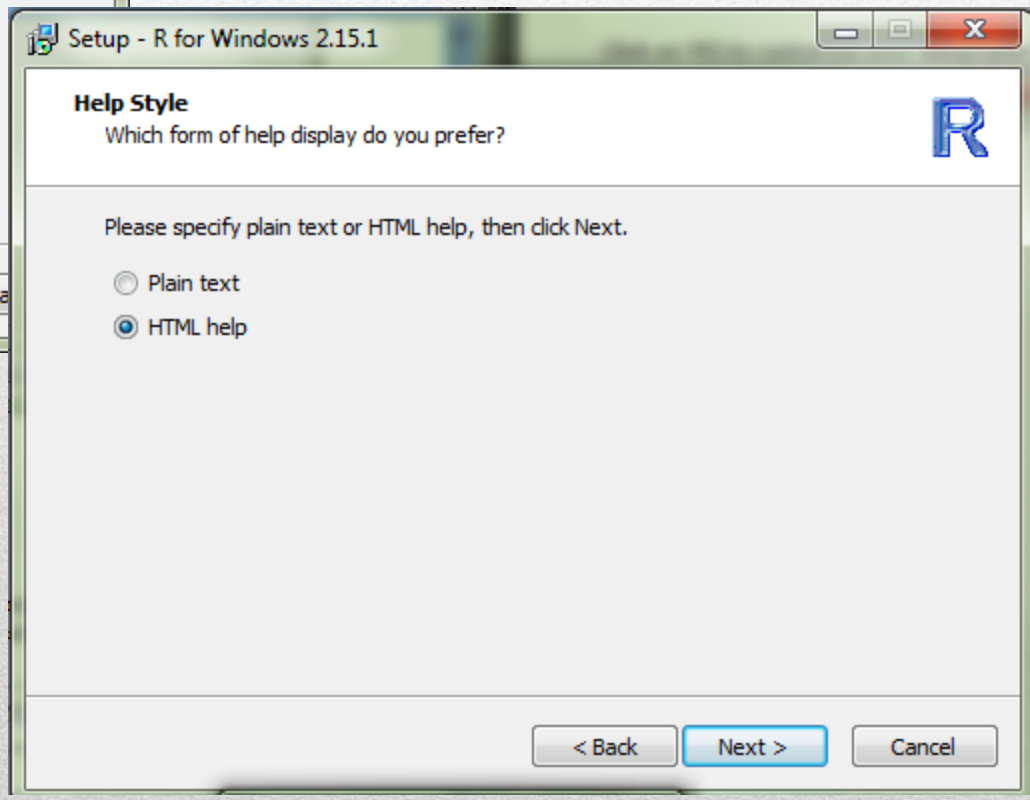

Setup - R for Windows 2.15.1
Information
Please read the following important information before continuing.
When you are ready to continue with Setup, click Next.
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991
Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
Preamble
The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
< Back
Next >

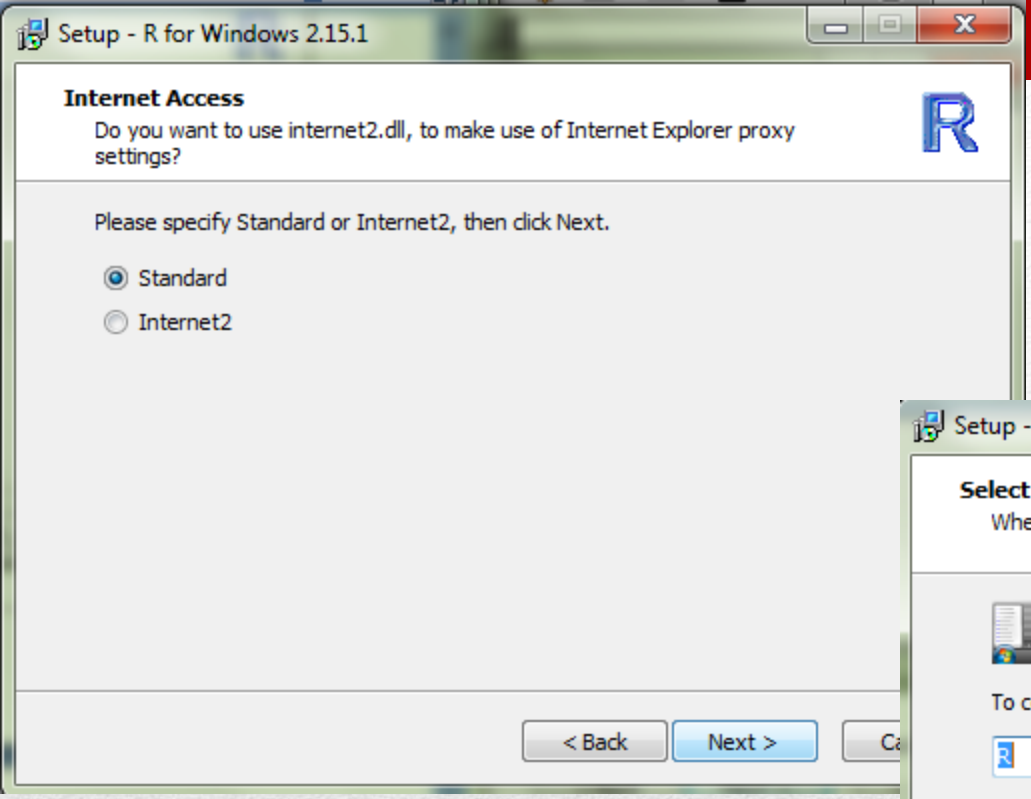

Setup - R for Windows 2.15.1
Select Destination Location
Where should R for Windows 2.15.1 be installed?
Setup will install R for Windows 2.15.1 into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\Program Files\R\R-2.15.1
Browse...
At least 1.2 MB of free disk space is required.
< Back
Next >
Cancel

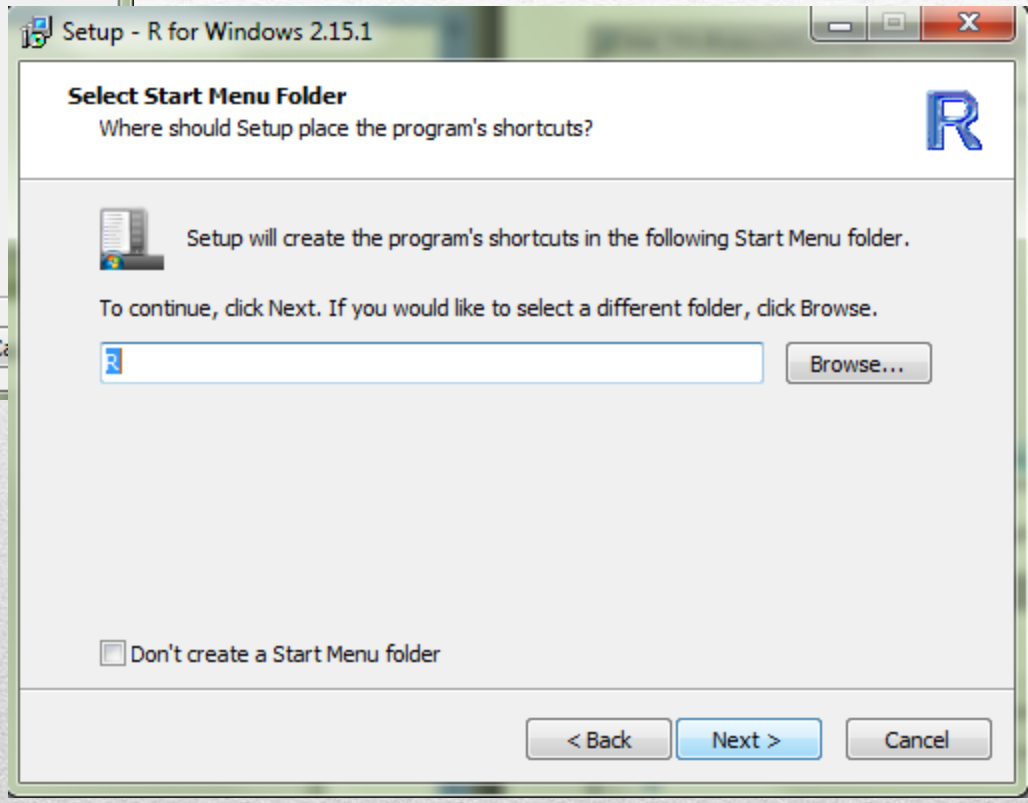

Setup - R for Windows 2.15.1
Select Components
Which components should be installed?
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
User installation
Core Files 61.9 MB
32-bit Files 12.5 MB
64-bit Files 14.4 MB
Current selection requires at least 89.8 MB of disk space.
< Back
Next >

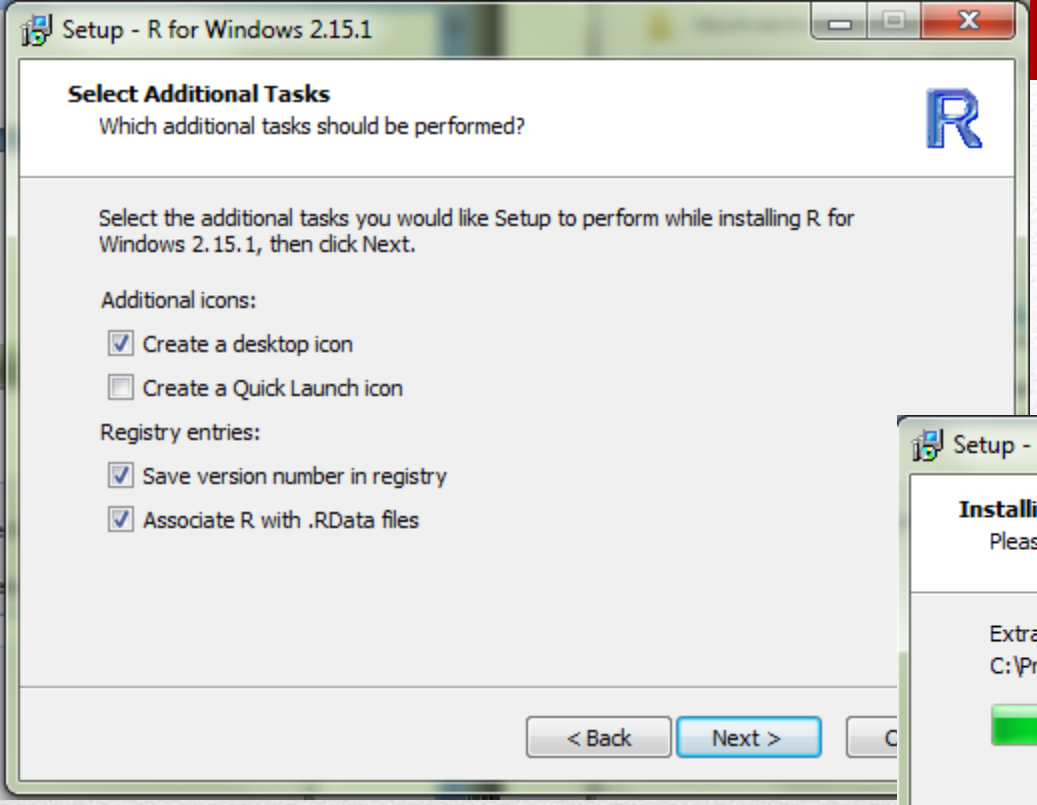

Setup - R for Windows 2.15.1
Startup options
Do you want to customize the startup options?
Please specify yes or no, then click Next.
Yes (customized startup)
No (accept defaults)
< Back
Next >
Cancel

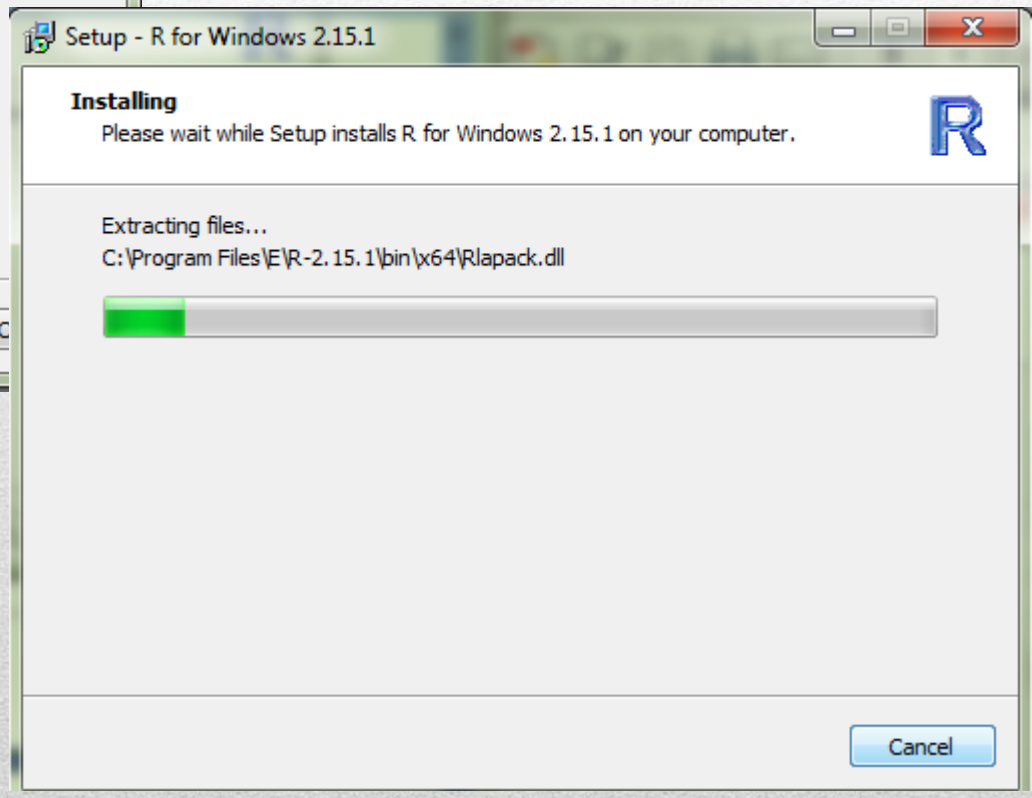

Setup - R for Windows 2.15.1
Display Mode
Do you prefer the MDI or SDI interface?
Please specify MDI or SDI, then click Next.
MDI (one big window)
SDI (separate windows)
< Back
Next >


