Factor Analysis with Binary items: A quick review with examples.

As published in Benchmarks RSS Matters, September 2014

http://web3.unt.edu/benchmarks/issues/2014/09/rss-matters

Jon Starkweather, PhD

Jon Starkweather, PhD
jonathan.starkweather@unt.edu
Consultant
Research and **S**tatistical **S**upport

http://www.unt.edu

http://www.unt.edu/rss

RSS hosts a number of “Short Courses”.
A list of them is available at:
http://www.unt.edu/rss/Instructional.htm

Those interested in learning more about R, or how to use it, can find information here:
http://www.unt.edu/rss/class/Jon/R_SC

Factor Analysis with Binary items: A quick review with examples.

There have been several clients in recent weeks that have come to us with binary survey data which they would like to factor analyze. The current article was written in order to provide a simple resource for others who may find themselves in a similar situation.

Of course, our professional conscience requires that we mention at the outset; if you are creating a survey (online, paper & pencil, or any other format) you should create the items and response choices in such a way that the responses may be considered interval or ratio; or at the very least, ordinal - not nominal categories and particularly not binary categories. We also feel compelled to advise you against the use of two other types of items. Please do not use any type of contingency or dependent items (e.g. if you answered 'yes' to item 6, go to item 6a; if you answered 'no' to item 6, please move forward to item 7). Please also do not use any type of multiple response items (e.g. 'choose all those which apply'). If you would like more information on why we make the recommendations above, please consult the substantial literature on survey development (e.g. McDonald, 1999; OECD, 2008, Statistics Canada, 2010).

1 Examples

First, import some (simulated) example data. The data used here is available at the URL given in the 'read.table' function below. The data contains eight binary items (x1, x2, x3, x4, x5, x6, x7, & x8) with 1000 cases (i.e. rows) which support two orthogonal factors.

```
df.1 <- read.table(
       "http://www.unt.edu/rss/class/Jon/Benchmarks/BinaryDataFA.txt",
        header = TRUE, sep = ",", na.strings = "NA", dec = ".")
head(df.1)
  x1 x2 x3 x4 x5 x6 x7 x8
1  0  0  0  0  0  0  0  0
2  0  1  1  1  0  0  0  0
3  0  0  0  0  0  0  0  0
4  0  0  0  0  0  0  0  0
5  0  0  0  0  0  0  0  0
6  0  1  0  0  0  0  0  0
nrow(df.1)
[1] 1000
```

Notice above, the data is numeric; this is important because if you simply supply this data to a factor analysis function, that function will (by default) calculate the matrix of association assuming those numbers are interval or ratio - which would be incorrect or potentially very biased. Therefore, what is really needed is a way to calculate the correct matrix of association (for the factor analysis) using the appropriate correlation statistic for each pair of variables in our data. Fortunately, the 'polycor' package (Fox, 2014) contains a function called 'hetcor' for doing just that. The 'hetcor' function basically looks at each pair of variables in a data frame and computes the appropriate *heterogeneous correlation* for each pair based on the type of variables which make up each pair. Recall that with categorical variables, the

polychoric correlation is appropriate, and the tetrachoric correlation is a special case of the polychoric correlation (for when both variables being correlated are binary). The 'hetcor' function is capable of calculating Pearson correlations (for numeric data), polyserial correlations (for numeric and ordinal data), and polychoric correlations (for ordered or non-ordered factors) - from a single data frame with all of the above mentioned types of variables.

So, because the data is imported as numeric, we must first recode it as factor (i.e. categorical); which can be done very easily using the 'sapply' function. There are other packages and functions which allow more precise control over recoding variables; such as the 'recode' function in the 'car' package (Fox, et al., 2014).

```
df.2 <- sapply(df.1, as.factor)
head(df.2)
     x1  x2  x3  x4  x5  x6  x7  x8
[1,] "0" "0" "0" "0" "0" "0" "0" "0"
[2,] "0" "1" "1" "1" "0" "0" "0" "0"
[3,] "0" "0" "0" "0" "0" "0" "0" "0"
[4,] "0" "0" "0" "0" "0" "0" "0" "0"
[5,] "0" "0" "0" "0" "0" "0" "0" "0"
[6,] "0" "1" "0" "0" "0" "0" "0" "0"
```

Once the numeric data have been recoded as factor, we can proceed by loading the 'polycor' package which contains the 'hetcor' function.

```
library(polycor)
Loading required package: mvtnorm
Loading required package: sfsmisc
```

Now we can compute the appropriate correlation matrix and assign that matrix to a new object (het.mat). Notice below, we are extracting only the correlation matrix ($cor) from the output of the 'hetcor' function.

```
het.mat <- hetcor(df.2)$cor
Warning messages:
1: In polychor(x, y, ML = ML, std.err = std.err) :
  inadmissible correlation set to 1
2: In hetcor.data.frame(dframe, ML = ML, std.err = std.err, bins = bins,  :
  the correlation matrix has been adjusted to make it positive-definite
het.mat
             x1           x2           x3           x4           x5
x1  1.000000000  0.910975550  0.844483311  0.691731074 -0.002245134
x2  0.910975550  1.000000000  0.859541108  0.808750265  0.037625262
x3  0.844483311  0.859541108  1.000000000  0.723304581 -0.026716610
x4  0.691731074  0.808750265  0.723304581  1.000000000 -0.001185206
x5 -0.002245134  0.037625262 -0.026716610 -0.001185206  1.000000000
x6 -0.039424602 -0.004851113 -0.046661991 -0.001214029  0.993573475
x7  0.002335945  0.005438252 -0.014930707 -0.009831874  0.879110898
```

```
x8 -0.036916591 -0.054512229  0.006043798  0.031313650  0.794959194
             x6           x7           x8
x1 -0.039424602  0.002335945 -0.036916591
x2 -0.004851113  0.005438252 -0.054512229
x3 -0.046661991 -0.014930707  0.006043798
x4 -0.001214029 -0.009831874  0.031313650
x5  0.993573475  0.879110898  0.794959194
x6  1.000000000  0.849171046  0.781588616
x7  0.849171046  1.000000000  0.703973732
x8  0.781588616  0.703973732  1.000000000
```

