R: a continued short course

William Revelle

Northwestern University
Evanston, Illinois USA
http://personality-project.org/r/book/Smep_R_course.pdf

October, 2010

Outline

Getting Started with R

- Download from R Cran (`http://cran.r-project.org/`)
  - Choose appropriate operating system and download compiled R
- Install R (Version 2.11.1 or 2.12.alpha)
- Start R
- Add useful packages
  - install.packages("ctv") #this downloads the task view package
  - library("ctv") #this activates the package
  - install.views("Psychometrics") #among others
  - Take a 5 minute break
- Activate the package(s) you want to use today (e.g.,)
  - library(psych) #necessary for most of today's examples
  - library(sem) #will be used for a few examples
- Use R

Start up R and get ready to play

```
R version 2.12.0 alpha (2010-09-22 r52970)
Copyright (C) 2010 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: i386-apple-darwin9.8.0/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.
[R.app GUI 1.34 (5584) i386-apple-darwin9.8.0]

[Workspace restored from /Volumes/WR/bill/.RData]
>    # >  is the prompt for all commands    #is for comments
```

Annotated installation guide

```
> install.packages("ctv")

> library(ctv)

> install.views("Psychometrics")
> #or
> install.packages("psych")
....
> library(psych)

> library(sem)
```

- Install the task view installer package
- Make it active
- Install all the packages in the "Psychometrics" task view
- Or, just install one package (e.g., psych)
- Make the psych package active (do this at every startup)
- Make the sem package active (will be automatically done from psych if needed)

Check the version number for R (should be $\geq$ 2.11) and for psych ($\geq$1.0-.92)

```
> sessionInfo()

R version 2.12.0 alpha (2010-09-22 r52970)
Platform: i386-apple-darwin9.8.0/i386 (32-bit)

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

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

other attached packages:
[1] ctv_0.6-0       GPArotation_2010.07-1 MASS_7.3-8      psych_1.0-93

loaded via a namespace (and not attached):
[1] tools_2.12.0
```

Basic R commands

R is just a fancy calculator. Add, subtract, sum, products, group

```
> 2 + 2  #addition
```

```
[1] 4
```

```
> 3 ^ 4 #exponentiation
```

```
[1] 81
```

```
> sum(1:10)   #summation of a range of numbers
```

```
[1] 55
```

```
> prod(c(1,2,3,5,7))   #products of a set of numbers
```

```
[1] 210
```

It is also a statistics table (the normal distribution, the t distribution)

```
>  pnorm(1)
```

```
[1] 0.8413447
```

```
>  pt(2,20)   # probability of a t of 2 with 20 observations
```

```
[1] 0.9703672
```

R is a set of distributions

Table: To obtain the density, prefix with *d*, probability with *p*, quantiles with *q* and to generate random values with *r*. (e.g., the normal distribution may be chosen by using dnorm, pnorm, qnorm, or rnorm.)

Distribution	base name	P 1	P 2	P 3	example applica
Normal	norm	mean	sigma		Most data
Multivariate normal	mvnorm	mean	r	sigma	Most data
Log Normal	lnorm	log mean	log sigma		income or reactio
Uniform	unif	min	max		rectangular distrib
Binomial	binom	size	prob		Bernuilli trials (e.g.
Student's t	t	df		nc	Finding significance
Multivariate t	mvt	df	corr	nc	Multivariate appli
Fisher's F	f	df1	df2	nc	Testing for significanc
χ^2	chisq	df		nc	Testing for significa
Exponential	exp	rate			Exponential de
Gamma	gamma	shape	rate	scale	distribution the
Hypergeometric	hyper	m	n	k	
Logistic	logis	location	scale		Item Response T
Poisson	pois	lambda			Count data
Weibull	weibull	shape	scale		Reaction time distr

R does descriptive statistics

Examples can use built in data sets, type data() to list all of them, or data(package="psych") to see just the ones in the psych package.

```
> library(psych)     #do this once per session to make the package available
> data(bfi)             #get a particular data set
> headtail(describe(bfi),8,6)  #show the first and last n lines of the output from describe()
```

	var	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
A1	1	2784	2.41	1.41	2	2.23	1.48	1	6	5	0.83	-0.31	0.03
A2	2	2773	4.8	1.17	5	4.98	1.48	1	6	5	-1.12	1.05	0.02
A3	3	2774	4.6	1.3	5	4.79	1.48	1	6	5	-1	0.44	0.02
A4	4	2781	4.7	1.48	5	4.93	1.48	1	6	5	-1.03	0.04	0.03
A5	5	2784	4.56	1.26	5	4.71	1.48	1	6	5	-0.85	0.16	0.02
C1	6	2779	4.5	1.24	5	4.64	1.48	1	6	5	-0.85	0.3	0.02
C2	7	2776	4.37	1.32	5	4.5	1.48	1	6	5	-0.74	-0.14	0.03
C3	8	2780	4.3	1.29	5	4.42	1.48	1	6	5	-0.69	-0.13	0.02
...	...	...	...	...	...	...	...	...	...	...	...	...	...
O3	23	2772	4.44	1.22	5	4.56	1.48	1	6	5	-0.77	0.3	0.02
O4	24	2786	4.89	1.22	5	5.1	1.48	1	6	5	-1.22	1.08	0.02
O5	25	2780	2.49	1.33	2	2.34	1.48	1	6	5	0.74	-0.24	0.03
gender	26	2800	1.67	0.47	2	1.71	0	1	2	1	-0.73	-1.47	0.01
education	27	2577	3.19	1.11	3	3.22	1.48	1	5	4	-0.05	-0.32	0.02
age	28	2800	28.78	11.13	26	27.43	10.38	3	86	83	1.02	0.56	0.21

```
>
```

bfi has 2800 cases on 25 IPIP/SAPA items + 3 demographic items

R is also a matrix calculator

Calculate covariances from scratch, compare with a built in function

```
> data(sat.act)
> D <- sat.act[2:5]
> M <- colMeans(D)
> C <- t(t(D) - M)
> Cov  <- t(C) %*% C/(dim(D)[1]-1)
> round(Cov,digits=2)
```

	education	age	ACT	SATV
education	2.03	7.42	1.06	7.48
age	7.42	90.22	5.06	-45.42
ACT	1.06	5.06	23.27	305.55
SATV	7.48	-45.42	305.55	12746.99

```
> print(cov(D),digits=2)
```

	education	age	ACT	SATV
education	2.0	7.4	1.1	7.5
age	7.4	90.2	5.1	-45.4
ACT	1.1	5.1	23.3	305.5
SATV	7.5	-45.4	305.5	12747.0