Setup - R for Windows 2.15.1
Help Style
Which form of help display do you prefer?
Please specify plain text or HTML help, then click Next.
Plain text
HTML help
< Back
Next >
Cancel

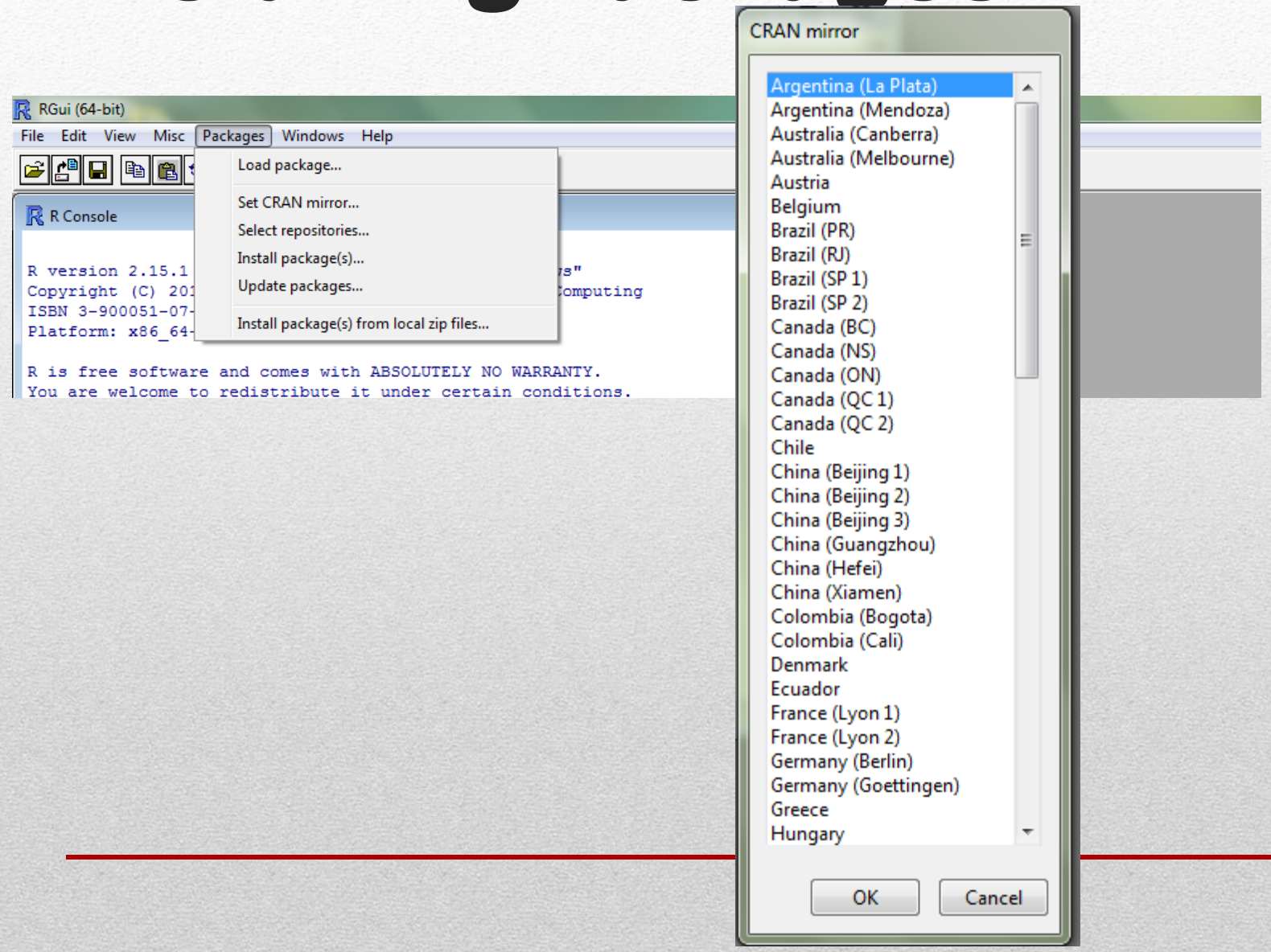

Setup - R for Windows 2.15.1
Internet Access
Do you want to use internet2.dll, to make use of Internet Explorer proxy settings?
Please specify Standard or Internet2, then click Next.
Standard
Internet2
< Back
Next >

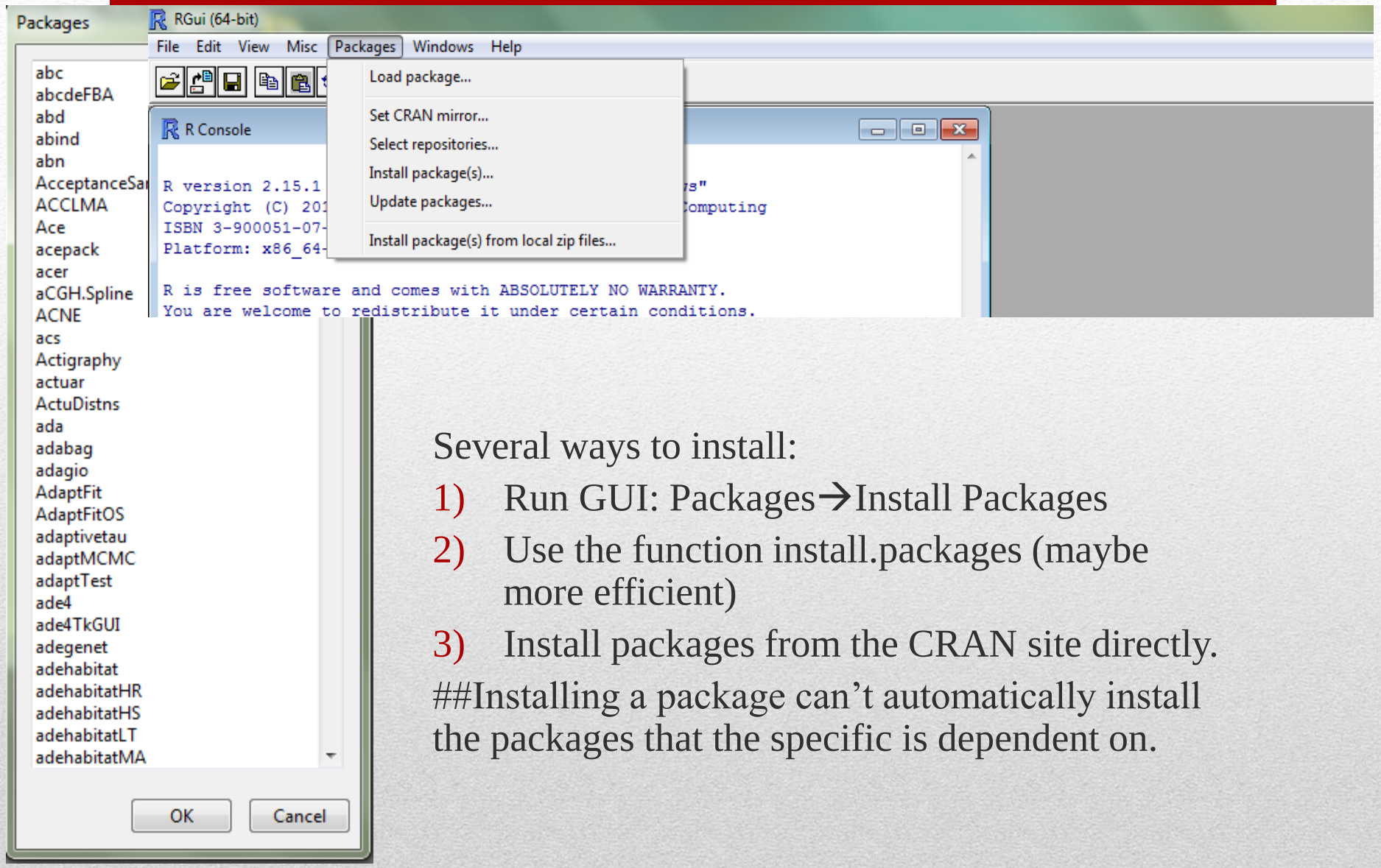

Setup - R for Windows 2.15.1
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
Browse...
Don't create a Start Menu folder
< Back
Next >
Cancel


Setup - R for Windows 2.15.1
Select Additional Tasks
Which additional tasks should be performed?
Select the additional tasks you would like Setup to perform while installing R for Windows 2.15.1, then click Next.
Additional icons:
Create a desktop icon
Create a Quick Launch icon
Registry entries:
Save version number in registry
Associate R with .RData files
< Back
Next >


Setup - R for Windows 2.15.1
Installing
Please wait while Setup installs R for Windows 2.15.1 on your computer.
Extracting files...
C:\Program Files\E\R-2.15.1\bin\x64\Rlapack.dll
Cancel

Installing Packages I





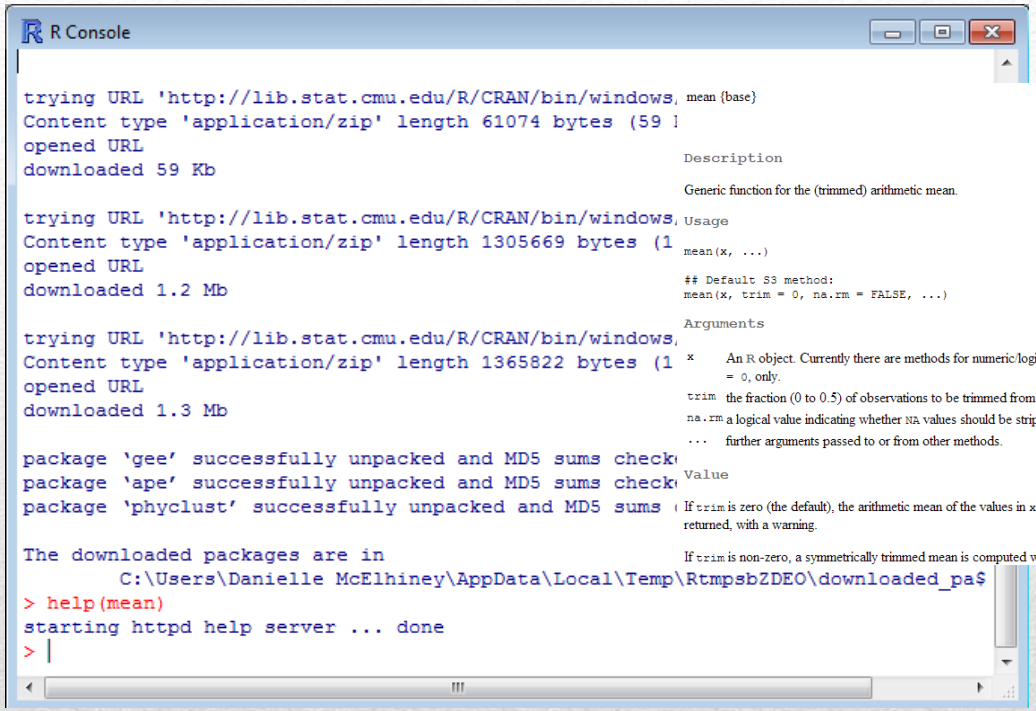
Several ways to install:

1) Run GUI: Packages→Install Packages
2) Use the function install.packages (maybe more efficient)
3) Install packages from the CRAN site directly.

##Installing a package can't automatically install the packages that the specific is dependent on.

Installing Packages II

Using Help Command



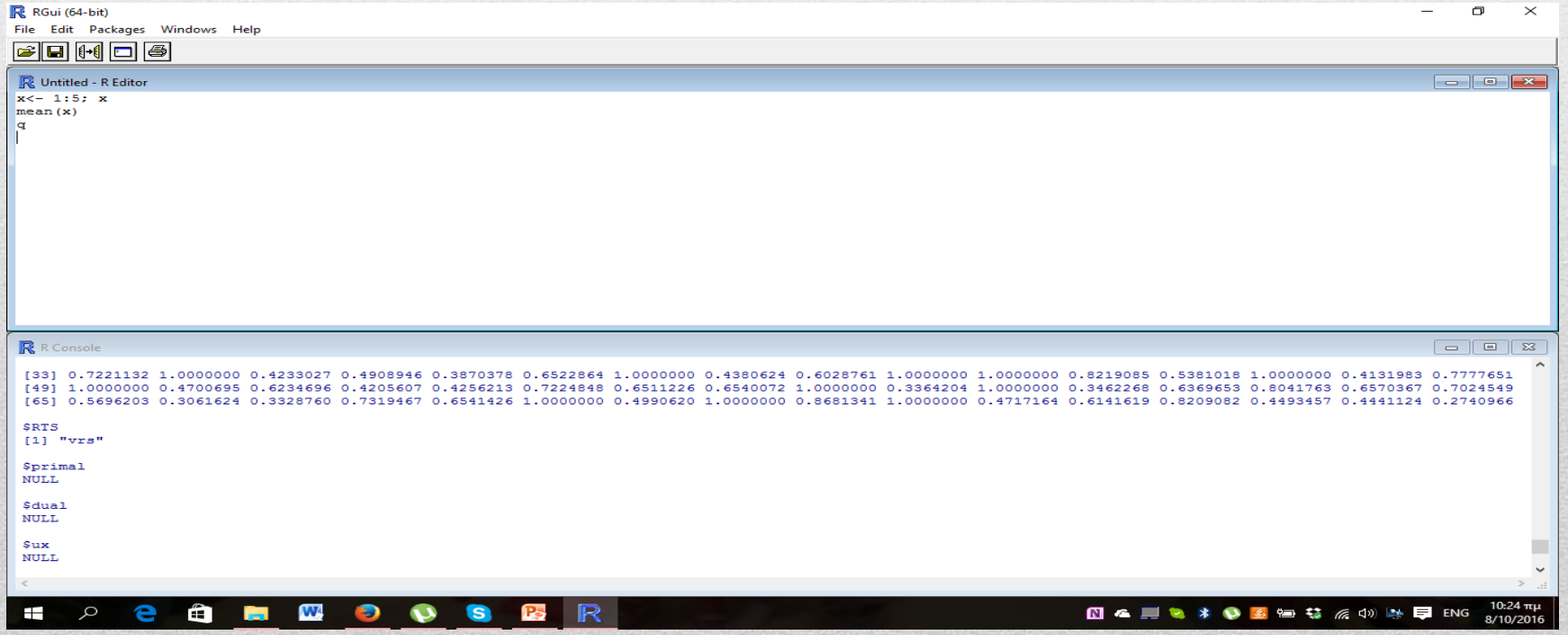
- ?solve
- help.search or ??
  - allows searching for help in various ways

Base R

The base R has two major types of windows R console and editor windows.

