

Installing R and the *psych* package

William Revelle
Department of Psychology
Northwestern University

August 17, 2014

Contents

1	Overview of this and related documents	2
2	Install R and relevant packages	2
2.1	Install R for the first time	3
2.1.1	Install R	4
2.1.2	For a PC: (Figure 2)	5
2.1.3	For a Mac: download and install the appropriate version – Mac (Figure 3)	5
2.2	Starting R	5
2.2.1	Startup for a Mac	8
2.2.2	Install relevant packages	8
3	Reading in the data for analysis	9
3.1	Copy the data from another program using the copy and paste commands of your operating system	10
3.2	Import from an SPSS or SAS file	11
4	Some simple descriptive statistics before you start	12

1 Overview of this and related documents

To do basic and advanced personality and psychological research using R is not as complicated as some think. This is one of a set of “How To” do various things using R ([R Core Team, 2014](#)), particularly using the *psych* ([Revelle, 2014](#)) package.

The current list of How To’s includes:

1. Installing R and some useful packages (this document)
2. Using R and the *psych* package to [find \$\omega\$](#) .
3. [Using the `scoreItems` function](#) to find scale scores and scale statistics.
4. Using R and the *psych* package to do [factor analysis](#).
5. An [overview](#) (vignette) of the *psych* package.

By following these simple guides, you soon will be able to do such things as find ω_h by issuing just three lines of code:

- `library(psych)`
- `my.data <- read.clipboard()`
- `omega(my.data)`

The resulting output will be both graphical and textual.

This guide helps the naive R user to issue those three lines.

2 Install R and relevant packages

To use R obviously requires installing R on your computer. This is very easy to do (see section [2.1.1](#)) and needs to be done once.

The power of R is in the supplemental *packages*. At a minimum, you will need to install three package (*psych* ([Revelle, 2014](#)), *GPArotation* ([Bernaards and Jennrich, 2005](#)) and *MASS* ([Venables and Ripley, 2002](#)). With these three packages, you will be able to find ω_h using Exploratory Factor Analysis. If you want to find it using Confirmatory Factor Analysis, you will also need to add the *sem* ([Fox et al., 2013](#)) and *matrixcalc* ([Novomestky, 2012](#)) packages.

Another powerful package for doing structural equation modeling is the *lavaan* ([Rosseel, 2012](#)). For a more complete installation of a number of psychometric packages, you can

install and activate a package (*ctv*) that installs a large set of psychometrically relevant packages. As is true for R, you will need to install packages just once.

2.1 Install R for the first time

1. Download from R Cran (<http://cran.r-project.org/>) (see section 2.1.1)
 - Choose appropriate operating system and download compiled R. Use the appropriate installer for your system.
2. Install R (current version is 3.1.1)
3. Start R
4. Add useful packages (just need to do this once) (see section 2.2.2)
 - (a) `install.packages(c("psych", "GPArotation", "MASS"))` #the minimum requirement for using the *psych* package.
 - (b) If you want to use the CFA estimates for *omegaSem* using the *sem* package from John Fox et al. (2012) you will need to install it also.
`install.packages("sem", dependencies=TRUE)`
 - (c) For structural equation modeling using the *lavaan* package (Rosseel, 2012) also install it: `install.packages("lavaan", dependencies=TRUE)`
5. Another option is install a large set of packages, all of which are meant to do psychometrics. To do this, you take advantage of the *task views* package.
 - (a) `install.packages("ctv")` #this downloads the task view package
 - (b) `library(ctv)` #this activates the *ctv* package
 - (c) `install.views("Psychometrics")` #among others
 - (d) Take a 5 minute break (this will take a while).
6. Once you have installed the necessary packages, you must activate the package(s) you want to use (e.g., *psych*) in a particular session. Although you need to install packages just once, you do need to make them active every time you start a new session.
 - `library(psych)` #necessary for finding omega
psych will automatically activate the other packages it needs, as long as they are installed. Note that *psych* is updated roughly quarterly, the current version is 1.4.8.