More on matrices: finding correlations

Find the correlation from scratch, find it from formula

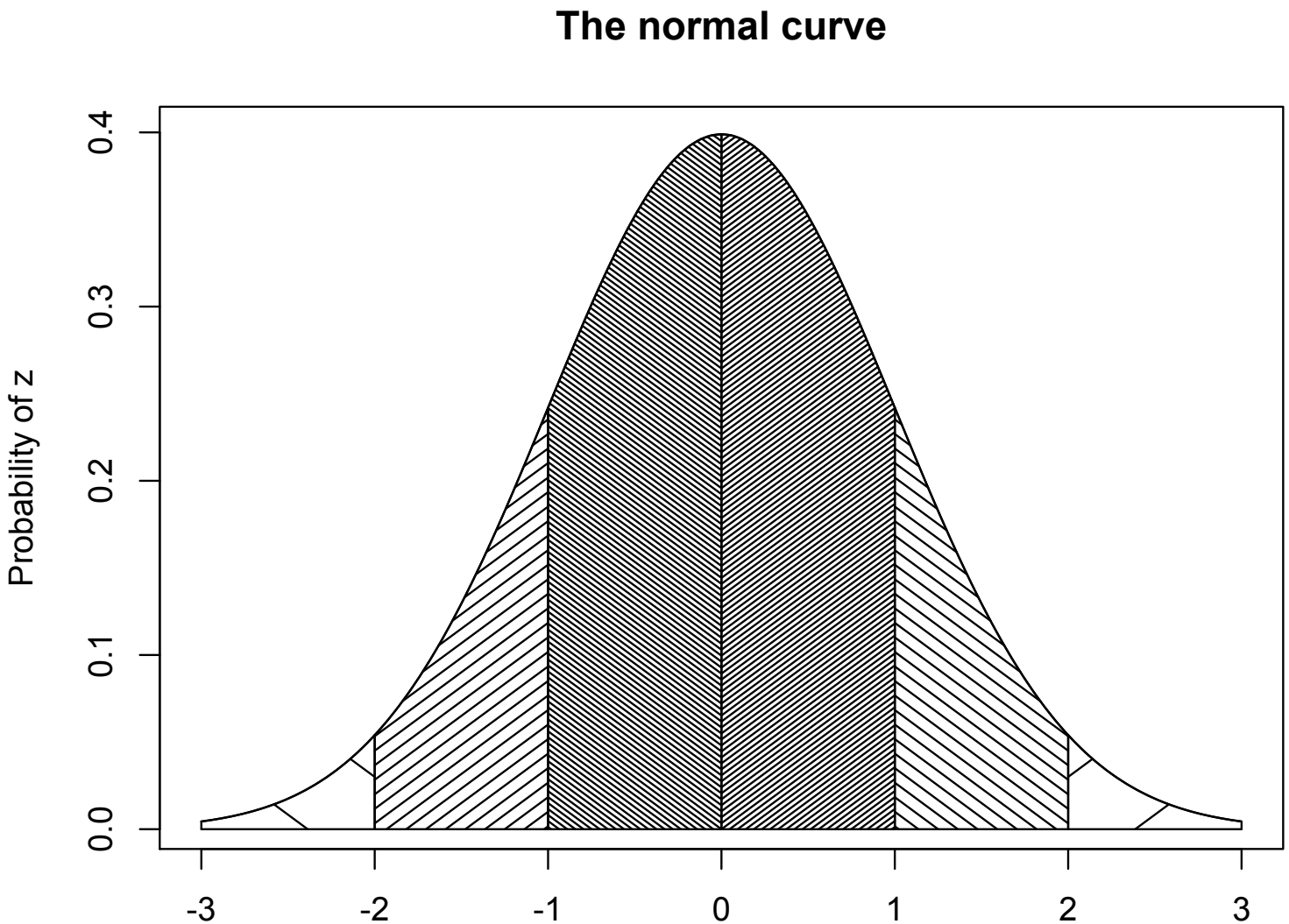
```
> M <- colMeans(D)
> C <- t(t(D) - M)
> Cov  <- t(C) %*% C/(dim(D)[1]-1)
> Var <- diag(Cov)
>  r <- diag(1/sqrt(Var)) %*% Cov %*% diag(1/sqrt(Var))
>  round(r,2)
```

```
     [,1]  [,2] [,3]  [,4]
[1,] 1.00  0.55 0.15  0.05
[2,] 0.55  1.00 0.11 -0.04
[3,] 0.15  0.11 1.00  0.56
[4,] 0.05 -0.04 0.56  1.00
```

```
>  r <- cor(D)
>  round(r,2)
```

```
          education   age  ACT  SATV
education      1.00  0.55 0.15  0.05
age            0.55  1.00 0.11 -0.04
ACT            0.15  0.11 1.00  0.56
SATV           0.05 -0.04 0.56  1.00
```

