

Using R to score personality scales

PMC lab
Northwestern University

Abstract

The *psych* package Revelle (2010) was developed to perform most basic psychometric functions using R (R Development Core Team, 2009). One frequently called upon is the need to take a set of items (e.g., a questionnaire) and score one or more scales on that questionnaire. Scores for subsequent analysis, reliabilities and intercorrelations are easily done using the `score.items` function.

Suppose you have given a questionnaire with some items (n) to some participants (N). You would like to create scale scores for each person on k different scales. This may be done using the *psych* package in R. The following assumes that you have installed R and downloaded the *psych* package.

Overview

Remember, before using *psych* you must make it active:

library(psych)

1. Enter the data into a spreadsheet (Excel or Numbers) or a text file using a text editor (Word, Pages, BBEdit). The first line of the file should include names for the variables (e.g., Q1, Q2, ... Qn).
2. Copy the data to the clipboard (using the normal copy command for your spreadsheet or word processor).
3. Read the data into R using the `read.clipboard` command. (Depending upon your data file, this might need to be `read.clipboard.csv` (for comma separated data fields) or `read.clipboard.tab` (for tab separated data fields).

4. Construct a set of scoring keys for the scales you want to score. This is simply the item numbers that go into each scale. A negative sign implies that the item will be reverse scored.

5. Use the `score.items` function to score the scales.

6. Use the output for `score.items` for further analysis.

An example

Suppose we have 12 items for 20 subjects. The items represent 4 different scales:

Table 1: A sample data file with 12 items for 20 subjects.

	active	alert	aroused	sleepy	tired	drowsy	anxious	jittery	nervous	calm	relaxed	at-ease
1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	0	1	1	1	0	0	0	1	1	1
3	1	0	0	0	1	0	0	0	0	1	2	2
4	1	1	1	1	1	1	1	3	2	1	2	1
5	2	1	2	1	1	1	NA	1	0	3	3	3
6	2	1	1	2	2	2	NA	0	0	2	2	1
7	0	1	0	2	3	3	NA	0	0	2	2	1
8	0	0	0	1	2	1	NA	0	0	1	2	0
9	1	0	1	2	0	2	NA	1	0	0	2	2
10	0	2	0	2	2	2	NA	1	0	2	2	1
11	0	0	0	3	2	2	NA	0	0	2	2	2
12	1	1	0	1	1	1	NA	1	0	1	1	0
13	0	0	0	3	3	2	NA	1	0	0	2	0
14	2	1	1	1	0	0	NA	0	0	2	2	1
15	0	2	0	0	2	1	NA	0	0	3	3	3
16	0	0	0	3	3	3	NA	1	0	1	1	1
17	0	1	1	1	1	1	NA	0	0	1	1	1
18	3	2	0	2	2	3	NA	0	0	3	3	3
19	0	0	0	3	3	2	NA	0	0	2	1	0
20	0	1	0	1	2	1	NA	0	0	3	2	2

```
library(psych)
my.data <- read.clipboard.tab() #tab delimited data from a spreadsheet or
my.data<- read.clipboard() #data from a text editor with spaces between the fields.
describe(my.data) # to make sure you got the right data in.
my.keys <- make.keys(nvars=12,list(EA=c(1:3,-4,-5,-6),TA=c(7:9,-10,-11,-12)))
my.scales <- score.items(my.keys,my.data)
```

```
my.scales #show the output
my.scores <- my.scales$scores
```

Produces this output:

```
> describe(my.data) # to make sure you got the right data in.
      var  n mean  sd median trimmed  mad min max range  skew kurtosis  se
active   1 20 0.75 0.91   0.5   0.62 0.74   0  3   3  0.87   -0.37 0.20
alert    2 20 0.80 0.70   1.0   0.75 0.74   0  2   2  0.25   -1.06 0.16
aroused  3 20 0.40 0.60   0.0   0.31 0.00   0  2   2  1.06   -0.01 0.13
sleepy   4 20 1.55 0.94   1.0   1.56 1.48   0  3   3  0.22   -1.10 0.21
tired    5 20 1.65 0.93   2.0   1.69 1.48   0  3   3 -0.05   -1.06 0.21
drowsy   6 20 1.50 0.89   1.0   1.50 1.48   0  3   3  0.21   -0.89 0.20
anxious  7  4 0.50 0.58   0.5   0.50 0.74   0  1   1  0.00   -2.44 0.29
jittery  8 20 0.50 0.76   0.0   0.38 0.00   0  3   3  1.70    3.00 0.17
nervous  9 20 0.15 0.49   0.0   0.00 0.00   0  2   2  2.94    7.68 0.11
calm     10 20 1.60 0.94   1.5   1.62 0.74   0  3   3  0.09   -1.10 0.21
relaxed  11 20 1.85 0.67   2.0   1.81 0.00   1  3   2  0.15   -0.93 0.15
at.ease  12 20 1.30 0.98   1.0   1.25 1.48   0  3   3  0.38   -0.96 0.22
```

```
> my.keys <- make.keys(nvars=12,list(EA=c(1:3,-4,-5,-6),TA=c(7:9,-10,-11,-12)))
> my.scales <- score.items(my.keys,my.data)
```

```
> my.scales #show the output
Call: score.items(keys = my.keys, items = my.data)
```