7. Use R

2.1.1 Install R

First go to the Comprehensive R Archive Network (CRAN) at <http://cran.r-project.org>: (Figure 1)

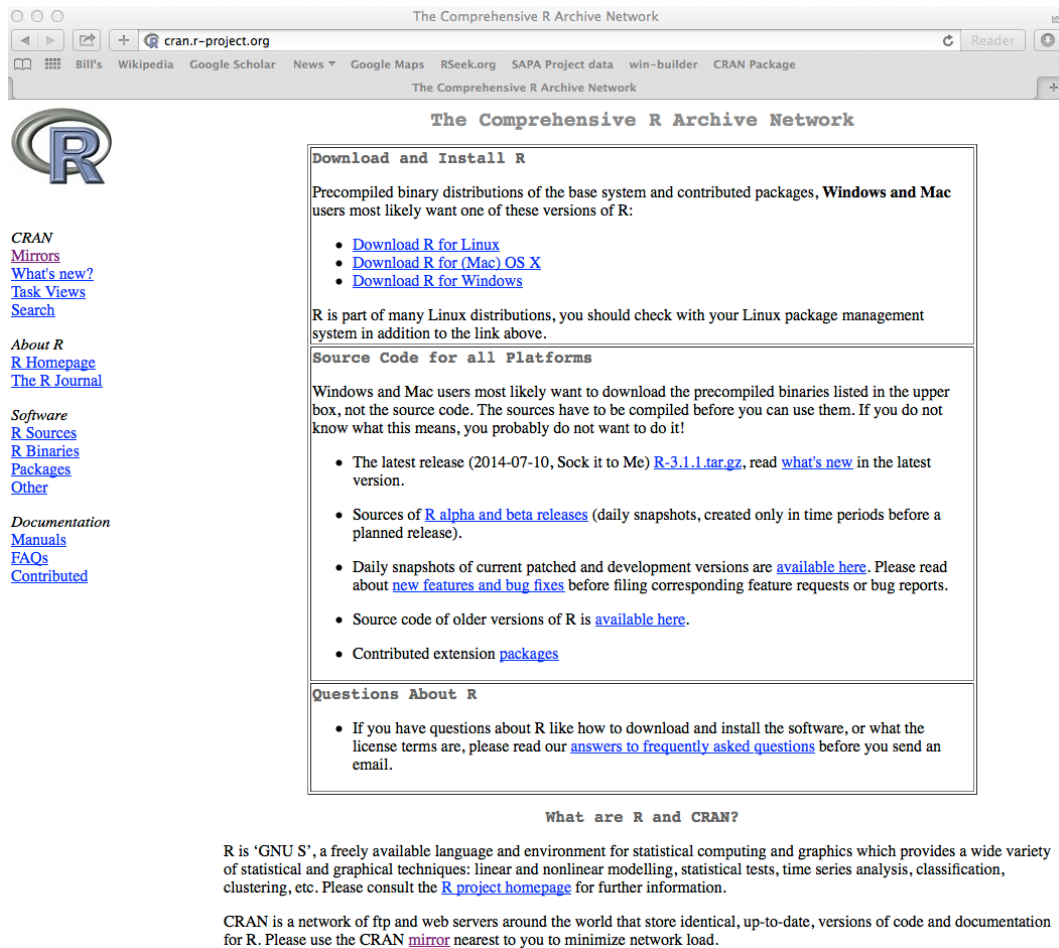


Figure 1: The basic CRAN window allows you choose your operating system. Comprehensive R Archive Network (CRAN) is found at <http://cran.r-project.org>:

Choose your operating system and then download and install the appropriate version