R is also a graphics calculator



R is also a graphics calculator

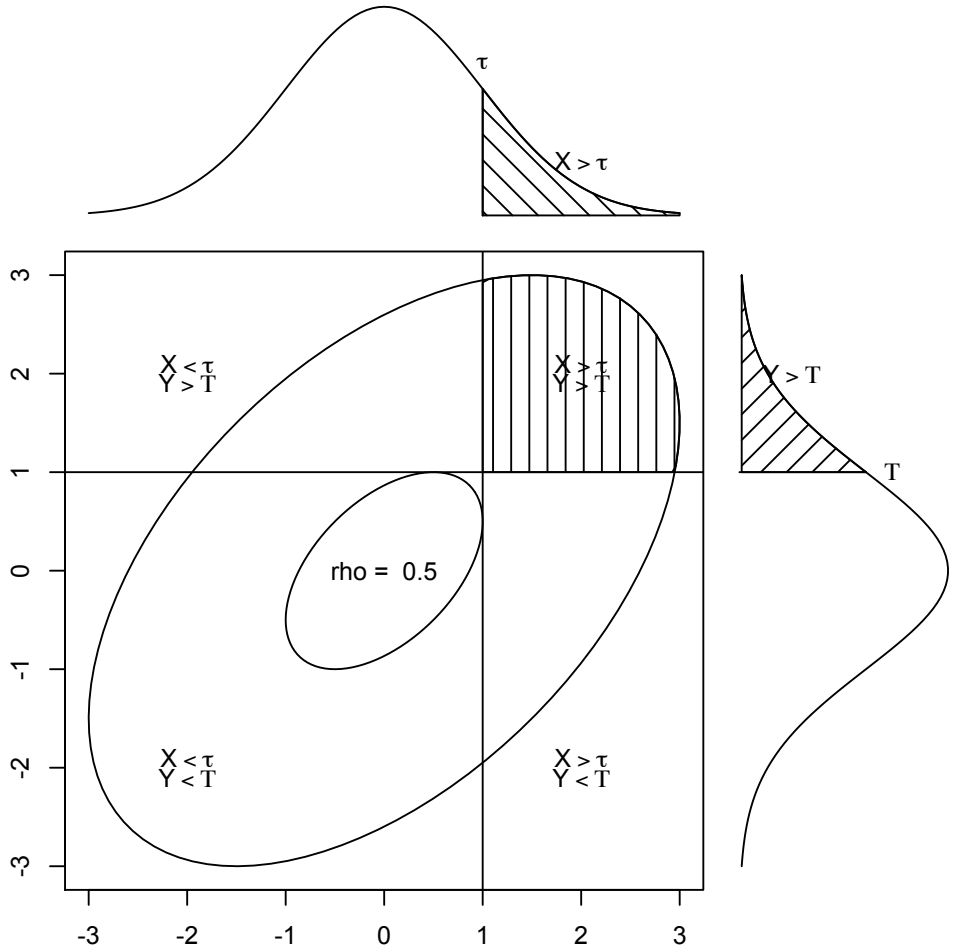
The first line draws the normal curve, the second prints the title, the next lines draw the cross hatching.

```
curve(dnorm(x),-3,3,xlab="",ylab="Probability of z")
title(main="The normal curve",outer=FALSE)
xvals <-  seq(-3,-2,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=2,angle=-45)
xvals <-  seq(-2,-1,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=14,angle=45)
xvals <-  seq(-1,-0,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=34,angle=-45)
xvals <-  seq(2,3,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=2,angle=45)
xvals <-  seq(1,2,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=14,angle=-45)
xvals <-  seq(0,1,length=100)
dvals <- dnorm(xvals)
polygon(c(xvals,rev(xvals)),c(rep(0,100),rev(dvals)),density=34,angle=45)
```

More graphics

It is possible to draw interesting figures, even with base graphics. The draw.tetra function is for showing students the relationship between a continuous bivariate normal and the dichotomous tetrachoric correlation.

> draw.tetra(.5,1,1)



Useful tools – help finding or using a function

- ? X #will open the help page for function X- but you need to know X!
- `help(X)` #same thing as ?
- ?? "X" #will list the help pages that include the term X.
- `help.search("X")` #same as ??
- `findFn("X")` #will search a web data base for all packages/functions that have "X", but needs to have the sos package active)
- ? packagename usually will give an overview of the package and the functions available in that package.
- R Package Manger (on the Mac menu) will display installed packages and allow you to find the index of the package.
- example(X) #Runs the examples from the help page for X
- Vignettes #available for many packages - a detailed description of how to use the package

R Basic concepts: Functions and Objects

```
>  q <- 1
>  p <-  pnorm(q,mean=0)
>  p

[1] 0.8413447

>  print(p,digits=2)

[1] 0.84

>  z <- seq(0,3,.5)
>  p <- pnorm(z)
>  zp <- data.frame(z,p)
>  round(zp,2)

    z    p
1 0.0 0.50
2 0.5 0.69
3 1.0 0.84
4 1.5 0.93
5 2.0 0.98
6 2.5 0.99
7 3.0 1.00
```

- All commands are functions
  - Usually with parameters
  - Sometime without parameters
  - Some functions are implicit
- Functions operate on objects
- Most functions return objects
  - visibily
  - invisibily (use str to see them)
- Objects can be a variety of data structures
  - scalers, vectors, matrices
  - data.frames
  - lists

The basic commands

- X <- some.function(Y, with parameters z1 ... zn)
- X (depending upon the function, might be a lot, might be the same as print(X)
- print(X) (What the writer of the function that produced X thought was important).
- summary(x) (What the writer of the function thought was really important).
- plot(X) (If the function produces suitable information for a graphic)
- See `http://cran.r-project.org/doc/contrib/Short-refcard.pdf` (a short reference card for many functions)

Useful tools for using objects

- Remember: the output of any function is an object.
  - What is shown on the screen might be a small subset of what is actually produced
  - The help page for the function should list the various outputs of the function (but sometimes not too clearly
- `str`(X) (shows the structure of an object, X)
  - The content is somewhat cryptic but does allow one to see what is available to use
- The object will probably have various objects attached to it, these can be retrieved by name, e.g.,
  - data(bifactor) #gets a data file including several different data sets (including Thurstone, Bechtoldt.1, etc.)
  - f3 <- fa(Thurstone,3,n.obs=213) #factor analyze the Thurstone 9 variable problem with 3 factors
  - str(f3) #show the structure (a list of 33 objects)
  - my.loadings <- f3$loadings #get the factor loadings matrix

Basic concepts of using functions

- Most functions are written in R
  - to use a function, just call it with parentheses, e.g.,
    - y <- fa(X,3) #to factor analyze X with three factors
  - to see how it works, just call it without the parentheses, e.g.,
    - fa #several screens of R
  - Some functions are "hidden" in namespaces but can be seen by calling them,
    - psych:::polyc
- Some functions are "primitives" and are written in C. These can be examined by finding the source file.
- Functions can be changed by listing them, editing them, and running them

Using packages

- More than 2552 packages are available for R (and growing daily)
- Can search all packages that do a particular operation by using the sos package
  - install.packages("sos") #if you haven't already
  - library(sos) # make it active once you have it
    - `findFn`("X") #will search a web data base for all packages/functions that have "X"
    - `findFn`("factor analysis") #will return 1907 matches and reports the top 400
    - `findFn`("Item Response Theory") # will return 114 matches
- install.packages(X) will install a particular package (add it to your R library – you need to do this just once)
- library(X) #will make the package X available to use if it has been installed (and thus in your library)