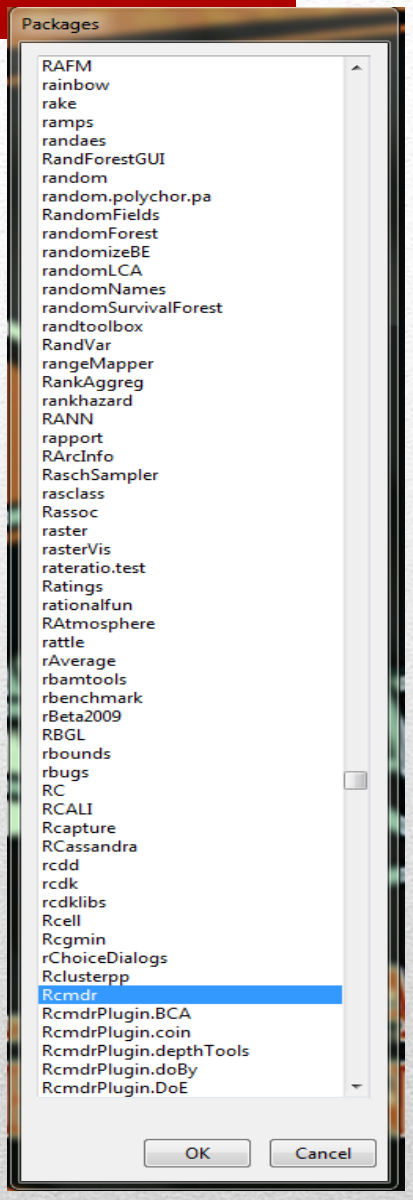
File→ new script or File→open script.

A saved file has an r extension i.e logit1.r



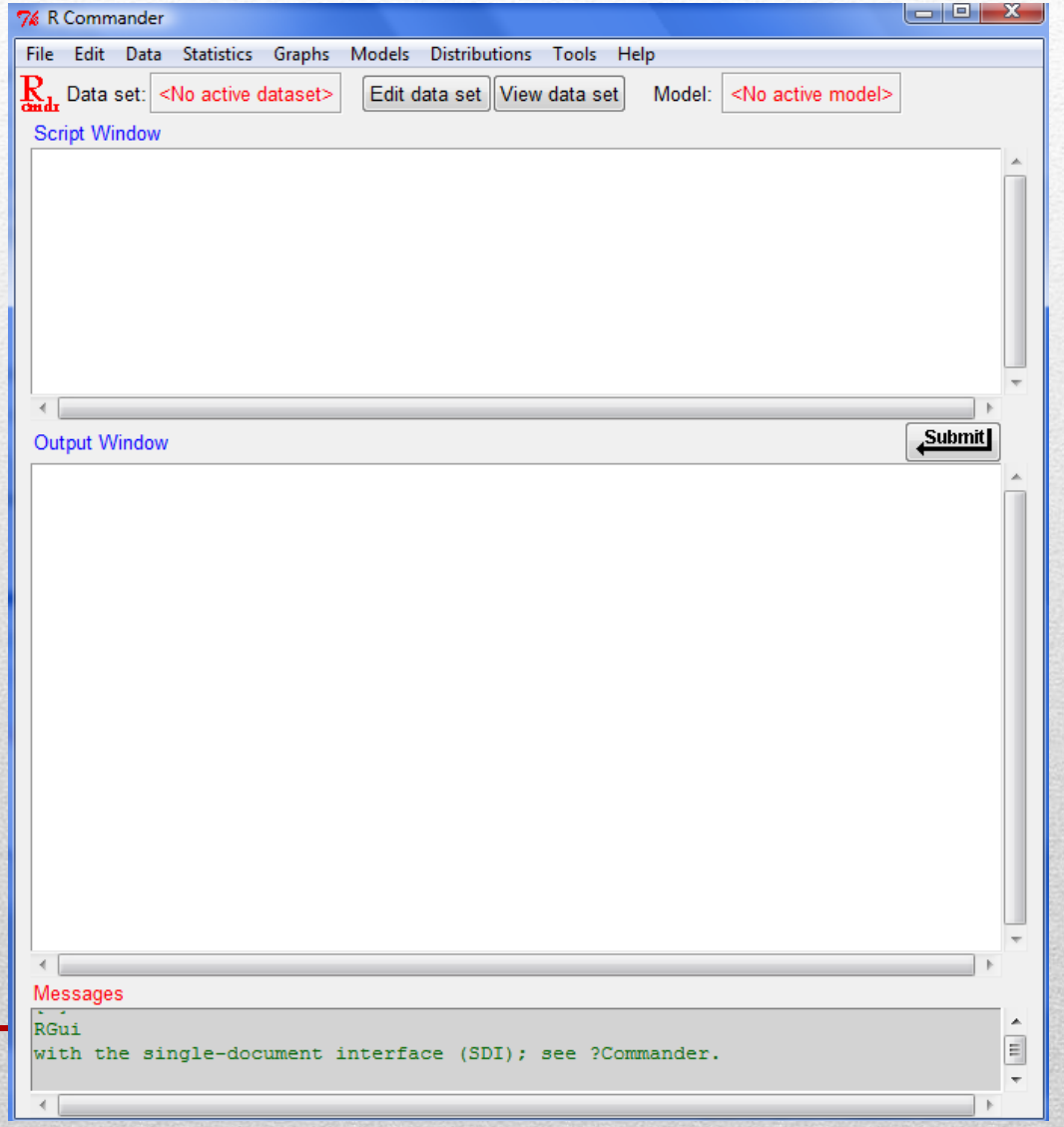
R Commander

- Loading R Commander
  - Packages -> Install Packages -> Cran Mirror Selection -> Rcmdr or install.packages('Rcmdr')



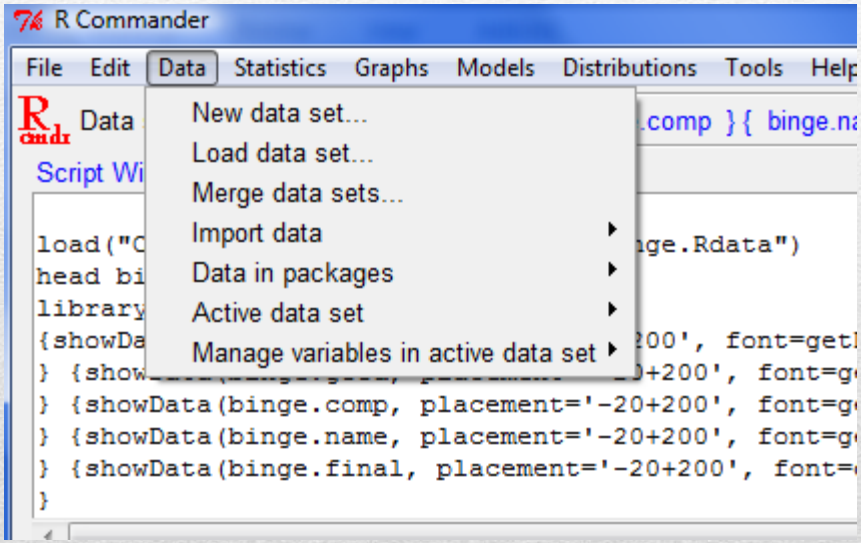
Opening R Commander

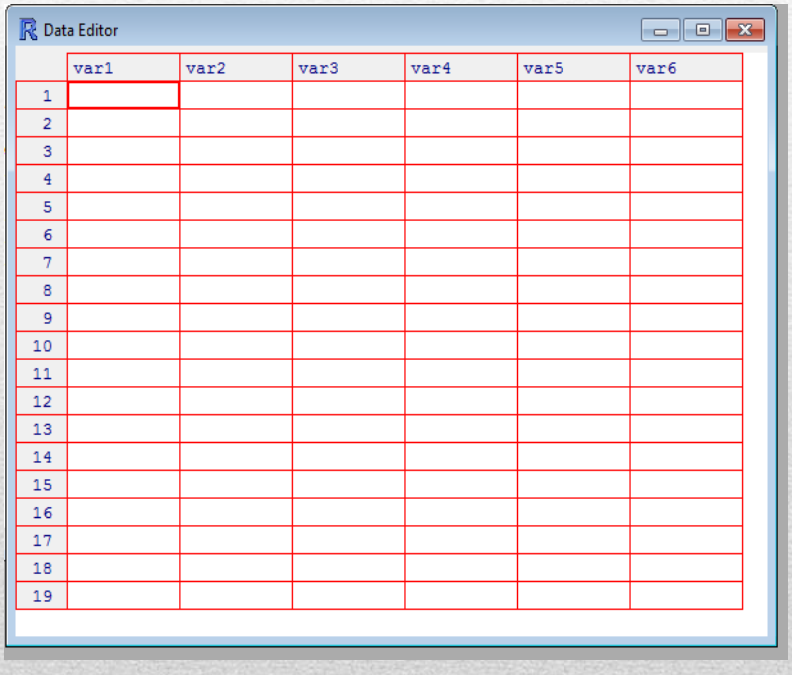
Open R -> Packages -> Load Packages -> Rcmdr