2.1.2 For a PC: (Figure 2)

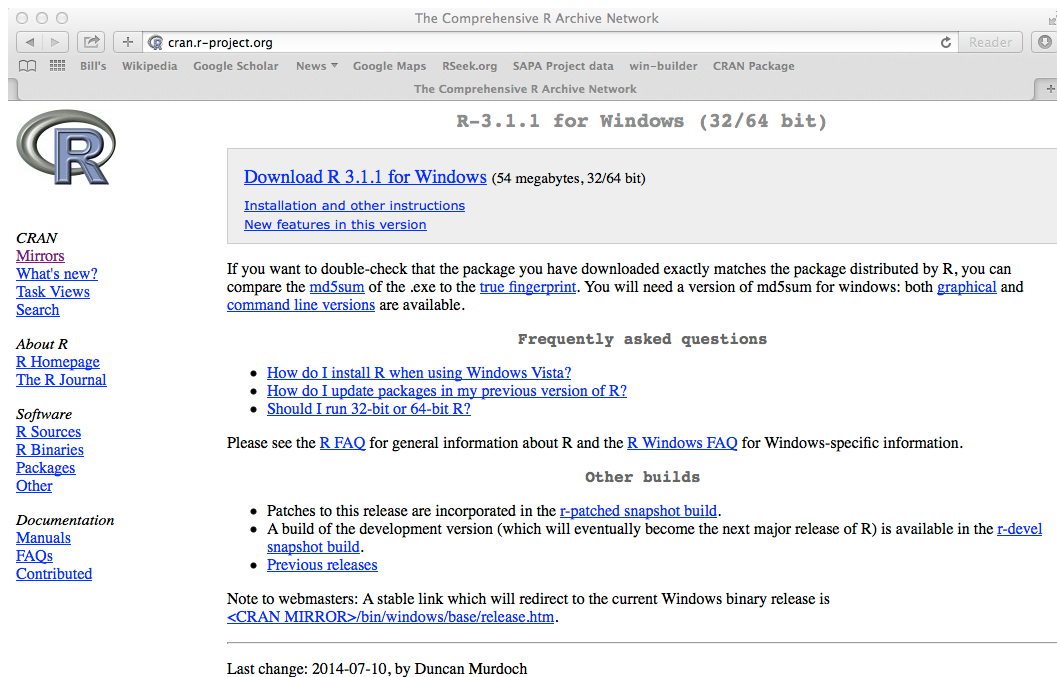
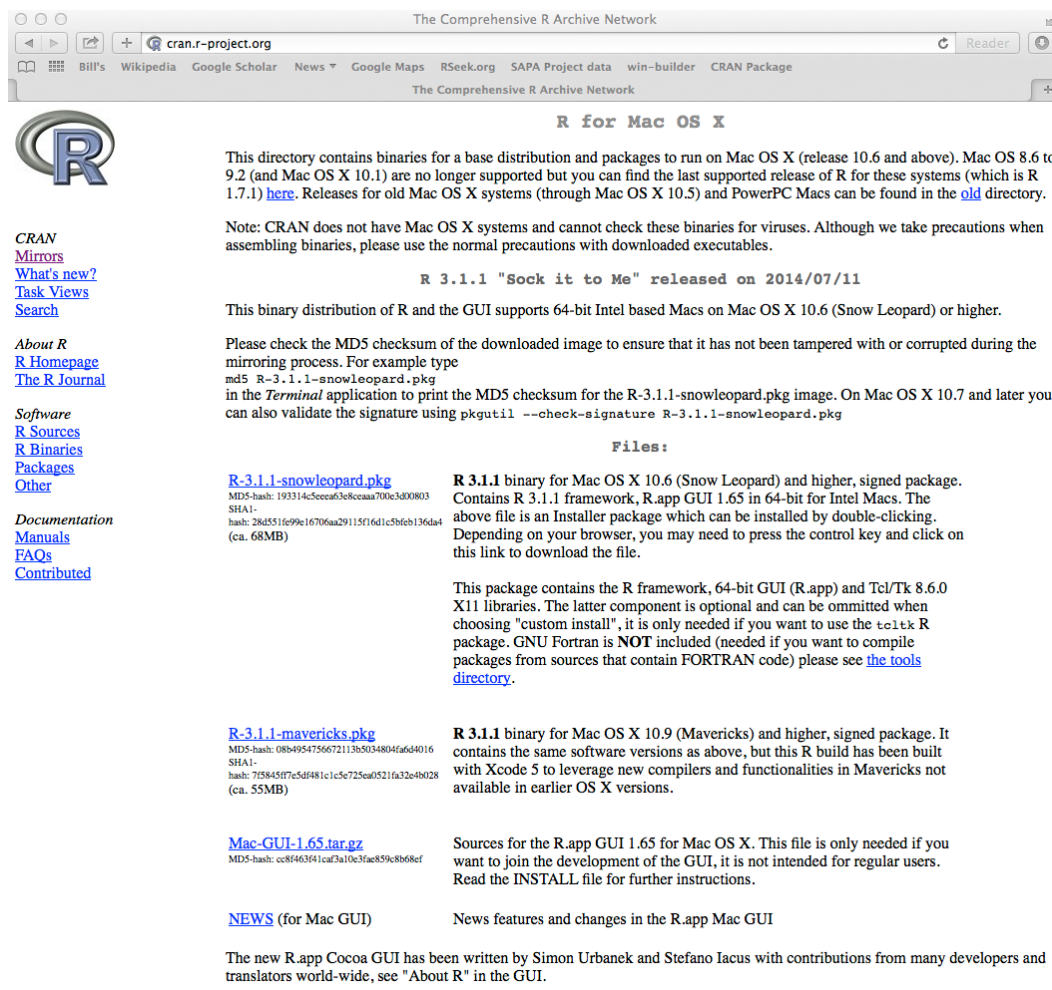


Figure 2: On a PC you want to choose the base system. This will then download an installer.

2.1.3 For a Mac: download and install the appropriate version – Mac (Figure 3)

2.2 Starting R

To start R with either a PC or a Mac, just click on the appropriate icon and you will be shown a welcome screen (Figure 4 for a PC, Section 2.2.1 for a Mac).