Although there are two warnings listed above, the function does in fact return the appropriate correlation matrix. Now we can proceed with the factor analysis using this 'het.mat' correlation matrix as the matrix of association for the factor analysis.

```
fa.1 <- factanal(covmat = het.mat, factors = 2, rotation = "varimax")
fa.1

Call:
factanal(factors = 2, covmat = het.mat, rotation = "varimax")

Uniquenesses:
   x1    x2    x3    x4    x5    x6    x7    x8
0.164 0.005 0.252 0.345 0.005 0.008 0.243 0.368

Loadings:
   Factor1 Factor2
x1          0.913
x2          0.997
x3          0.863
x4          0.809
x5  0.997
x6  0.996
x7  0.870
x8  0.794

               Factor1 Factor2
SS loadings      3.378   3.232
Proportion Var   0.422   0.404
Cumulative Var   0.422   0.826

The degrees of freedom for the model is 13 and the fit was 12.2084
```

Another equally effective way to factor analyze binary data (or any other type of data), using a correlation matrix, is with the 'fa' function from the 'psych' package (Revelle, 2014). Again, we use the correlation matrix we generated with the 'hetcor' function. Please note, the default method of extraction for the 'fa' function is minimum residuals (method = minres) and not maximum likelihood (method = ml).

```
library(psych)
fa.2 <- fa(r = het.mat, nfactors = 2, n.obs = nrow(df.2), rotate = "varimax")
Loading required package: MASS
Loading required package: GPArotation
Loading required package: parallel
fa.2
Factor Analysis using method =  minres
Call: fa(r = het.mat, nfactors = 2, n.obs = nrow(df.2), rotate = "varimax")
Standardized loadings (pattern matrix) based upon correlation matrix
     MR1  MR2   h2    u2 com
x1 -0.11 0.93 0.87 0.128   1
x2 -0.09 0.96 0.94 0.062   1
x3 -0.11 0.92 0.86 0.141   1
x4 -0.07 0.86 0.75 0.250   1
x5  0.98 0.10 0.96 0.036   1
x6  0.97 0.08 0.94 0.058   1
x7  0.91 0.09 0.84 0.160   1
x8  0.87 0.07 0.76 0.242   1

                       MR1  MR2
SS loadings           3.51 3.41
Proportion Var        0.44 0.43
Cumulative Var        0.44 0.87
Proportion Explained  0.51 0.49
Cumulative Proportion 0.51 1.00

Mean item complexity =  1
Test of the hypothesis that 2 factors are sufficient.

The degrees of freedom for the null model are  28  and the objective function was
The degrees of freedom for the model are 13  and the objective function was  13.77

The root mean square of the residuals (RMSR) is  0.04
The df corrected root mean square of the residuals is  0.06

The harmonic number of observations is  1000 with the empirical chi square  99.24
The total number of observations was  1000  with MLE Chi Square =  13694.45  with

Tucker Lewis Index of factoring reliability =  -0.273
RMSEA index =  1.029  and the 90 % confidence intervals are  1.011 1.04
BIC =  13604.65
Fit based upon off diagonal values = 0.99
Measures of factor score adequacy
                                                MR1 MR2
Correlation of scores with factors                1   1
Multiple R square of scores with factors          1   1
```

```
Minimum correlation of possible factor scores    1   1
```

2 Conclusions

As demonstrated above, using binary data for factor analysis in R is no more difficult than using continuous data for factor analysis in R. Although not demonstrated here, if one has polytomous and other types of mixed variables one wants to factor analyze, one may want to use the 'hetcor' function (i.e. heterogeneous correlations) located in the 'polycor' package (Fox, 2014). An example of the use of the 'hetcor' function is available at the RSS *Do-it-yourself Introduction to R* course page[1] where many other examples (not just factor analysis) are provided. Lastly, a copy of the script file used for the above examples is available here[2].

Until next time; remember what George Carlin said: "*inside every cynical person, there is a disappointed idealist.*"

[1] http://www.unt.edu/rss/class/Jon/R_SC/
[2] http://www.unt.edu/rss/class/Jon/Benchmarks/BinaryFA.R

3 References & Resources

Carlin, G. (1937 - 2008). http://www.just-one-liners.com/ppl/george-carlin

Fox, J. (2014). The 'polycor' package. Documentation available at CRAN:
http://cran.r-project.org/web/packages/polycor/index.html

Fox, J., et al. (2014). The 'car' package. Documentation available at CRAN:
http://cran.r-project.org/web/packages/car/index.html

McDonald, R. P. (1999). Test Theory: A Unified Treatment. Mahwah, NJ: Erlbaum.

Organization for Economic Co-operation and Development (OECD). (2008). Handbook on Constructing Composite Indicators.
http://www.oecd.org/std/42495745.pdf

Revelle, W. (2014). The 'psych' package. Documentation available at CRAN:
http://cran.r-project.org/web/packages/psych/index.html

Statistics Canada. (2010). Survey Methods and Practices. Ottawa, Canada: Minister of Industry.
http://www.statcan.gc.ca/bsolc/olc-cel/olc-cel?lang=eng&catno=12-587-X

This article was last updated on December 8, 2014.

This document was created using LaTeX