Loading Data with R Commander

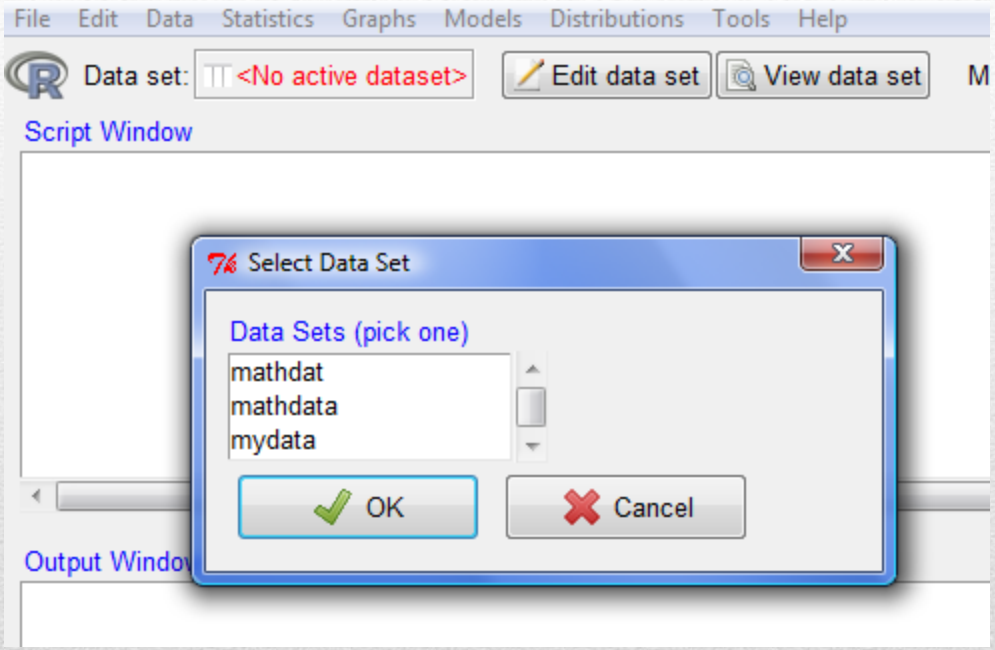
- Data -> Load data



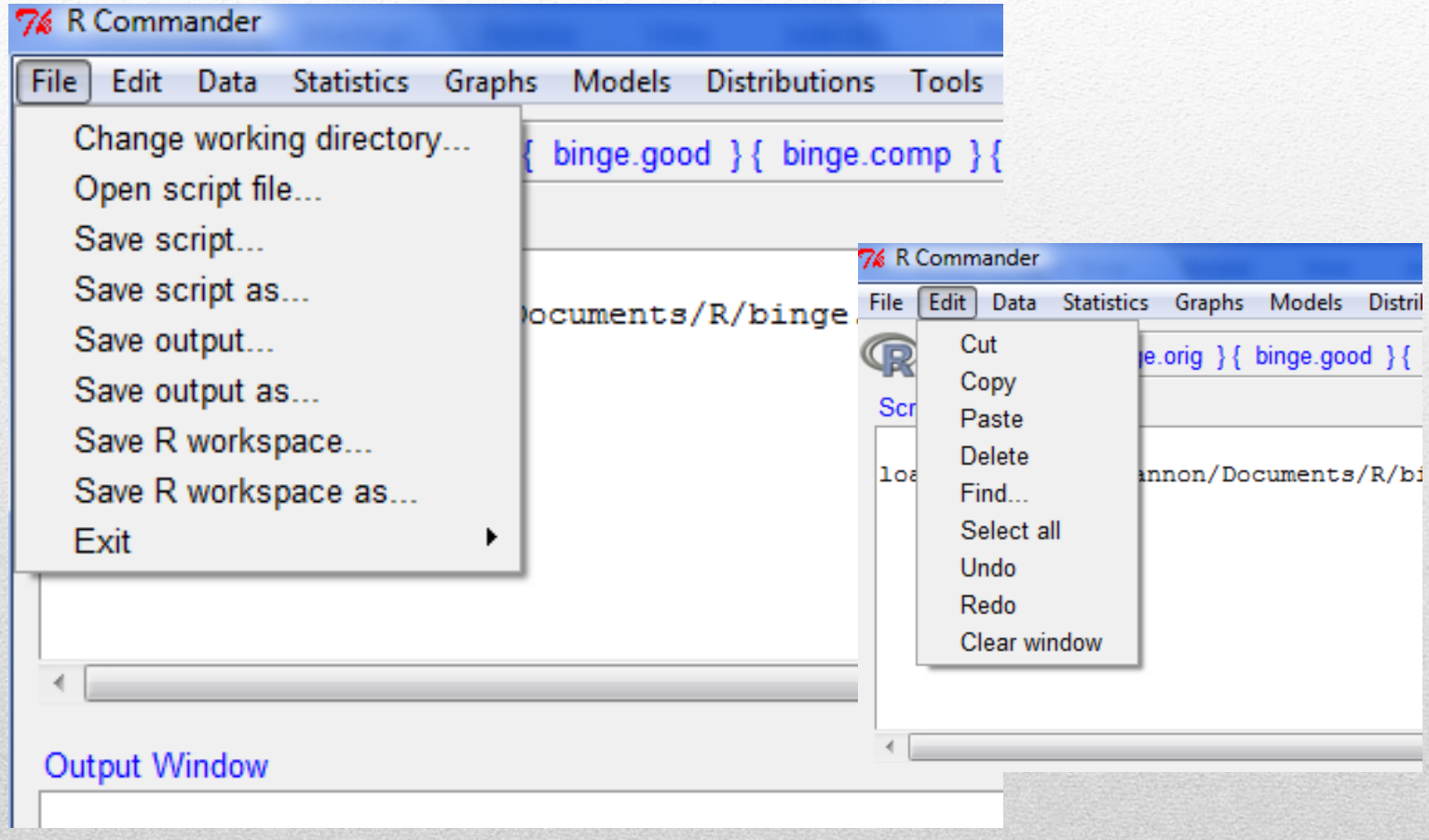


Active Data with R Commander

Data ->Active data set -> Select active data set

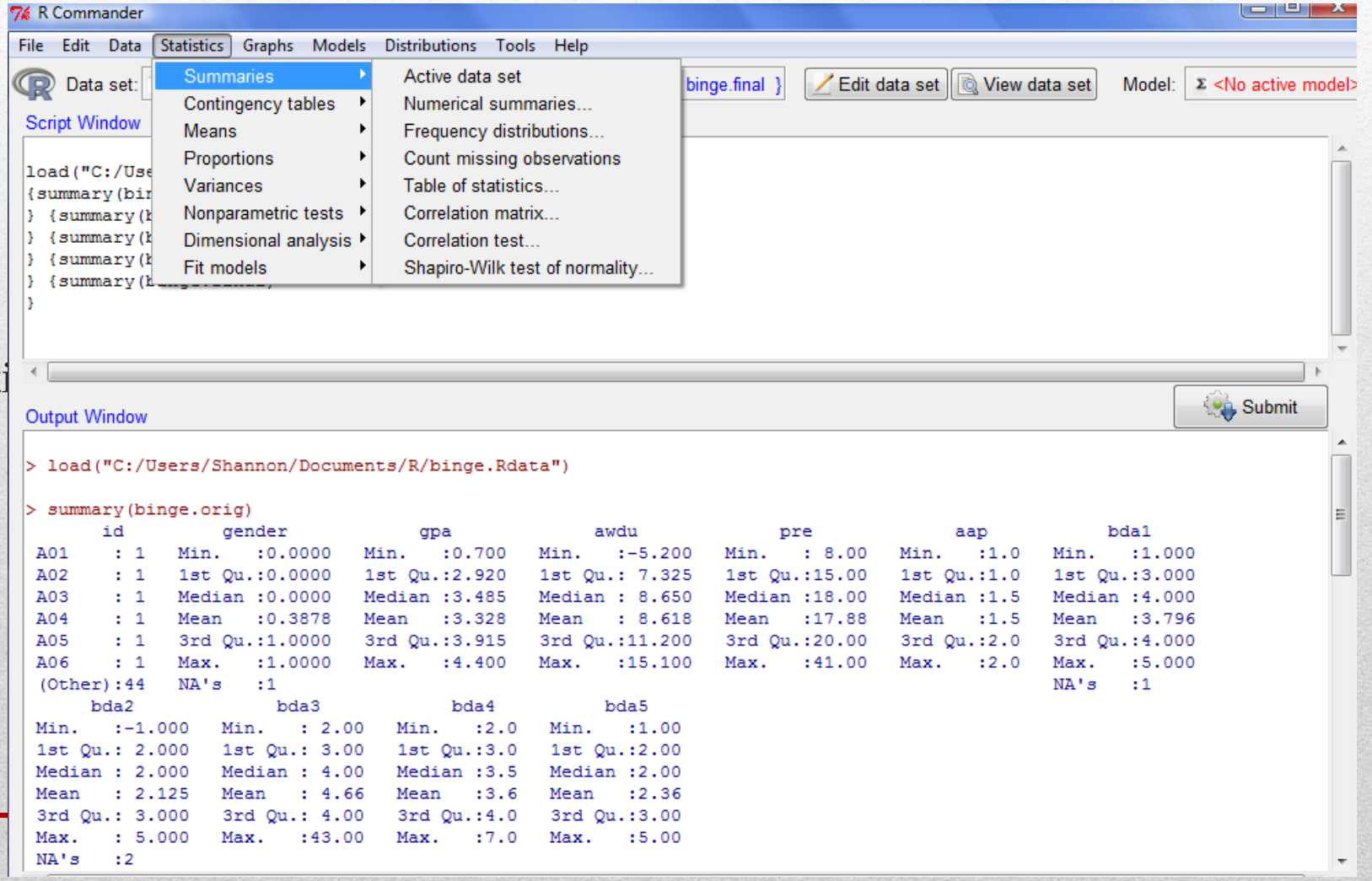


File/Edit Options

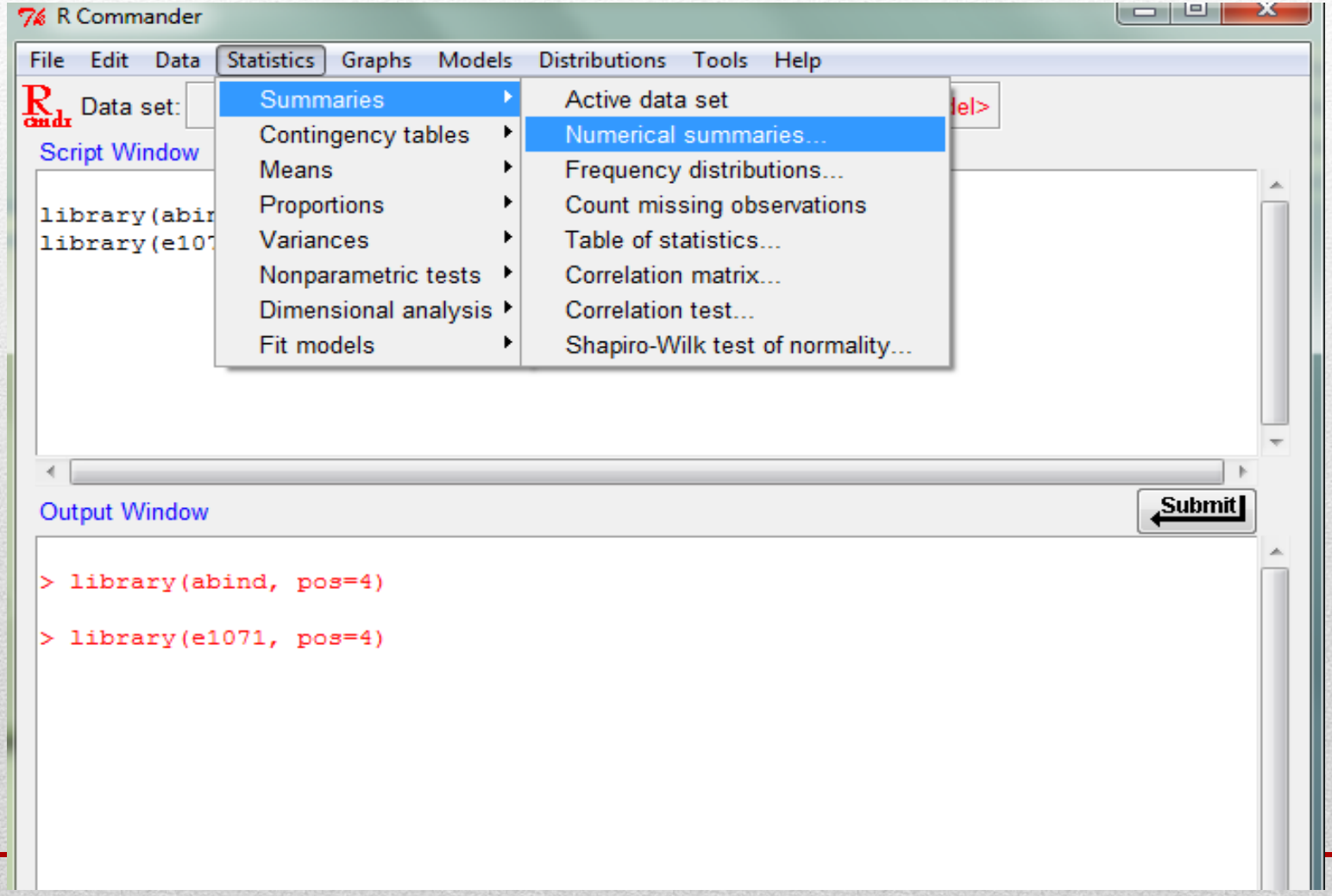


Summaries

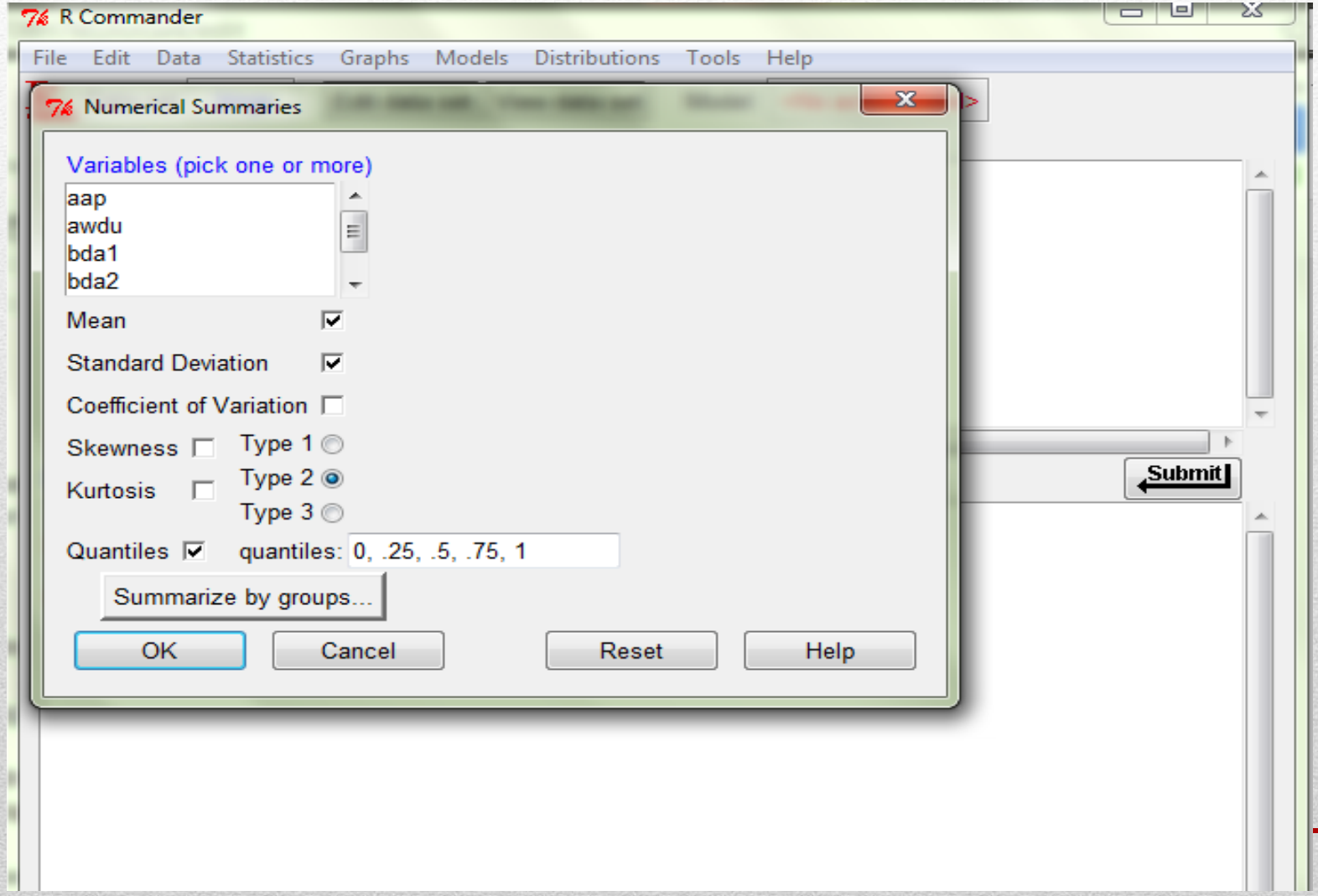
Statisti

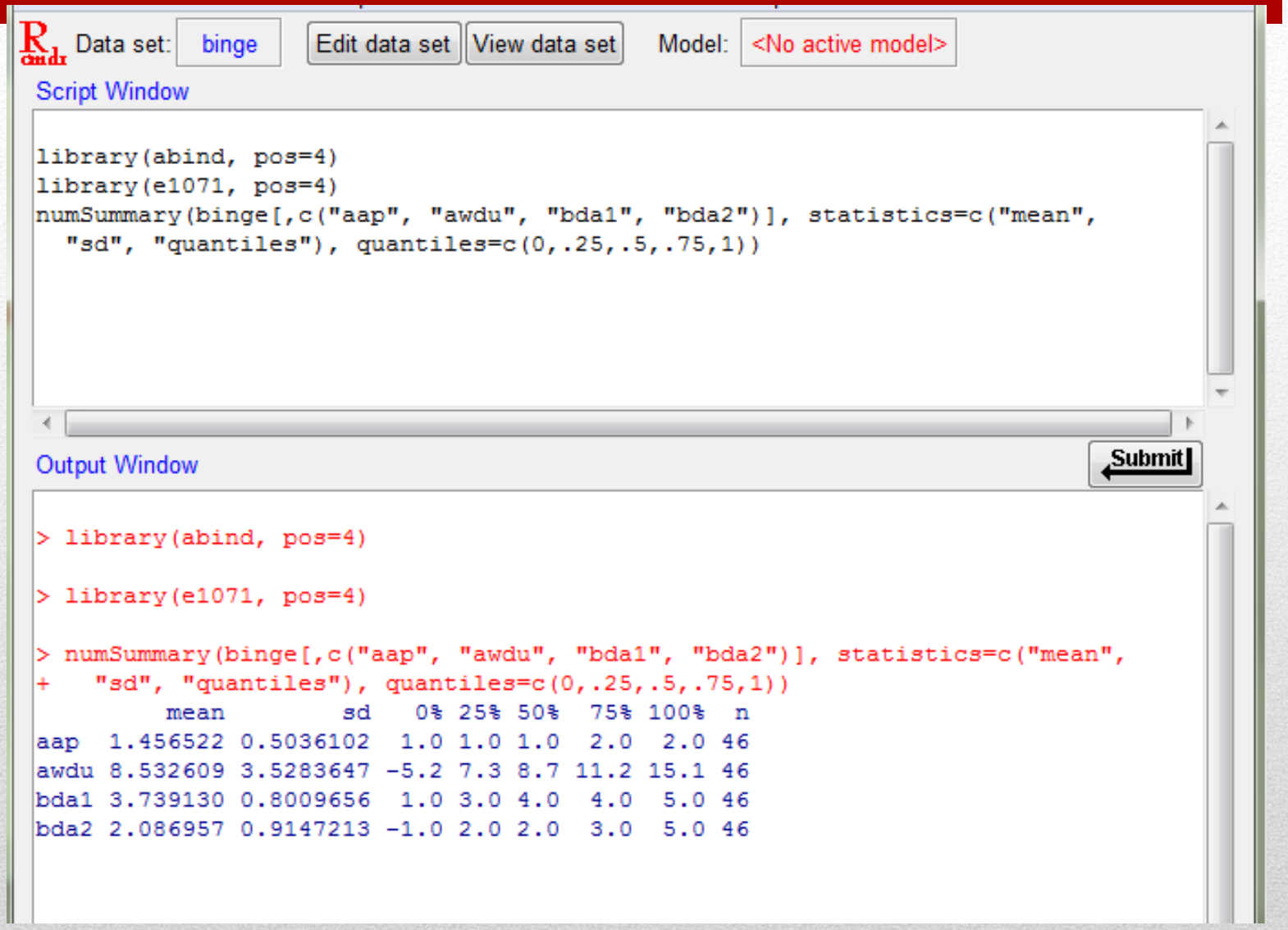


Descriptive Statistics

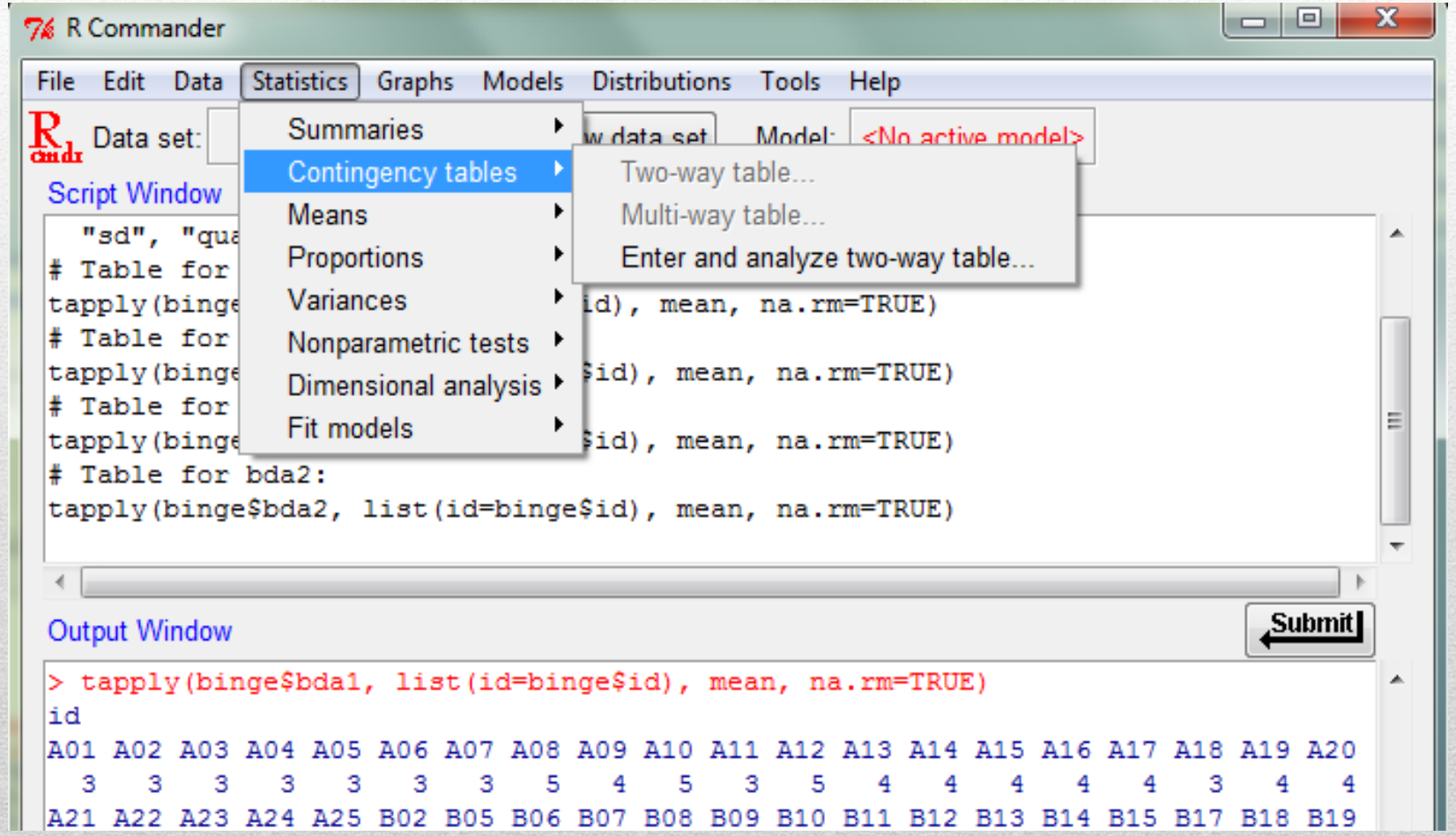


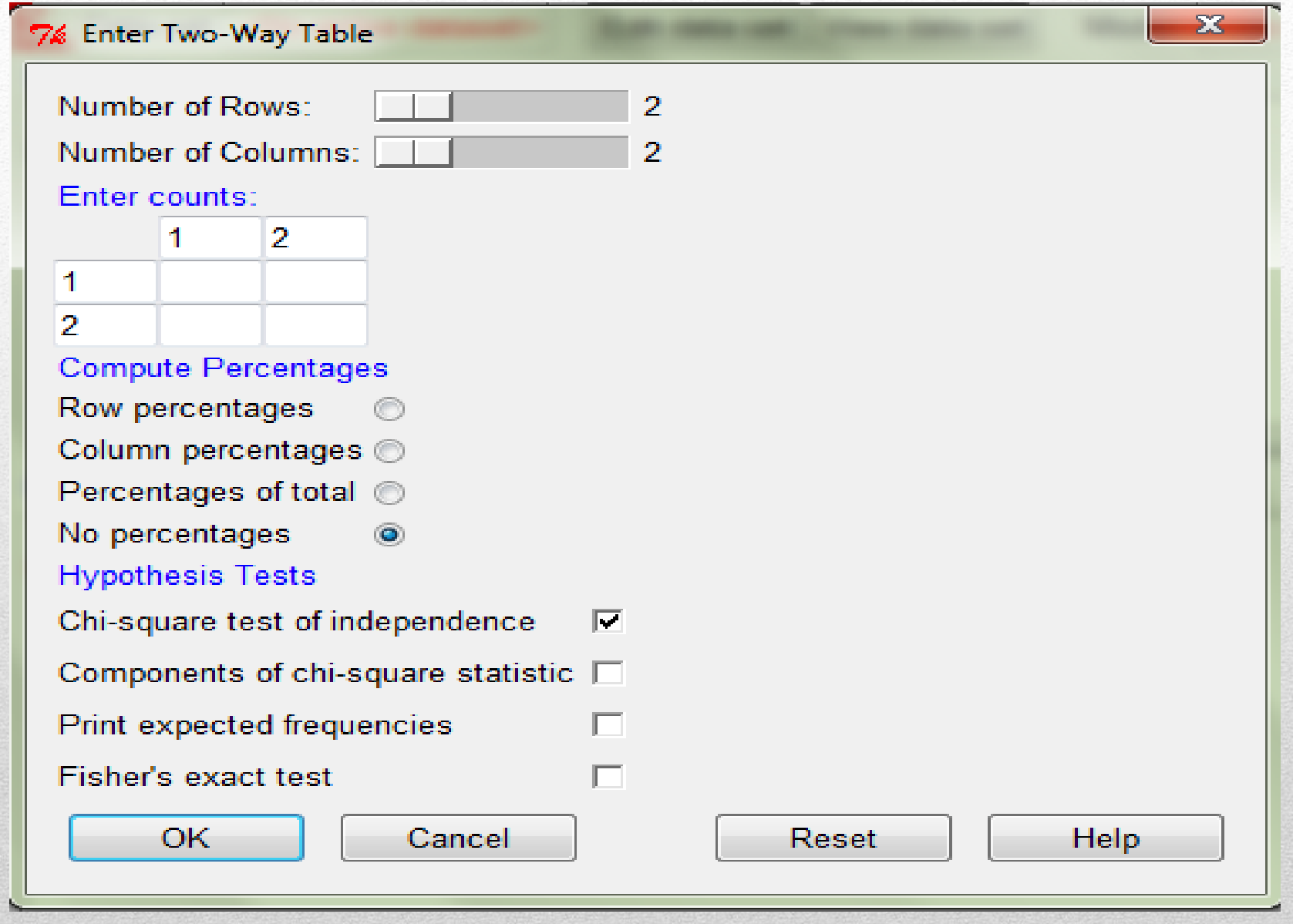
Mean, Standard Deviation, Skewness, Kurtosis




Data set: binge
Edit data set
View data set
Model: <No active model>
Script Window
library(abind, pos=4)
library(e1071, pos=4)
numSummary(binge[,c("aap", "awdu", "bda1", "bda2")], statistics=c("mean",
"sd", "quantiles"), quantiles=c(0,.25,.5,.75,1))
Output Window
Submit
> library(abind, pos=4)
> library(e1071, pos=4)
> numSummary(binge[,c("aap", "awdu", "bda1", "bda2")], statistics=c("mean",
+ "sd", "quantiles"), quantiles=c(0,.25,.5,.75,1))
mean sd 0% 25% 50% 75% 100% n
aap 1.456522 0.5036102 1.0 1.0 1.0 2.0 2.0 46
awdu 8.532609 3.5283647 -5.2 7.3 8.7 11.2 15.1 46
bda1 3.739130 0.8009656 1.0 3.0 4.0 4.0 5.0 46
bda2 2.086957 0.9147213 -1.0 2.0 2.0 3.0 5.0 46

Contingency Tables




Enter Two-Way Table
Number of Rows: 2
Number of Columns: 2
Enter counts:
1
2
1
2
Compute Percentages
Row percentages
Column percentages
Percentages of total
No percentages
Hypothesis Tests
Chi-square test of independence
Components of chi-square statistic
Print expected frequencies
Fisher's exact test
OK
Cancel
Reset
Help