The Comprehensive R Archive Network

cran.r-project.org

Bill's Wikipedia Google Scholar News Google Maps RSeek.org SAPA Project data win-builder CRAN Package

The Comprehensive R Archive Network

R for Mac OS X

This directory contains binaries for a base distribution and packages to run on Mac OS X (release 10.6 and above). Mac OS 8.6 to 9.2 (and Mac OS X 10.1) are no longer supported but you can find the last supported release of R for these systems (which is R 1.7.1) [here](#). Releases for old Mac OS X systems (through Mac OS X 10.5) and PowerPC Macs can be found in the [old](#) directory.

Note: CRAN does not have Mac OS X systems and cannot check these binaries for viruses. Although we take precautions when assembling binaries, please use the normal precautions with downloaded executables.

R 3.1.1 "Sock it to Me" released on 2014/07/11

This binary distribution of R and the GUI supports 64-bit Intel based Macs on Mac OS X 10.6 (Snow Leopard) or higher.

Please check the MD5 checksum of the downloaded image to ensure that it has not been tampered with or corrupted during the mirroring process. For example type

```
md5 R-3.1.1-snowleopard.pkg
```

in the *Terminal* application to print the MD5 checksum for the R-3.1.1-snowleopard.pkg image. On Mac OS X 10.7 and later you can also validate the signature using `pkgutil --check-signature R-3.1.1-snowleopard.pkg`

Files:

[R-3.1.1-snowleopard.pkg](#)
MD5-hash: 193314c5eeea3e8ccaa700c3d00803
SHA1-
hash: 26d551e99e16706aa29115f16d1c5bfeb136da4
(ca. 68MB)

R 3.1.1 binary for Mac OS X 10.6 (Snow Leopard) and higher, signed package. Contains R 3.1.1 framework, R.app GUI 1.65 in 64-bit for Intel Macs. The above file is an Installer package which can be installed by double-clicking. Depending on your browser, you may need to press the control key and click on this link to download the file.

This package contains the R framework, 64-bit GUI (R.app) and Tcl/Tk 8.6.0 X11 libraries. The latter component is optional and can be omitted when choosing "custom install", it is only needed if you want to use the `tcltk` R package. GNU Fortran is **NOT** included (needed if you want to compile packages from sources that contain FORTRAN code) please see [the tools directory](#).

[R-3.1.1-mavericks.pkg](#)
MD5-hash: 08b4954756672113b5034804fa64016
SHA1-
hash: 7f584507e5d481c1e56725ea0521fa32e4b028
(ca. 55MB)

R 3.1.1 binary for Mac OS X 10.9 (Mavericks) and higher, signed package. It contains the same software versions as above, but this R build has been built with Xcode 5 to leverage new compilers and functionalities in Mavericks not available in earlier OS X versions.

[Mac-GUI-1.65.tar.gz](#)
MD5-hash: cc8f463f41ca73a10c3fae859c8b68ef

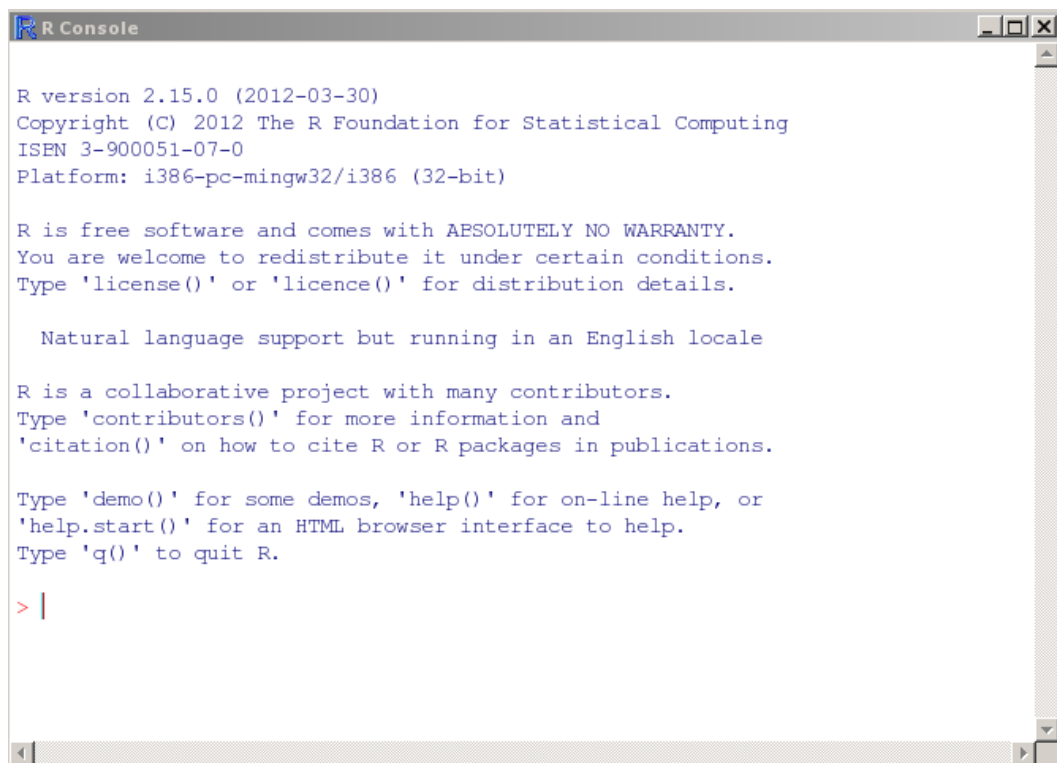
Sources for the R.app GUI 1.65 for Mac OS X. This file is only needed if you want to join the development of the GUI, it is not intended for regular users. Read the INSTALL file for further instructions.