A small subset of very useful packages

- General use
  - lattice
  - lme4 (core)
  - MASS
  - psych
  - Zelig
- Special use
  - ltm
  - sem
  - OpenMx
  - GPArotation
  - mvtnorm
  - $>$ 2550 known
  - $+$?
- General applications
  - Lattice or Trellis graphics
  - Linear mixed-effects models
  - Modern Applied Statistics with S
  - Personality and psychometrics
  - General purpose toolkit
- More specialized packages
  - Latent Trait Model
  - SEM and CFA (one group)
  - SEM and CFA (multiple groups $+$)
  - Jennrich rotations
  - Multivariate distributions
  - Thousands of more packages on CRAN
  - Code on webpages/journal articles

An example package: psych

- The psych package is meant to be useful for basic data analysis for psychologists with a particular emphasis upon psychometrics and personality research
  - Some of the functions are used in an undergraduate research methods course and an advanced course in personality research
  - Additional functions used in graduate courses on psychometric theory and an introduction to sem
- Vignettes give detailed information and examples of using psych (are included in the psych package but may also be downloaded from http://personality-project.org/r/book/
  - Overview (http://cran.at.r-project.org/web/packages/psych/vignettes/overview.pdf)
  - Psych_for_Sem (http://cran.at.r-project.org/web/packages/psych/vignettes/psych_for_sem.pdf)
- data(package = "psych") #lists 38 different data sets included in psych

Overview of psych package – selected functions

- Descriptives
  - describe, describe.by
  - pairs.panels, error.bars, error.bars.by
- latent variable analyis
  - fa, iclust, principal,
  - irt.fa
- reliability analysis
  - score.items
  - omega, omegaSem
- graphic displays
- simulation
  - sim.structure, sim.minor, sim.hierarchical
  - sim.items, sim.irt, sim.congeneric,
  - sim.simplex, sim.omega
- preprocessing for sem

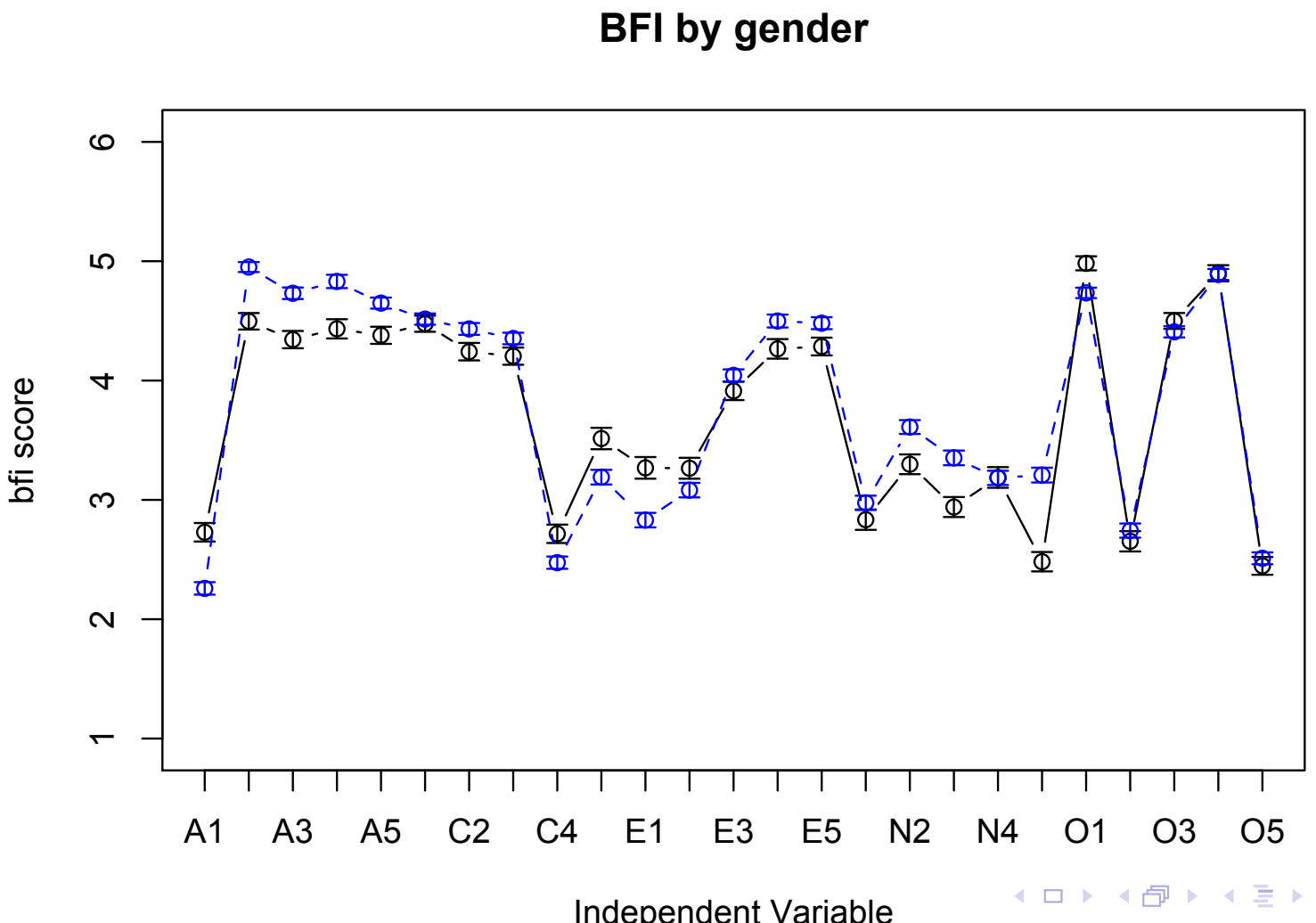
Data description

```
> data(sat.act)
> describe(sat.act)
```

	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

error.bars.by for the bfi items

```
data(bfi)
error.bars.by(bfi[1:25],bfi$gender,main="BFI by gender",ylab="bfi score")
```



Create a hierarchical model - factor the model

```
> my.data <- sim.hierarchical()
> f3 <- fa(my.data,3)
> f3
```

```
Factor Analysis using method =  minres
Call: fa(r = my.data, nfactors = 3)
Standardized loadings based upon correlation matrix
   MR1 MR2 MR3   h2   u2
V1 0.8 0.0 0.0 0.64 0.36
V2 0.7 0.0 0.0 0.49 0.51
V3 0.6 0.0 0.0 0.36 0.64
V4 0.0 0.7 0.0 0.49 0.51
V5 0.0 0.6 0.0 0.36 0.64
V6 0.0 0.5 0.0 0.25 0.75
V7 0.0 0.0 0.6 0.36 0.64
V8 0.0 0.0 0.5 0.25 0.75
V9 0.0 0.0 0.4 0.16 0.84

                 MR1  MR2  MR3
SS loadings     1.49 1.10 0.77
Proportion Var  0.17 0.12 0.09
Cumulative Var  0.17 0.29 0.37

 With factor correlations of
```