Script Window

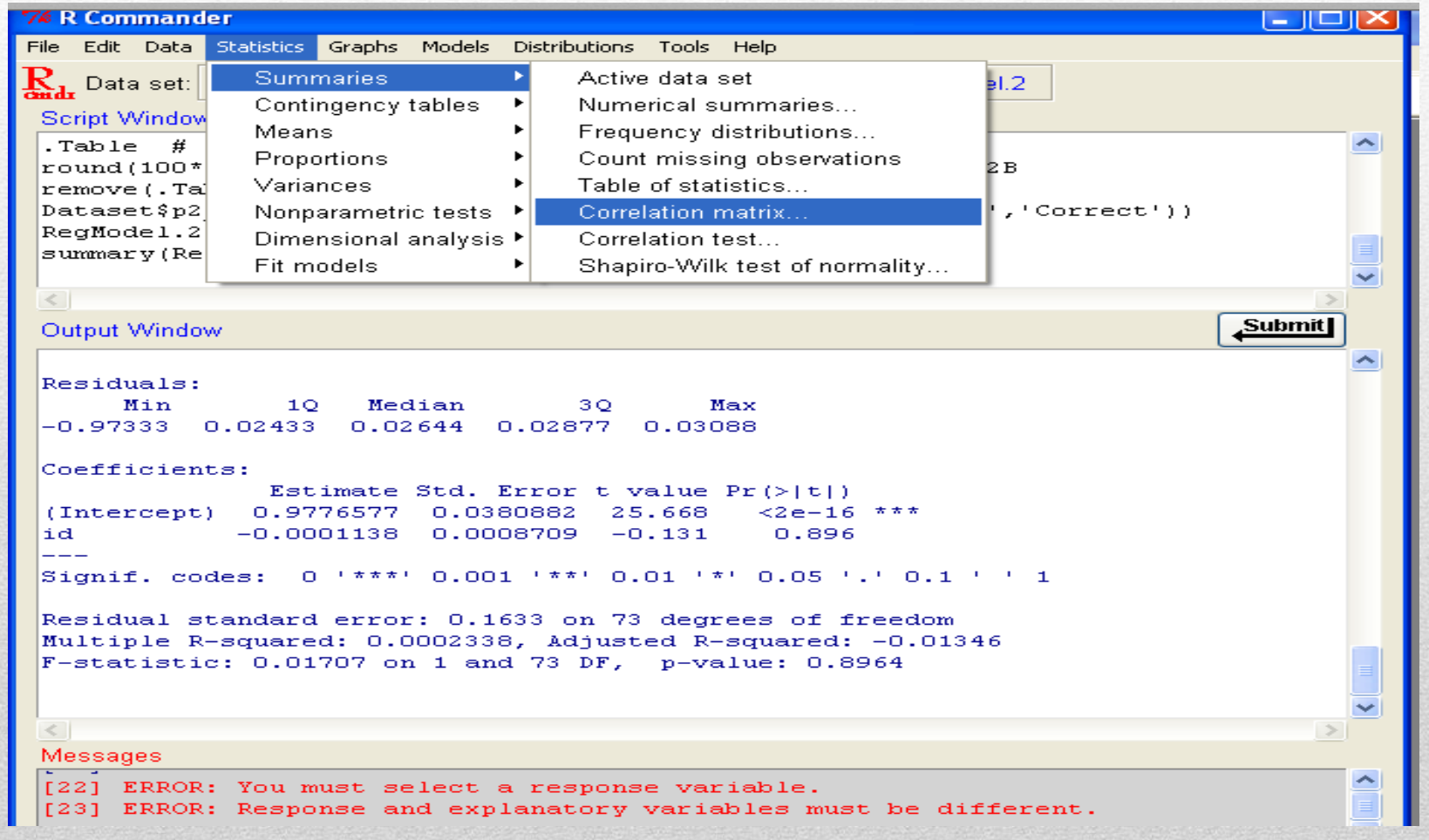
```
library(abind, pos=4)
.Table <- matrix(c(13,412,399,15), 2, 2, byrow=TRUE)
rownames(.Table) <- c('1', '2')
colnames(.Table) <- c('1', '2')
.Table  # Counts
.Test <- chisq.test(.Table, correct=FALSE)
.Test
remove(.Test)
remove(.Table)
```

Output Window

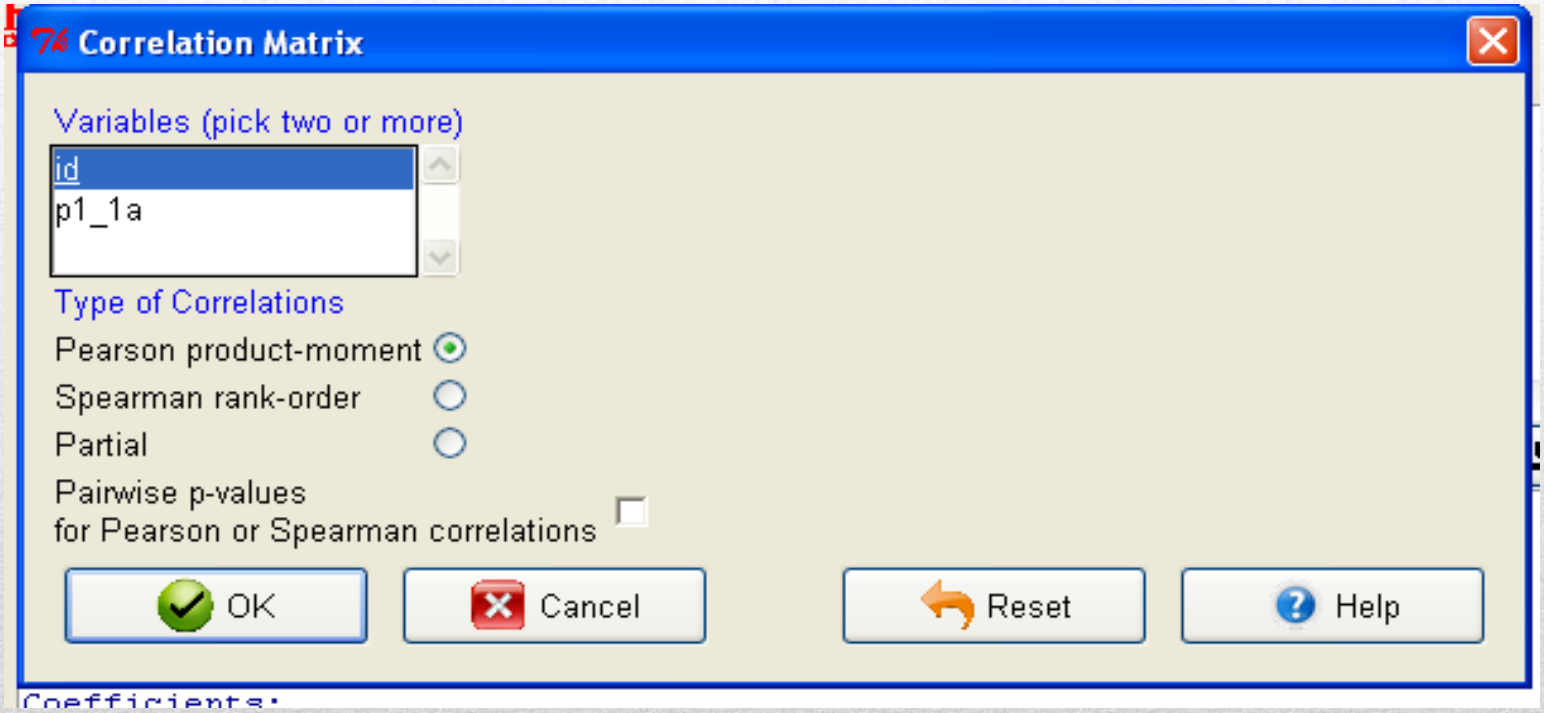
Submit

```
> .Table  # Counts
    1   2
1  13 412
2 399  15

> .Test <- chisq.test(.Table, correct=FALSE)

> .Test

        Pearson's Chi-squared test

data:  .Table
X-squared = 730.7299, df = 1, p-value < 2.2e-16


> remove(.Test)

> remove(.Table)
```

Correlations in R Commander



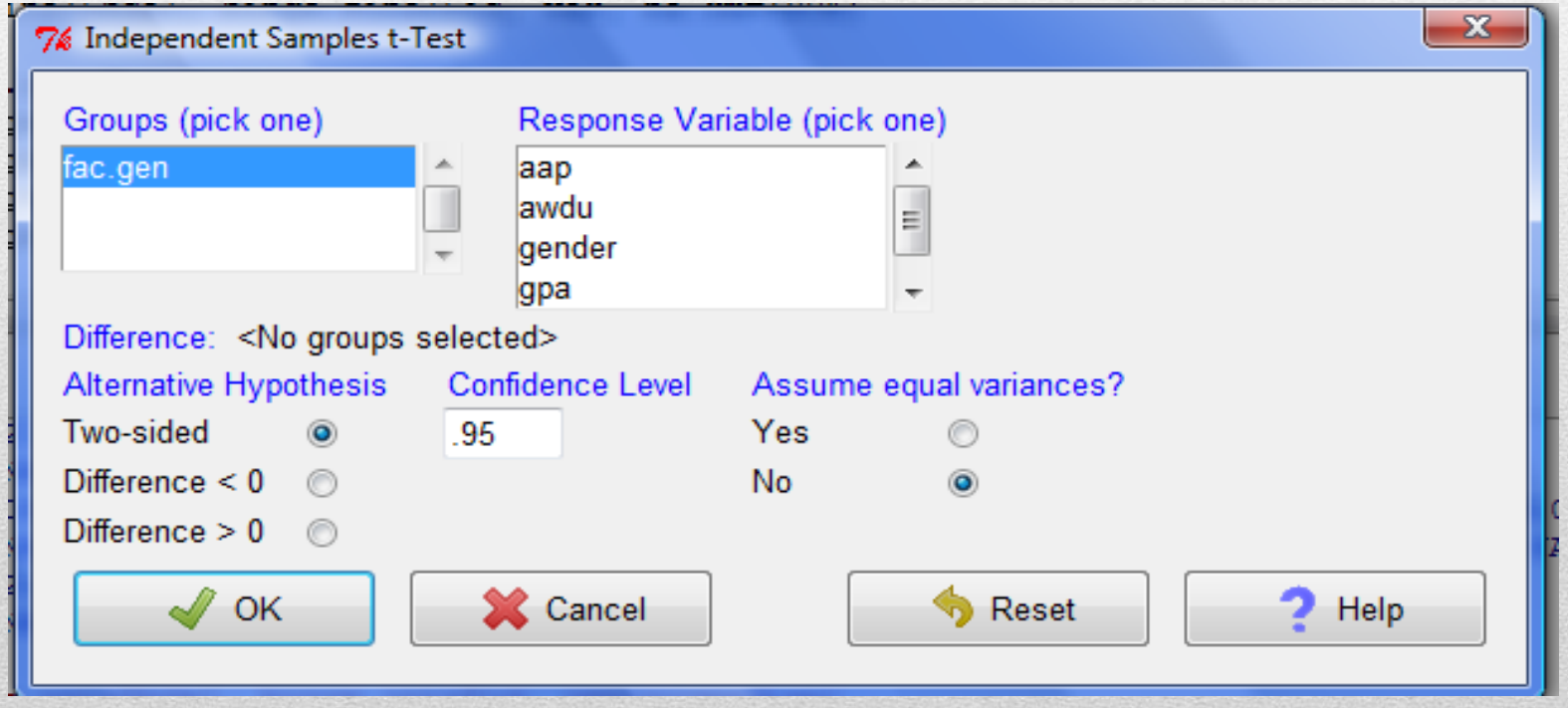
Correlations in R Commander



Independent T-Test

Statistics ->

Independent T Test



Script Window

```
load("C:/Users/Shannon/Documents/R/binge.Rdata")
t.test(awdu~fac.gen, alternative='two.sided', conf.level=.95,
  var.equal=FALSE, data=binge.final)
tapply(binge.final$awdu, binge.final$fac.gen, var, na.rm=TRUE)
leveneTest(binge.final$awdu, binge.final$fac.gen, center=median)
```

Submit

Output Window

```
> load("C:/Users/Shannon/Documents/R/binge.Rdata")

> t.test(awdu~fac.gen, alternative='two.sided', conf.level=.95,
+   var.equal=FALSE, data=binge.final)

        Welch Two Sample t-test

data:  awdu by fac.gen
t = -1.1991, df = 33.405, p-value = 0.2389
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.9636463  0.7650749
sample estimates:
mean in group 0 mean in group 1
       8.360714        9.460000


> tapply(binge.final$awdu, binge.final$fac.gen, var, na.rm=TRUE)
        0         1
```

Script Window

```
load("C:/Users/Shannon/Documents/R/binge.Rdata")
t.test(awdu~fac.gen, alternative='two.sided', conf.level=.95,
  var.equal=FALSE, data=binge.final)
tapply(binge.final$awdu, binge.final$fac.gen, var, na.rm=TRUE)
leveneTest(binge.final$awdu, binge.final$fac.gen, center=median)
```

Submit

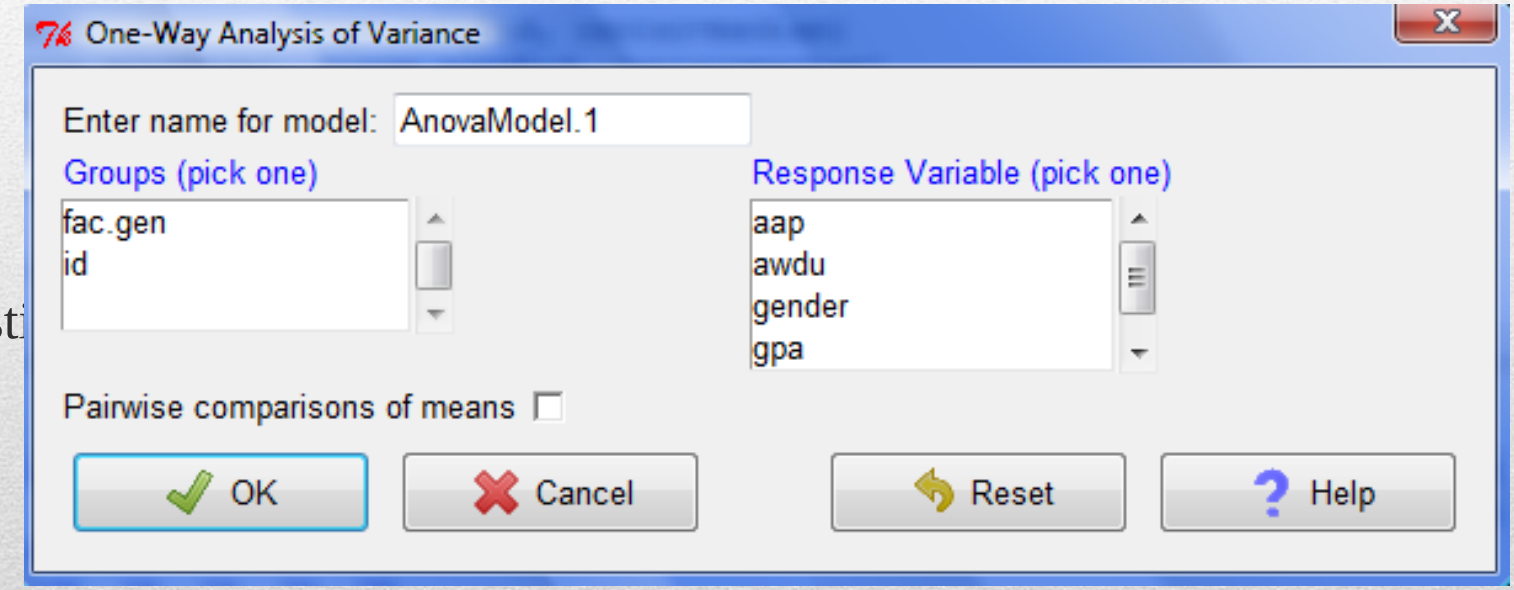
Output Window

```
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.9636463  0.7650749
sample estimates:
mean in group 0 mean in group 1
       8.360714        9.460000


> tapply(binge.final$awdu, binge.final$fac.gen, var, na.rm=TRUE)
        0         1
 6.698029 12.025684

> leveneTest(binge.final$awdu, binge.final$fac.gen, center=median)
Levene's Test for Homogeneity of Variance (center = median)
      Df F value  Pr(>F)
group  1  3.8056 0.05719 .
      46
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```