[NEWS](#) (for Mac GUI)

News features and changes in the R.app Mac GUI

The new R.app Cocoa GUI has been written by Simon Urbanek and Stefano Iacus with contributions from many developers and translators world-wide, see "About R" in the GUI.

Figure 3: For the Mac, you want to choose the latest version which includes a very helpful GUI.



```
R Console

R version 2.15.0 (2012-03-30)
Copyright (C) 2012 The R Foundation for Statistical Computing
ISEN 3-900051-07-0
Platform: i386-pc-mingw32/i386 (32-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, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

Figure 4: The startup screen on a PC

2.2.1 Startup for a Mac

Start up R and get ready to play (Mac version). (Note that this is the development version. The officially released version is 3.1.1.)

```
R Under development (unstable) (2014-08-10 r66331) — "Unsuffered_Consequences"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.1.0 (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, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

```
[R.app GUI 1.66 (6793) x86_64-apple-darwin13.1.0]
```

```
[Workspace restored from /Users/revelle/.RData]
[History restored from /Users/revelle/.Rapp.history]
```

To find out what packages are active in your release ask for `sessionInfo()`

R code

```
> sessionInfo()
```

```
R Under development (unstable) (2014-08-10 r66331)
Platform: x86_64-apple-darwin13.1.0 (64-bit)
```

```
locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
```

```
attached base packages:
[1] stats      graphics  grDevices  utils      datasets  methods   base
```

```
other attached packages:
[1] psych_1.4.8.11
```

```
> #> is the prompt for all commands #is for comments
```

2.2.2 Install relevant packages

Once R is installed on your machine, you still need to install a few relevant “packages”. Packages are what make R so powerful, for they are special sets of functions that are

designed for one particular application. In the case of the *psych* package, that application is for doing the kind of basic data analysis and psychometric analysis that personality psychologists find particularly useful.

You may either install the minimum set of packages necessary to do the analysis using an EFA approach (recommended) or a few more packages to do both the EFA and a CFA approach.

Perhaps the most useful package to add is *lavaan* by [Rosseel \(2012\)](#).

It is also possible to add many psychometrically relevant packages all at once by using the “task views” approach.

Install the minimum set

- `install.packages(c("psych", "GPArotation", "MASS"))`

Install the *sem* package if you want to do omegaSem. Install the *lavaan* package to do structural equation modeling.

- `install.packages(c("psych", "GPArotation", "MASS", "sem", "matrixcalc"))`
- For structural equation modeling using the *lavaan* package ([Rosseel, 2012](#)) also install it: `install.packages("lavaan", dependencies=TRUE)`

Install all the psychometric packages from the “psychometrics” task view.

- `install.packages("ctv")` #this downloads the task view package
- `library(ctv)` #this activates the ctv package
- `install.views("Psychometrics")` #among others
- Take a 5 minute break

You are almost ready. But first you need to make the *psych* package active. You only need to do this once per session.

- `library(psych)` #necessary for finding omega

3 Reading in the data for analysis

There are of course many ways to enter data into R. Reading from a local file using `read.table` is perhaps the most preferred. You first need to find the file and then read it.

This can be done with the `file.choose` and `read.table` functions:

R code

```
file.name <- file.choose()
my.data <- read.table(file.name)
```

`file.choose` opens a search window on your system just like any open file command does. It doesn't actually read the file, it just finds the file. The read command is also necessary.

3.1 Copy the data from another program using the copy and paste commands of your operating system

However, many users will enter their data in a text editor or spreadsheet program and then want to copy and paste into R. This may be done by using `read.table` and specifying the input file as "clipboard" (PCs) or "pipe(pbpaste)" (Macs). Alternatively, the `read.clipboard` set of functions are perhaps more user friendly:

`read.clipboard` is the base function for reading data from the clipboard.

`read.clipboard.csv` for reading text that is comma delimited.