(Unstandardized) Alpha:

```
      EA  TA
alpha 0.77 0.73
```

Average item correlation:

```
      EA  TA
average.r 0.36 0.31
```

Guttman 6* reliability:

```
      EA  TA
Lambda.6 0.92 0.91
```

Scale intercorrelations corrected for attenuation

raw correlations below the diagonal, alpha on the diagonal
corrected correlations above the diagonal:

```

      EA    TA
EA  0.77 -0.22
TA -0.16  0.73

```

```

Item by scale correlations:
corrected for item overlap and scale reliability

```

```

      EA    TA
active  0.55 -0.29
alert   0.40 -0.42
aroused 0.57  0.06
sleepy -0.79  0.20
tired  -0.85 -0.08
drowsy -0.73 -0.05
anxious 0.03  0.40
jittery 0.12  0.59
nervous 0.27  0.55
calm    0.17 -0.78
relaxed 0.26 -0.65
at.ease 0.38 -0.74
> my.scores <- my.scales$scores
>
> my.scores

```

```

      EA          TA
1  1.5000000 1.5000000
2  1.3333333 1.0000000
3  1.5000000 0.6666667
4  1.5000000 1.8333333
5  1.8333333 0.2500000
6  1.1666667 0.7500000
7  0.3333333 0.7500000
8  0.8333333 1.0833333
9  1.1666667 1.0833333
10 0.8333333 0.9166667
11 0.3333333 0.5833333
12 1.3333333 1.4166667
13 0.1666667 1.4166667
14 2.0000000 0.7500000
15 1.3333333 0.0833333
16 0.0000000 1.2500000
17 1.3333333 1.0833333
18 1.1666667 0.0833333

```

```
19 0.1666667 1.08333333
20 1.0000000 0.41666667
```

The example, continued

Suppose after looking at the data that we want to score the items differently, and form 4 scales instead of 2.

```
my.keys <- make.keys(nvars=12,list(EAP=c(1:3),EAN =c(4:6),TAP=c(7:9),TAN =c(10:12)))
my.scales <- score.items(my.keys,my.data)
my.scales #show the output
my.scores <- my.scales$scores
```

Produces this output:

```
> my.keys <- make.keys(nvars=12,list(EAP=c(1:3),EAN =c(4:6),TAP=c(7:9),TAN =c(10:12)))
> my.scales <- score.items(my.keys,my.data)
> my.scales #show the output
Call: score.items(keys = my.keys, items = my.data)
```

```
(Unstandardized) Alpha:
      EAP  EAN  TAP  TAN
alpha 0.57 0.86 0.78 0.82
```

```
Average item correlation:
      EAP  EAN  TAP  TAN
average.r 0.3 0.68 0.54 0.6
```

```
Guttman 6* reliability:
      EAP  EAN  TAP  TAN
Lambda.6 0.78 0.94 0.89 0.9
```

```
Scale intercorrelations corrected for attenuation
raw correlations below the diagonal, alpha on the diagonal
corrected correlations above the diagonal:
      EAP  EAN  TAP  TAN
EAP  0.57 -0.57 0.29 0.73
EAN -0.40 0.86 -0.12 -0.11
TAP 0.19 -0.10 0.78 -0.24
```

TAN 0.50 -0.09 -0.20 0.82

Item by scale correlations:

corrected for item overlap and scale reliability

	EAP	EAN	TAP	TAN
active	0.75	-0.30	0.06	0.40
alert	0.58	-0.20	0.07	0.57
aroused	0.60	-0.43	0.40	0.17
sleepy	-0.40	0.86	-0.03	-0.27
tired	-0.60	0.82	-0.20	-0.02
drowsy	-0.18	0.90	-0.05	0.04
anxious	0.24	0.10	0.77	-0.05
jittery	0.17	-0.06	0.86	-0.24
nervous	0.23	-0.23	0.91	-0.16
calm	0.45	0.04	-0.30	0.81
relaxed	0.48	-0.06	-0.06	0.79
at.ease	0.53	-0.21	-0.14	0.85

Even more analysis

Far more analyses could be done with these data, but the basic scale scoring techniques is a start. Download the vignette for using *psych* for even more guidance. <http://cran.r-project.org/web/packages/psych/vignettes/overview.pdf>

References

- R Development Core Team. (2009). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Available from <http://www.R-project.org> (ISBN 3-900051-07-0)
- Revelle, W. (2010). *psych*: Procedures for personality and psychological research [Computer software manual]. Available from <http://personality-project.org/r> (R package version 1.0-88)