```
> leveneTest(binge.final$awdu, binge.final$fac.gen, center=median)
Levene's Test for Homogeneity of Variance (center = median)
      Df F value  Pr(>F)
group  1  3.8056 0.05719 .
      46
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

One Way ANOVA

Statisti



Script Window

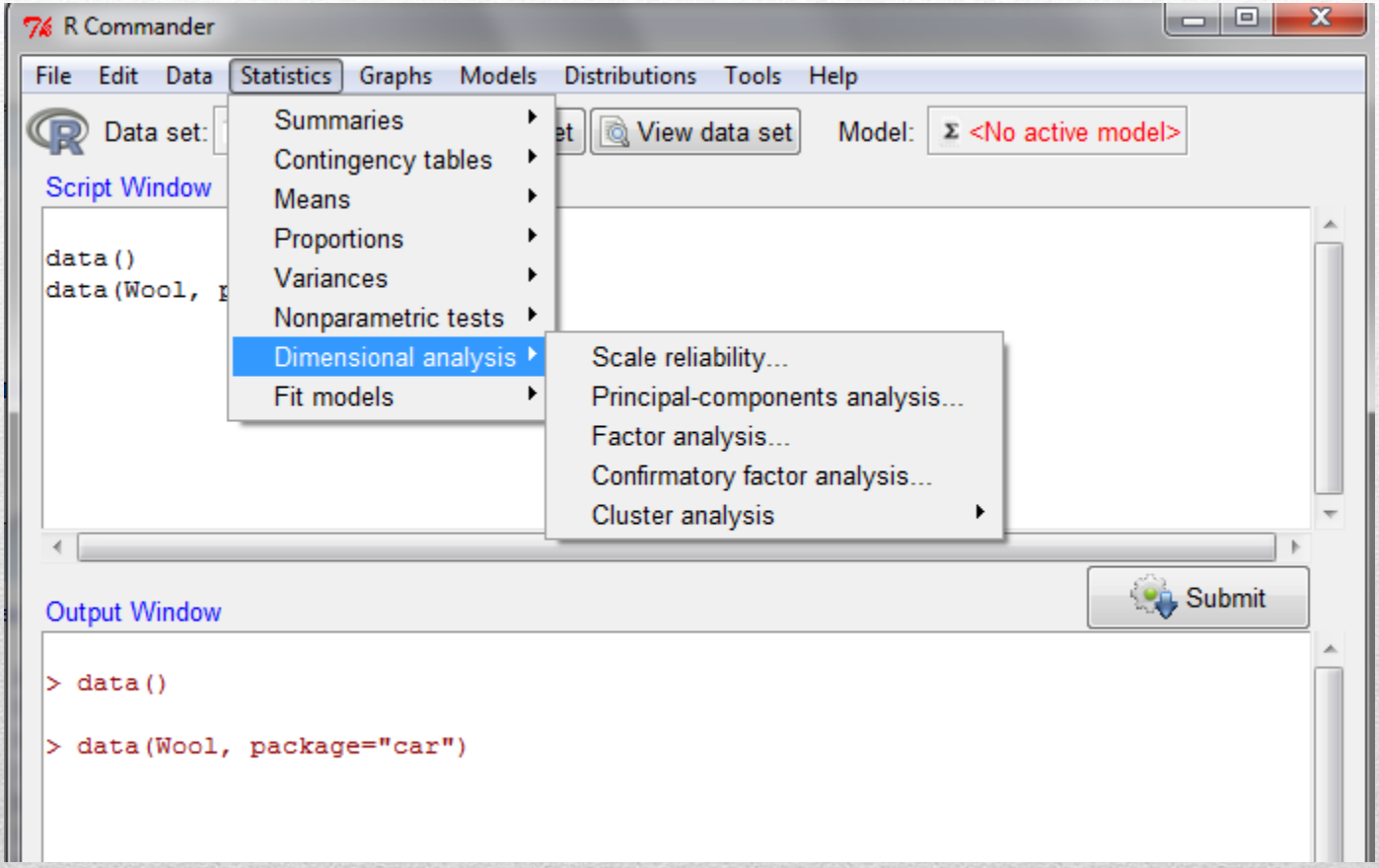
```
  var.equal=FALSE, data=binge.final)
tapply(binge.final$awdu, binge.final$fac.gen, var, na.rm=TRUE)
leveneTest(binge.final$awdu, binge.final$fac.gen, center=median)
library(multcomp, pos=4)
library(abind, pos=4)
AnovaModel.1 <- aov(awdu ~ fac.gen, data=binge.final)
summary(AnovaModel.1)
numSummary(binge.final$awdu , groups=binge.final$fac.gen,
  statistics=c("mean", "sd"))
```

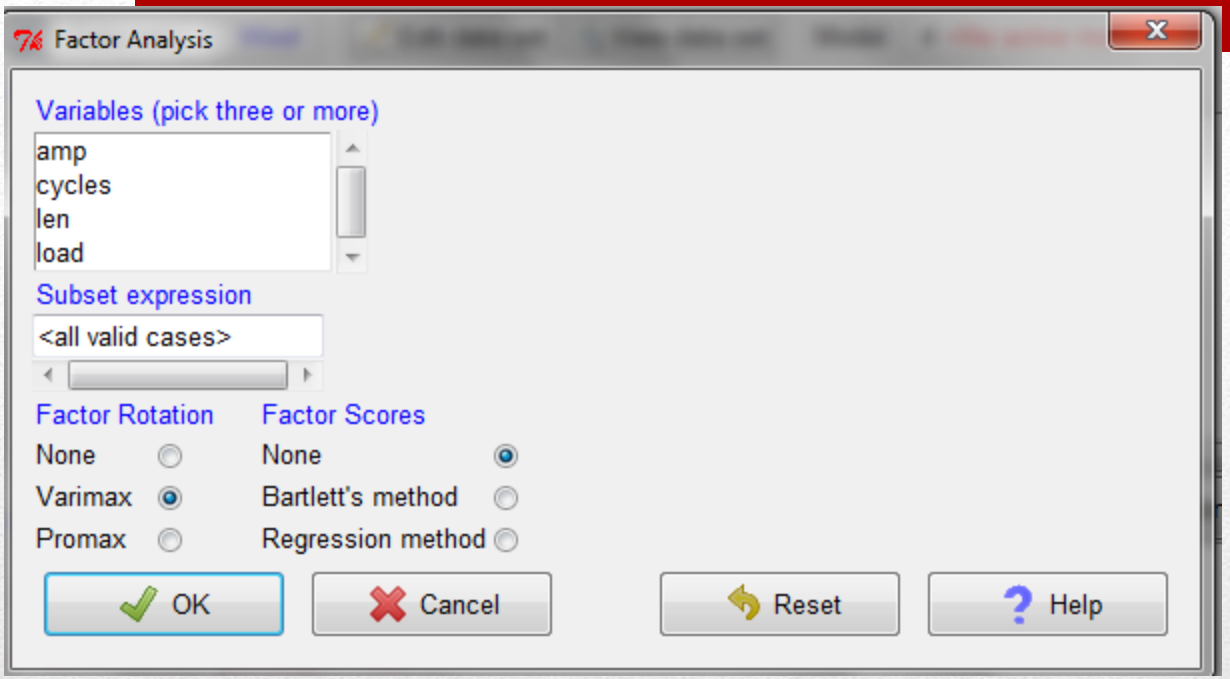
Submit

Output Window

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> library(multcomp, pos=4)

> library(abind, pos=4)

> AnovaModel.1 <- aov(awdu ~ fac.gen, data=binge.final)

> summary(AnovaModel.1)
             Df Sum Sq Mean Sq F value Pr(>F)
fac.gen       1   14.1  14.098   1.584  0.214
Residuals    46  409.3   8.899
2 observations deleted due to missingness

> numSummary(binge.final$awdu , groups=binge.final$fac.gen,
+   statistics=c("mean", "sd"))
      mean       sd data:n data:NA
0 8.360714 2.588055     28       2
1 9.460000 3.467807     20       0
```

Factor Analysis



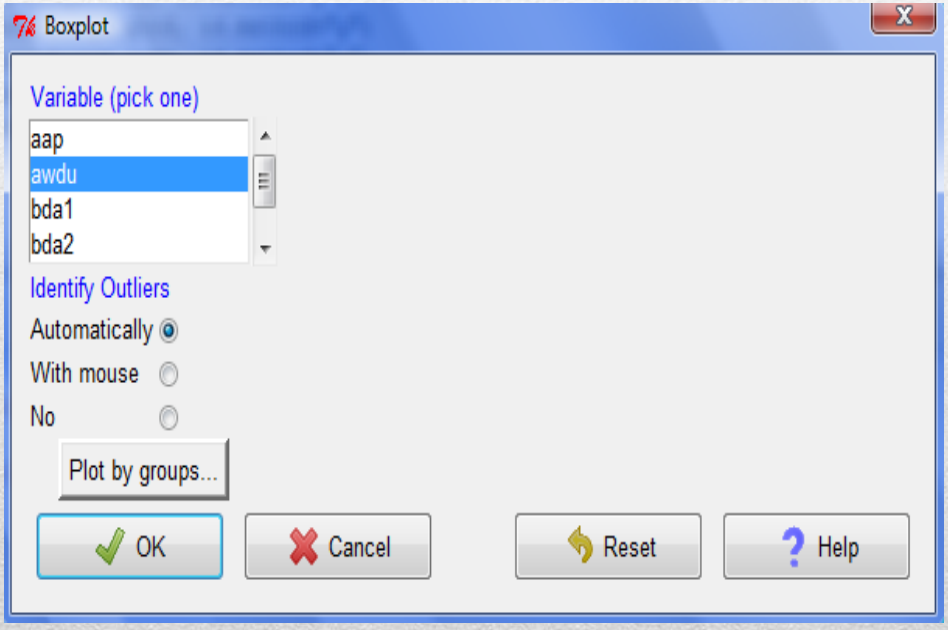


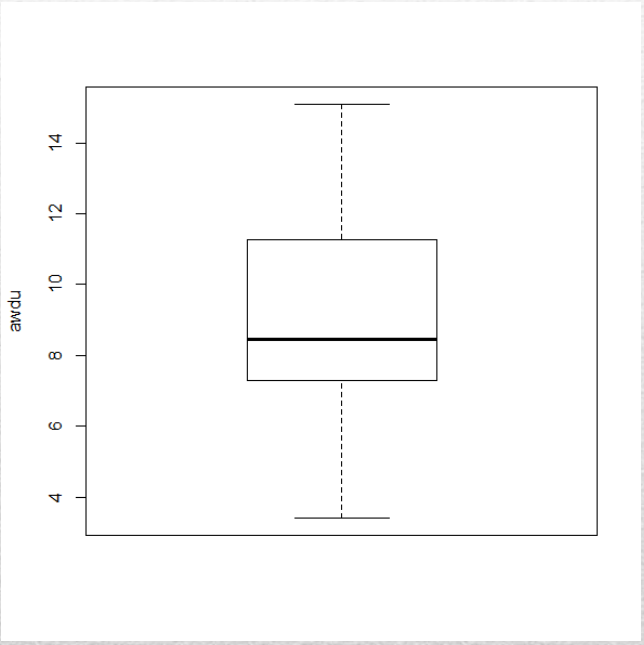
Script Window

```
.FA <- factanal(~aap+awdu+bda1, factors=1, rotation="varimax",
  scores="none", data=binge.orig)
.FA
remove(.FA)
library(sem, pos=4)
```

```
> .FA <- factanal(~aap+awdu+bda1, factors=1, rotation="varimax",
+   scores="none", data=binge.orig)

> .FA

Call:
factanal(x = ~aap + awdu + bda1, factors = 1, data = binge.orig,     scores = "none", rotation = "varimax")

Uniquenesses:
  aap  awdu  bda1
0.849 0.324 0.596

Loadings:
     Factor1
aap  0.388
awdu 0.822
bda1 0.636

               Factor1
SS loadings      1.231
Proportion Var   0.410

The degrees of freedom for the model is 0 and the fit was 0

> remove(.FA)