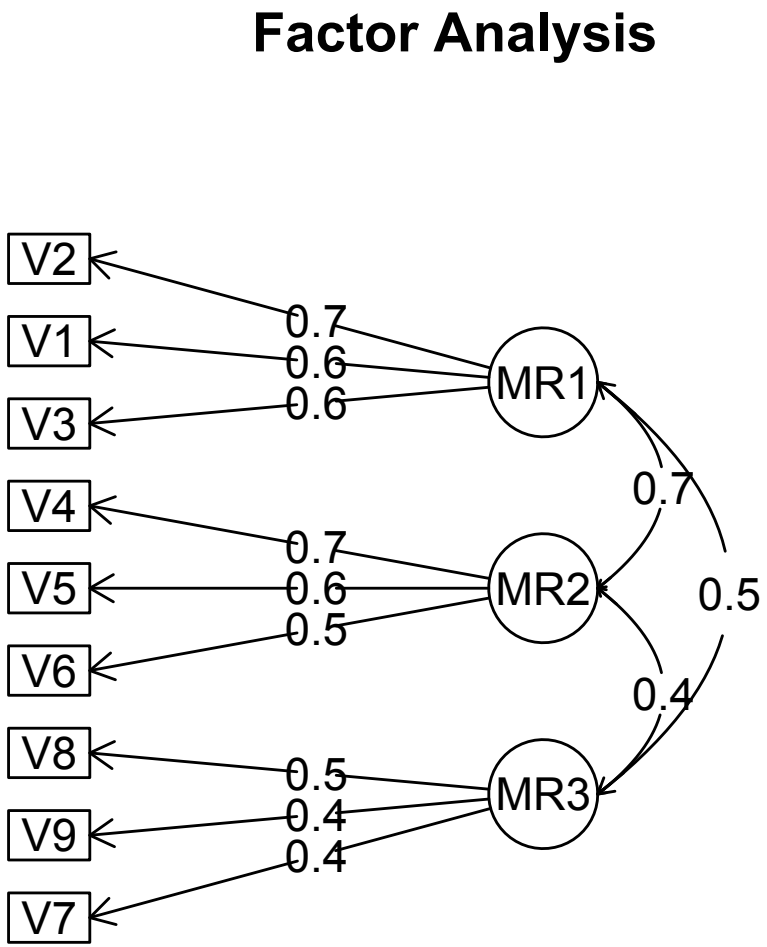
Create a hierarchical model - simulate it with 500 subjects

```
> set.seed(42)
> my.data <- sim.hierarchical(n=500,raw=TRUE)
> f3 <- fa(my.data$observed,3)
> f3
```

```
Factor Analysis using method =  minres
Call: fa(r = my.data$observed, nfactors = 3)
Standardized loadings based upon correlation matrix
     MR1  MR2   MR3   h2   u2
V1  0.64 0.01  0.14 0.53 0.47
V2  0.71 0.02 -0.01 0.52 0.48
V3  0.61 0.00 -0.09 0.32 0.68
V4  0.03 0.68  0.01 0.50 0.50
V5 -0.05 0.65  0.04 0.39 0.61
V6  0.11 0.46 -0.10 0.26 0.74
V7  0.02 0.14  0.37 0.21 0.79
V8  0.06 0.00  0.51 0.30 0.70
V9  0.02 0.00  0.42 0.19 0.81

                MR1  MR2  MR3
SS loadings    1.39 1.17 0.66
Proportion Var 0.15 0.13 0.07
Cumulative Var 0.15 0.28 0.36

 With factor correlations of
     MR1  MR2  MR3
MR1 1.00 0.71 0.51
MR2 0.71 1.00 0.39
MR3 0.51 0.39 1.00

Test of the hypothesis that 3 factors are sufficient.

The degrees of freedom for the null model are  36  and the objective function was  1.59 with Chi Square of
```

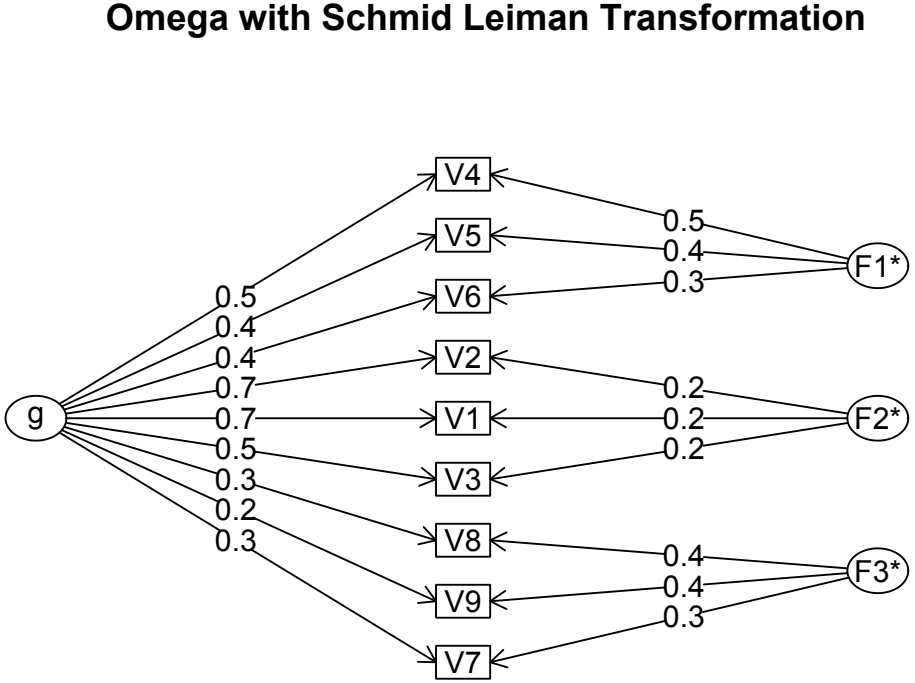
Plot the 3 factors

```
fa.diagram(f3)
```