`read.clipboard.tab` for reading text that is tab delimited (e.g., copied directly from an Excel file).

`read.clipboard.lower` for reading input of a lower triangular matrix with or without a diagonal. The resulting object is a square matrix.

`read.clipboard.upper` for reading input of an upper triangular matrix.

`read.clipboard.fwf` for reading in fixed width fields (some very old data sets)

For example, given a data set copied to the clipboard from a spreadsheet, just enter the command

R code

```
> my.data <- read.clipboard()
```

This will work if every data field has a value and even missing data are given some values (e.g., NA or -999). If the data were entered in a spreadsheet and the missing values were just empty cells, then the data should be read in as a tab delimited or by using the `read.clipboard.tab` function.

R code

```
> my.data <- read.clipboard(sep="\t")    #define the tab option, or  
> my.tab.data <- read.clipboard.tab()    #just use the alternative function
```

For the case of data in fixed width fields (some old data sets tend to have this format), copy to the clipboard and then specify the width of each field (in the example below, the first variable is 5 columns, the second is 2 columns, the next 5 are 1 column the last 4 are 3 columns).

R code

```
> my.data <- read.clipboard.fwf(widths=c(5,2,rep(1,5),rep(3,4)))
```

3.2 Import from an SPSS or SAS file

To read data from an SPSS, SAS, or Systat file, you must use the *foreign* package. This might need to be installed (see [2.2.2](#) for instructions for installing packages).

`read.spss` reads a file stored by the SPSS save or export commands.

```
read.spss(file, use.value.labels = TRUE, to.data.frame = FALSE,  
          max.value.labels = Inf, trim.factor.names = FALSE,  
          trim_values = TRUE, reencode = NA, use.missings = to.data.frame)
```

file Character string: the name of the file or URL to read.

use.value.labels Convert variables with value labels into R factors with those levels?

to.data.frame return a data frame? Defaults to FALSE, probably should be TRUE in most cases.

max.value.labels Only variables with value labels and at most this many unique values will be converted to factors if `use.value.labels = TRUE`.

trim.factor.names Logical: trim trailing spaces from factor levels?

trim_values logical: should values and value labels have trailing spaces ignored when matching for `use.value.labels = TRUE`?

use.missings logical: should information on user-defined missing values be used to set the corresponding values to NA?

Unfortunately, these default values do not seem optimal. An example of reading from an SPSS file and then describing the data set to make it looks ok. It is better to not use `value.labels` and to transfer to a `data.frame`. The following data set is from Eli Finkel and was saved as an `SPSS.sav` file.

R code

```
> library(foreign)
> datafilename <- "http://personality-project.org/r/datasets/finkel.sav"
> eli <- read.spss(datafilename,to.data.frame=TRUE,
                  use.value.labels=FALSE)

> describe(eli,skew=FALSE)
```

The output of the `describe` function shows that we read the data correctly. Note that we read this from a remote server!

	var	n	mean	sd	median	trimmed	mad	min	max	range	se
USER*	1	69	35.00	20.06	35	35.00	25.20	1	69	68	2.42
HAPPY	2	69	5.71	1.04	6	5.82	0.00	2	7	5	0.13
SOULMATE	3	69	5.09	1.80	5	5.32	1.48	1	7	6	0.22
ENJOYDEX	4	68	6.47	1.01	7	6.70	0.00	2	7	5	0.12
UPSET	5	69	0.41	0.49	0	0.39	0.00	0	1	1	0.06

4 Some simple descriptive statistics before you start

Although you probably want to jump right in and find ω , or do a factor analysis using `fa` you should first make sure that your data are reasonable. Use the `describe` function to get some basic descriptive statistics. This next example takes advantage of a built in data set (`sat.act`) in the *psych* package. There is a fairly clear outlier on the ACT data.

```
my.data <- sat.act #built in example -- replace with your data
pairs.panels(sat.act,pch='.')
describe(my.data)
```

	var	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis
se												
gender	1	700	1.65	0.48	2	1.68	0.00	1	2	1	-0.61	-1.62 0.02
education	2	700	3.16	1.43	3	3.31	1.48	0	5	5	-0.68	-0.07 0.05
age	3	700	25.59	9.50	22	23.86	5.93	13	65	52	1.64	2.42 0.36
ACT	4	700	28.55	4.82	29	28.84	4.45	3	36	33	-0.66	0.53 0.18
SATV	5	700	612.23	112.90	620	619.45	118.61	200	800	600	-0.64	0.33 4.27
SATQ	6	687	610.22	115.64	620	617.25	118.61	200	800	600	-0.59	-0.02 4.41

In addition to simple descriptives, it is always helpful to graphically examine your data using the `pairs.panels` function.