> library(sem, pos=4)
```

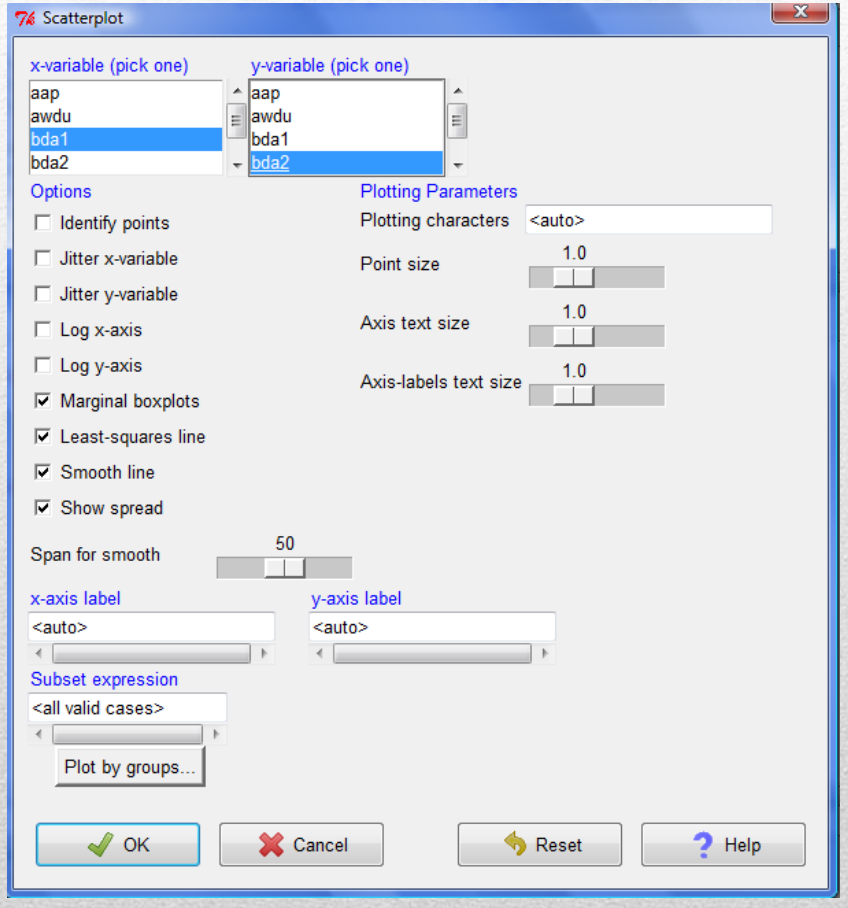
Graphs in R Commander Box Plot



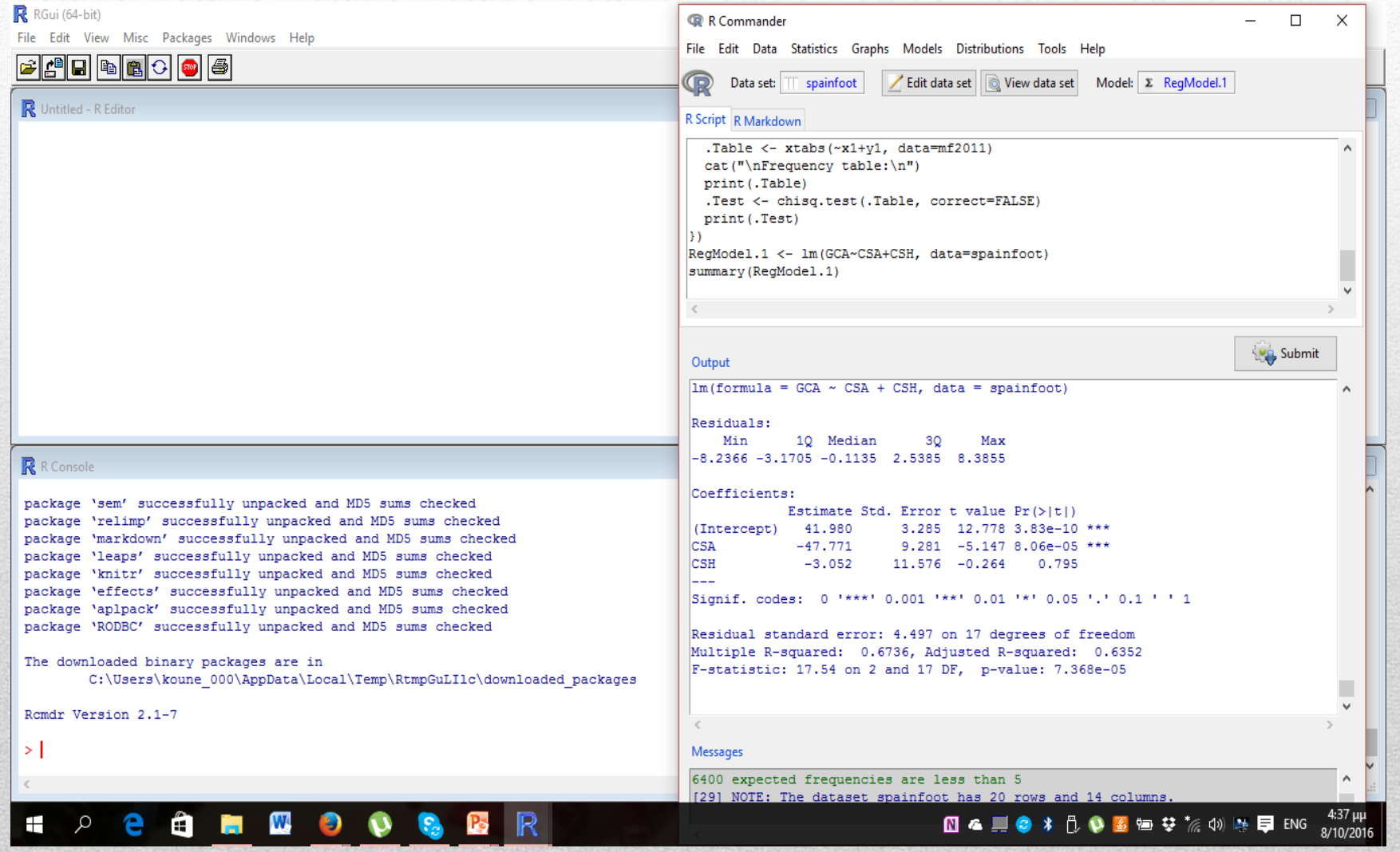


Graphs in R Commander Scatter Plot

Graphs -> Scatter Plot



Linear regression



Data Inputs and creation in R

- BB <- read.csv(file="heisenberg.csv",head=TRUE,sep=",")
- dir()
- getwd()
- BB <- read.csv(file="heisenberg.csv",head=TRUE,sep=",")
- library(nonparaeff)
- data(heisenberg)
- attributes(heisenberg)
- is.data.frame(heisenberg)

- ls()
- remove(x,y,...)
- rm(x)
- x=c(1.2,2,3,4,5,6)
- dat<-data.frame(x=c(1:10,1:10), y=1:20)
- attach(dat)
- x+y
- rm(x)
- x
- setwd("f:/temp")
- getwd()
- plot(x

Data Inputs and creation in R

Simulation Data in R

- set.seed(40); rnorm(n=2)
- set.seed(40); rnorm(n=3, mean=0, sd=1)
- set.seed(40); runif(n=4, min=0, max=1)
- set.seed(40); mb<- sample(x=11:15, size=3)
- mb
- wri<-data.frame(inc=1:5, year=2001:2005)
- wri
- set.seed(40); sam<- sample(x=1:nrow(wri), size=nrow(wri)-2)
- wri1<-wri[sam,]
- wri; sam; wri1

Reading External data in R

- BB <- read.csv(file="heisenberg.csv",head=TRUE,sep=",")
- dir()
- getwd()
- BB <- read.csv(file="heisenberg.csv",head=TRUE,sep=",")
- library(nonparaeff)
- data(heisenberg)
- attributes(heisenberg)
- is.data.frame(heisenberg)

Exporting data in R

- Tables can be saved with write,table() command. The write.table function allows you to export data to a wider range of file formats, including tab-delimited files. Use the sep argument to specify which character should be used to separate the values. To export a dataset to a tab-delimited file, set the sep argument to "\t" (which denotes the tab symbol), as shown below.
- write.table(mydata, "c:/mydata.txt", sep="\t")
- To save the file somewhere other than in the working directory, enter the full path for the file as shown.
- write.csv(dataset, "C:/folder/filename.csv")
- library(xlsx)
  write.xlsx(mydata, "c:/mydata.xlsx")
- export data frame to Stata binary format
  library(foreign)
  write.dta(mydata, "c:/mydata.dta")

- 3+5
- "+"(3,5)
- 3*5
- 3%%5
- aa<-3+c(5,6)
- bb<-"+"(3,c(5,6))*aa
- bb
- my.score<-95
- my.score

Maths in R

- x <- 1:8
- mean(x)
- y<- c(1,2,3,4,5,6,7,8)
- mean(y)
- y1<- c(1,2,3,4,5,6,7,8,NA)
- mean(y1)
- mean(y1,na.rm=TRUE)
- dog<-c(1,3,5,2^4,70,100%%8)
- pig<-c(1,2,6)+1
- cow<-70
- r1<-dog==pig; r2<-dog<cow
- r3<-r1 & r2;r4<-r1+r2

Numbers and expressions

Vectors

- x=c(1,2,3,4,5)
- x
- length(x)
- mode(x)
- names(x)
- x[2]
- x>10
- names <-c("A","B","C","D","E")
- names(x)<-names
- x
- x["A"]
- rep(NA,8)
- 1:100

Matrix

- B<-matrix<-rep(1:4,rep(3,4))
- dim(B)<-c(3,4)
- C<-seq(-2,2,length=25)
- C
- D<-rbind(c(1,2,-1),c(-3,1,5))
- D
- E<-cbind(B,C)
- A = matrix(c(2, 4, 3, 1, 5, 7), nrow=2,ncol=3,byrow = TRUE);A
- wq<- matrix((1:30),nrow=30,ncol=1, byrow=TRUE);wq
- wq<- matrix((1:30),nrow=30,ncol=100, byrow=TRUE);wq
- length(wq)
- dim(wq)
- mode(wq)
- dimnames(wq)

Arrays

- Aarray<-c(1:8, 11:18, 111:118);Aarray
- arr1<- array(c(2:9,12:19,112:119), dim=c(2,4,3))
- arr1
- arr1[,,2]
- arr1[1,,]
- arr1[1,,2]
- length(arr1)
- dim(arr1)
- mode(arr1)
- dimnames(arr1)

- iris[c(1:3,147:150), ,]
- names(iris)
- z<-iris$Sepal.Width
- z<-iris[[2]]z
- z
- c(mean=mean(z),st_dev=sd(z))
- table(iris$Species)
- attach(iris)
- x1<-Sepal.Length[1:50];x2=Sepal.Length[51:100];x3=Sepal.Length[101:150]
- summary(x1)
- summary(x2)
- summary(x3)
- myf<-sample(c(T,F), size=20, replace=T)
- myf
- myl<-rnorm(20)+runif(20)*1i
- myl
- mym<-matrix(rnorm(40),ncol=2)
- mym
- mydataframe<-data.frame(myf,myl,mym)
- mydataframe

Data Frames

```
cars <- c(1, 3, 6, 4, 9, 11,22,32,44,54,123,32,45,67,89,112)
plot(cars)
plot(cars, type="o", col="blue")
# Create a title with a red, bold/italic font
title(main="Autos", col.main="red", font.main=4)
# Define 2 vectors
cars <- c(1, 3, 6, 4, 9,18,22,32,34,54,43,56,65,11,12,23,45,67,112)
trucks <- c(2, 5, 4, 5, 12,32,34,32,35,34,56,76,65,45,45,64,43,23,112)
plot(cars, type="o", col="blue", ylim=c(0,250))
lines(trucks, type="o", pch=22, lty=2, col="red")
title(main="Autos", col.main="red", font.main=4)

##BoxPlot##
cars <- c(1, 3, 6, 4, 9,18,22,32,34,54,43,56,65,11,12,23,45,67,112)
trucks <- c(2, 5, 4, 5, 12,32,34,32,35,34,56,76,65,45,45,64,43,23,112)
barplot(cars)
barplot(trucks)
##Histograms##
cars <- c(1, 3, 6, 4, 9,18,22,32,34,54,43,56,65,11,12,23,45,67,112)
trucks <- c(2, 5, 4, 5, 12,32,34,32,35,34,56,76,65,45,45,64,43,23,112)
hist(cars, col="lightblue", ylim=c(0,120))
max_num <- max(cars)
hist(cars, col=heat.colors(max_num), breaks=max_num,
   xlim=c(0,max_num), right=F, main="Autos Histogram", las=1)
```

Plotting in R

From Erer library download the daLaw archive. First explore this file. Second, the first column of daLaw[,"Y"] has the mode of numeric. Please converted into a factor mode. Third, the labels of the four levels need to be strict liability for the value of 0, uncertain liability for the value of 1, simple negligence for 2 and gross negligence for 3. The factor needs to be ordered. Save the new data frame as Law1. Fourth, sort daLaw by the column of Y and STATE and save the data as Law2. Fifht, extract a subset and save it as Law3 (with the condition of value Y is 2 and the value of FYNIP >15).Finally, merge the Law3 and Law2 files and Law1 with Law2.

Things to do I

Things to do II

Create the two matrices $A = \begin{bmatrix} 10 & 98 \\ 24 & 30 \end{bmatrix}, B = \begin{bmatrix} 5 & 33 \\ 14 & 28 \end{bmatrix}$.

Please calculate the addition, subtraction, multiplications and division. Put the A matrix before the arithmetic operator. Finally, calculate the inversion, determinant, trace, transpose and ranks of matrix A and B.

Helpful Resources

Fox, J. (2005). R commander: A basic-statistics user interface to R. *Journal of Statistical Software*. 14, (9), 1-42.

Teetor, P. (2011). *25 Recipes for Getting Started with R.* Sebastopol, CA: O'Reilly Media Inc.

Teetor, P. (2011). *R cookbook.* Sebastopol, CA: O'Reilly Media Inc.

Crowley, M. J. (2007). *The R Book.* Chichester, New England: John Wiley & Sons, Ltd.

https://www.youtube.com/watch?v=9f2g7RN5N0I

https://stat.ethz.ch/mailman/listinfo/r-help