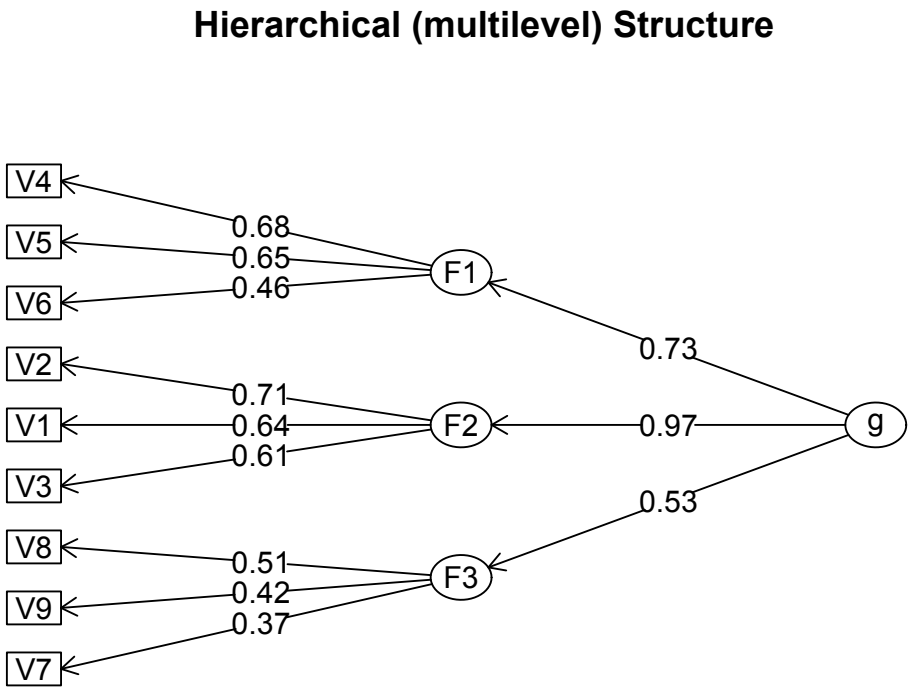
Do a Schmid Leiman transformation and find omega

```
om <- omega(my.data$observed)
omega.diagram(om,cut=.15)
```



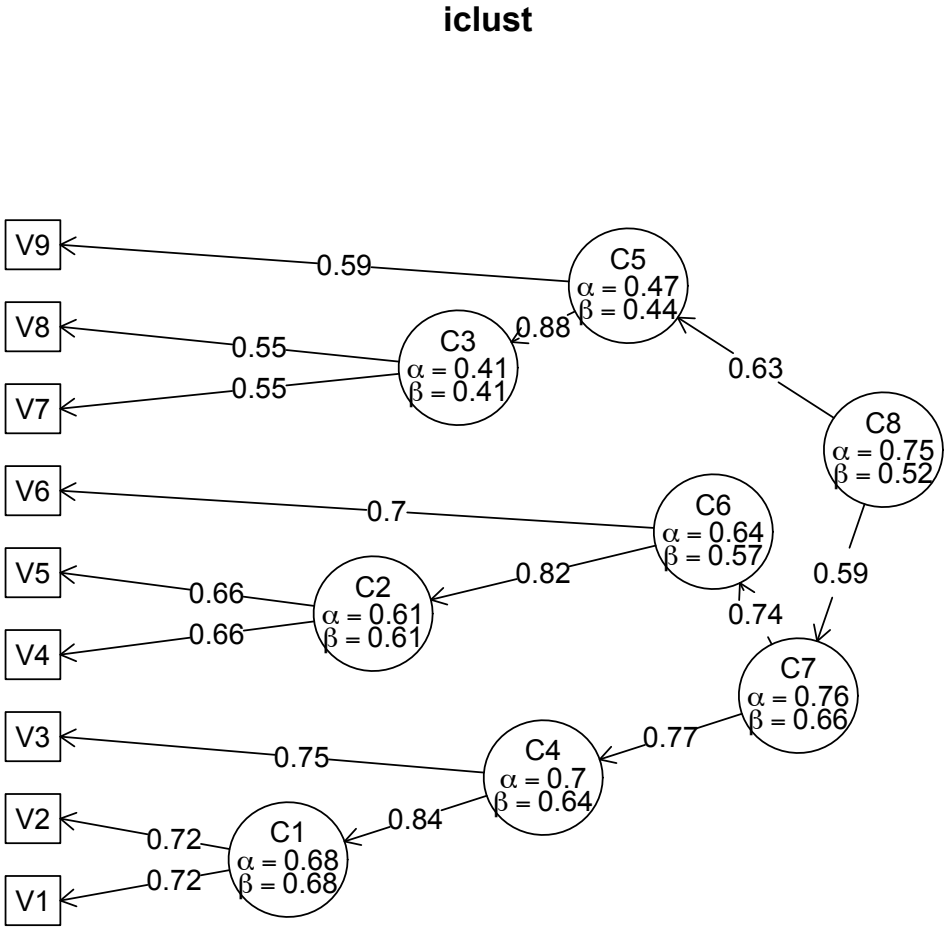
Show the same result, but with a hierarchical solution

```
> omega.diagram(om,cut=.15,sl=FALSE,digits=2)
```