Looking at that scatter plot, there seems to be one very clear outlier. However, by using the `outlier` function which finds the squared Mahalanobis distance

$$D^2 = (x - \mu)' \Sigma^{-1} (x - \mu)$$

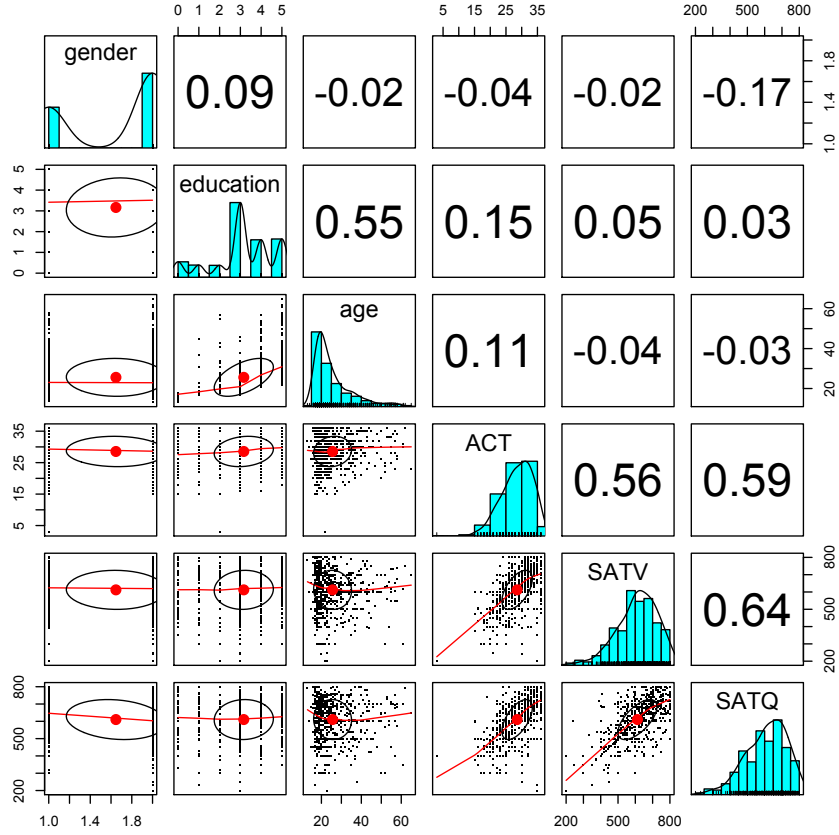


Figure 5: A scatter plot matrix (splom) of a data set is a powerful way to examine a data set. Elements on the diagonal show the histograms and densities of the data, lower off diagonal elements are the pairwise scatter plots, upper off diagonal elements are the pairwise correlations.

where Σ is the covariance of the x matrix and then plotting the Quantile x Quantile plot of D^2 by χ^2 for that number of degrees of freedom (Figure 6), we can recognize at 5 multivariate outlets. We can show this by using the `pairs.panels` function with blue representing the most extreme values (Figure 7).

R code

```
#first, just find and graph the outliers
d2 <- outlier(sat.act,cex=.8)
#combine with the data frame and plot it with the outliers highlighted in blue
sat.d2 <- data.frame(sat.act,d2)
pairs.panels(sat.d2,bg=c("yellow","blue")[(d2 > 25)+1],pch=21)
```