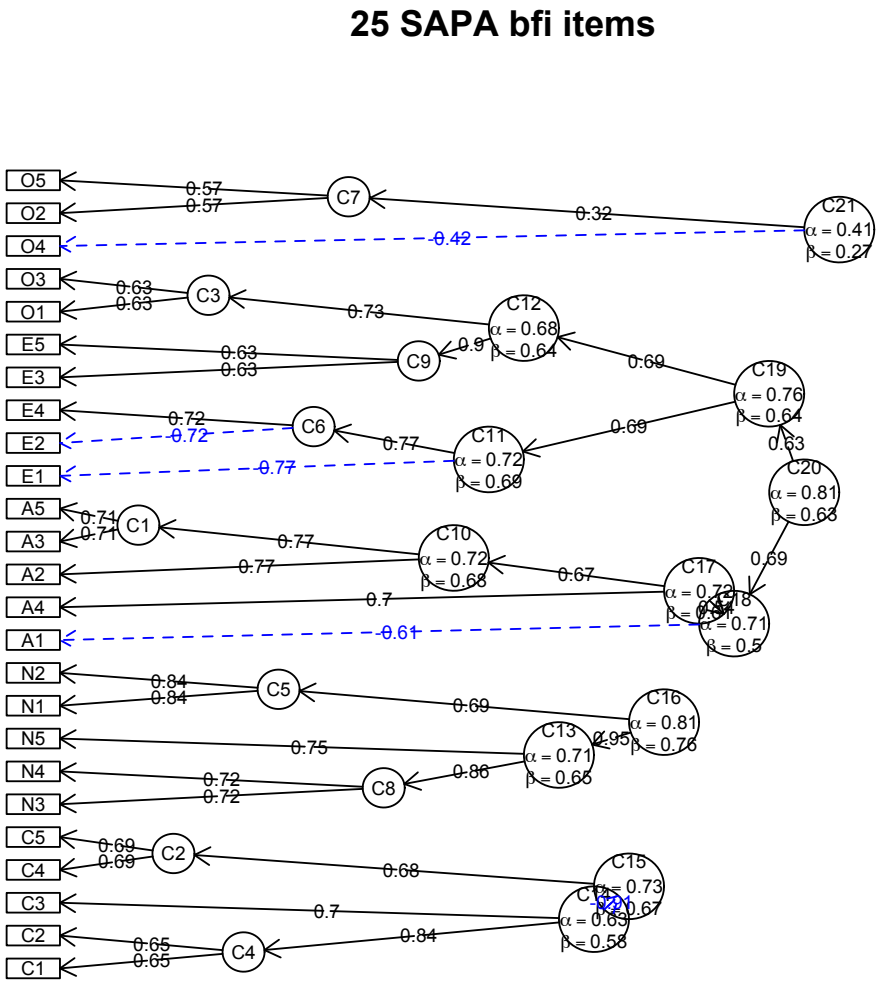
Compare to a hierarchical cluster analysis

```
> iclust(my.data$observed)
```



Item Cluster Analysis (iclust) of BFI – Pearson correlations

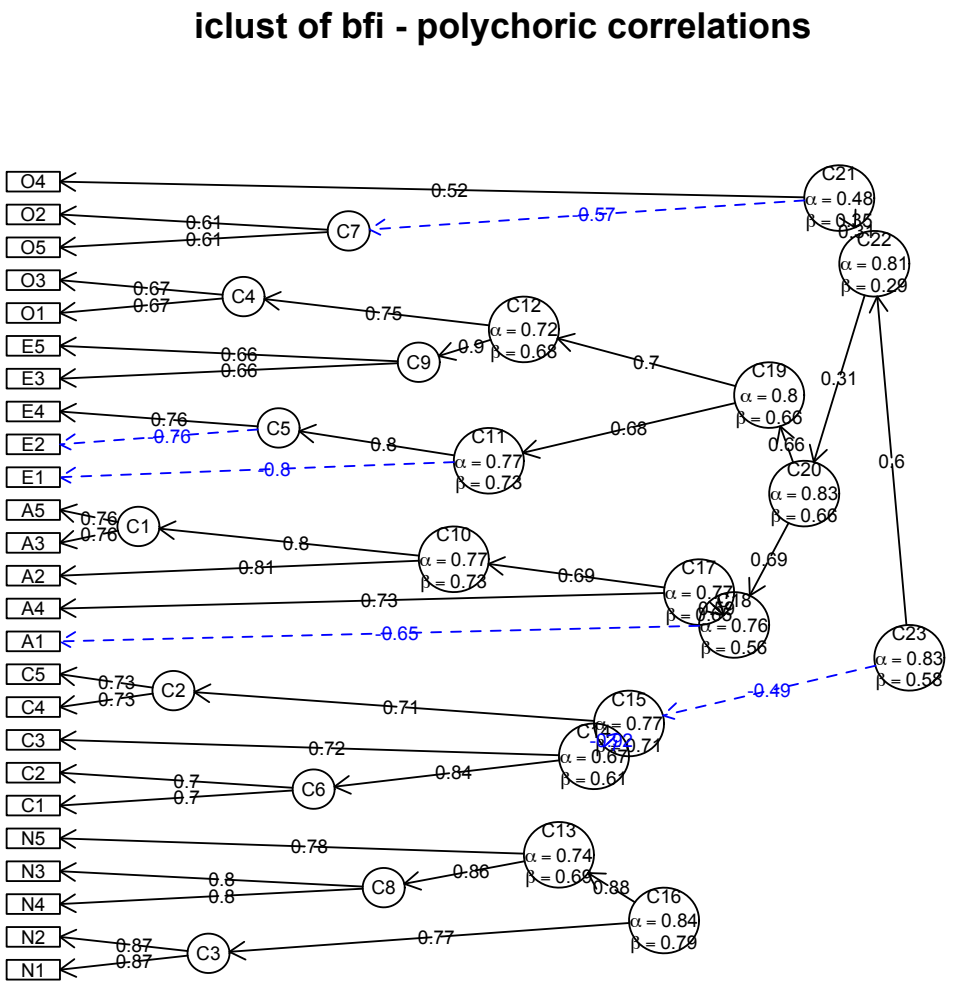
```
> data(bfi)
> ic <- iclust(bfi[1:25])
```



- 25 BFI items from SAPA (from 100)
- 2800 SAPA subjects
- Pearson Correlations

Item Cluster Analysis (iclust) of BFI – Polychoric correlations

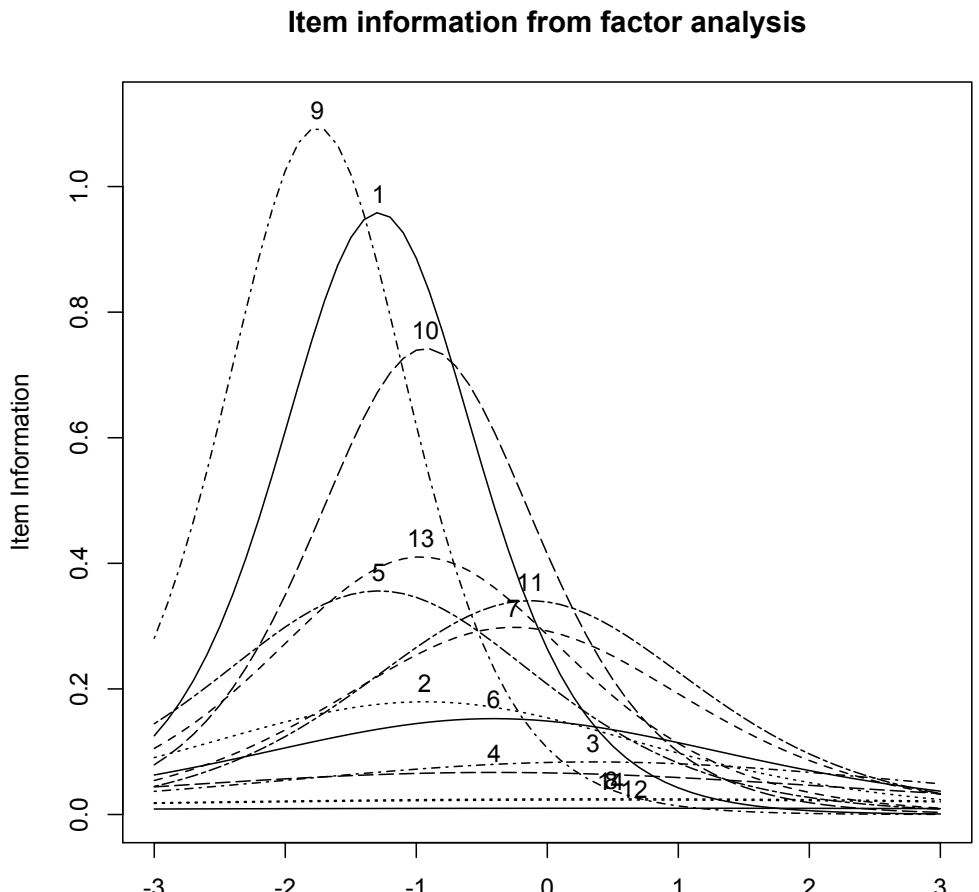
```
> data(bfi)
> bfi.poly <- polychoric(bfi[1:25]
> ic <- iclust(bfi.poly$rho)
```



- 25 BFI items from SAPA (from 100)
- 2800 SAPA subjects
- Polychoric Correlations

Item Response Theory analysis of homebrewed IQ items

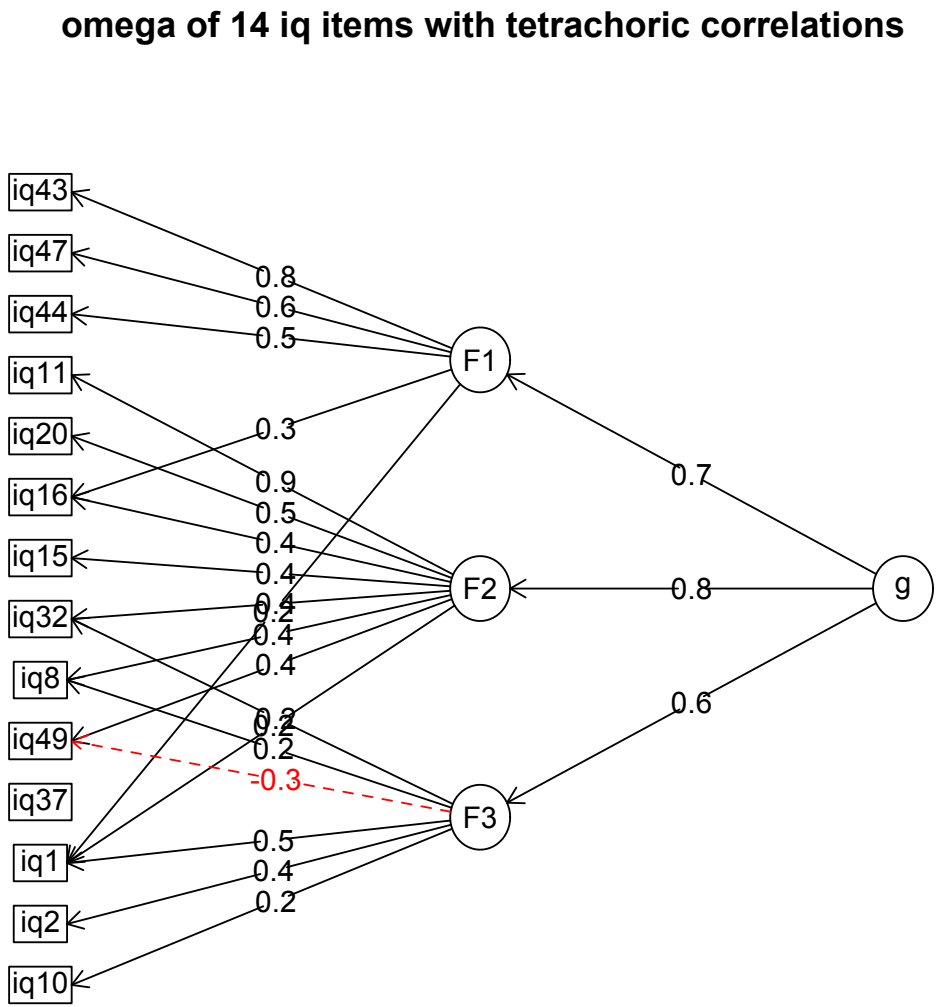
```
> data(iqitems)
> iq.keys <- c(4,4,3,1,4,3,2,3,1,4,1,3,4,3)
> iq.tf <- score.multiple.choice(iq.keys,iqitems,score=FALSE)  #just the responses
> iq.irt <- irt.fa(iq.tf)
> plot(iq.irt)
```



- 14 SAPA iq items (from 56)
- 1000 SAPA subjects
- Not screened for missing responses!
- factor analysis of tetrachoric correlations
- fa parameters -> irt parameters

Omega analysis of iq items with tetrachoric correlations

```
> om.iq <- omega(iq.irt$rho)
> omega.diagram(om.iq,sl=FALSE,main="omega of 14 iq items with tetrachoric correlatio
```



- tetrachorics were found by irt.fa
- 14 SAPA iq items (from 56)
- 1000 SAPA subjects
- F1 are geometric analogies
- F2 are reasoning and vocabulary
- F3 are number series + reasoning

Simulations of distributions

- Simulation of item structures
  - sim.irt, sim.rasch, sim.npl (unidimensional structures)
  - sim.congeneric
  - sim.item (2 dimensional structures, simple structure or circumplex)
- Simulation of test structures
  - sim.hierarchical (hierarchical test structure)
  - sim.simplex
  - sim.minor (n major factors, nvar/2 minor factors)

Conclusion

- R is "easy" to use
- Multiple packages are available
- psych package is appropriate for many analyses in personality and psychometrics
- Program development is straightforward
- Use of R should be encouraged for us, for our students
- General invitation for SMEP members to add to the psych package
- Should we create a smep package?