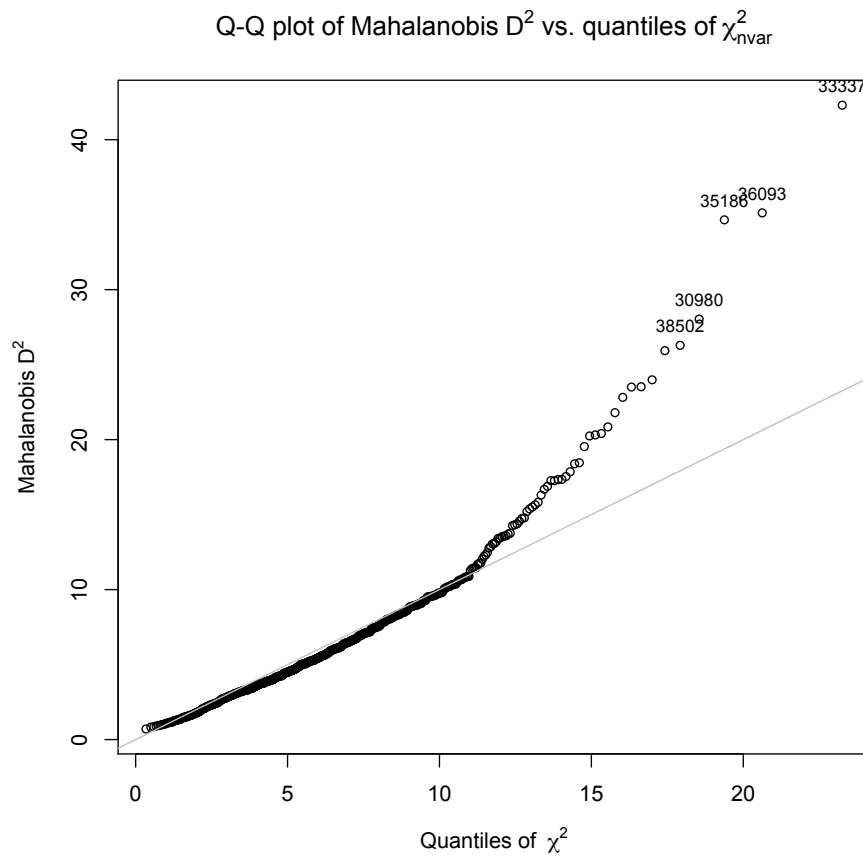


Figure 6: The squared Mahalanobis distance should be distributed as χ^2 except for outliers.

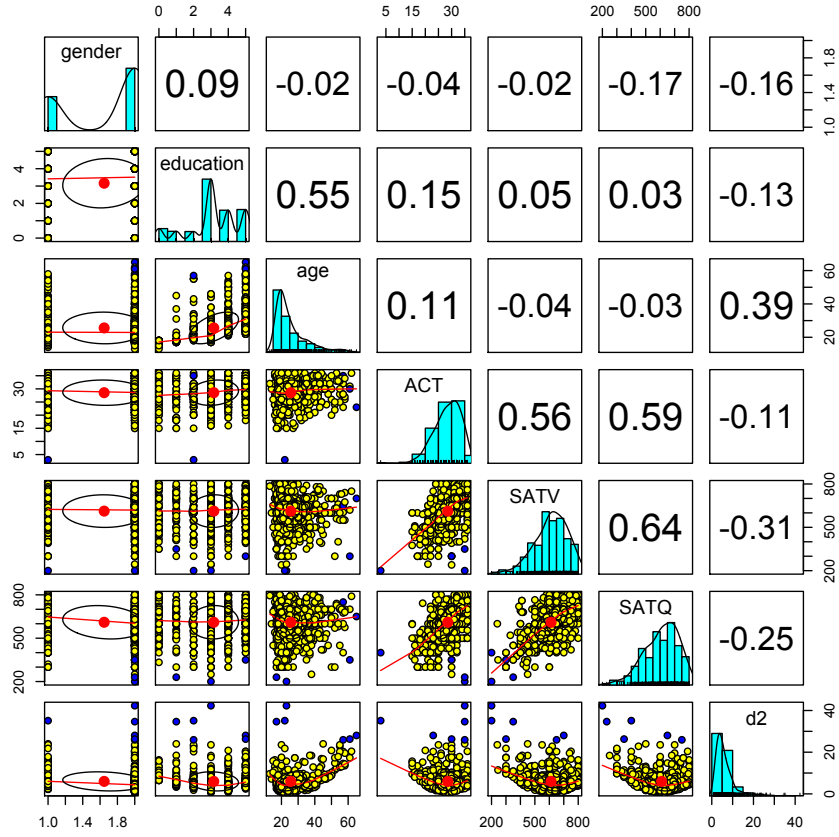


Figure 7: A SPLOM of the sat.act data highlighting those cases with extreme values of d^2 .

There are, of course, all kinds of things you could do with your data at this point, and you may read about some of what is available in the *psych* package in the accompanying vignette for the *psych* package <http://cran.r-project.org/web/packages/psych/vignettes/overview.pdf>. (On a Mac, a list of vignettes for the packages installed on your system may be found under the Help Menu).

References

- Bernaards, C. and Jennrich, R. (2005). Gradient projection algorithms and software for arbitrary rotation criteria in factor analysis. *Educational and Psychological Measurement*, 65(5):676.
- Fox, J., Nie, Z., and Byrnes, J. (2012). *sem: Structural Equation Models*.
- Fox, J., Nie, Z., and Byrnes, J. (2013). *sem: Structural Equation Models*. R package version 3.1-0.
- Novomestky, F. (2012). *matrixcalc: Collection of functions for matrix calculations*. R package version 1.0-3.
- R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.
- Revelle, W. (2014). *psych: Procedures for Personality and Psychological Research*. Northwestern University, Evanston, <http://cran.r-project.org/web/packages/psych/>. R package version 1.4.2.
- Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, 48(2):1–36.
- Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition. ISBN 0-387-95457-0.