How to use the psych package to display item and scale correlations

William Revelle

September 20, 2022

Contents

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" to do various things using R (R Core Team, 2020), particularly using the *psych* (Revelle, 2020) package.

The current list of How To's includes:

1. An introduction (vignette) of the *psych* package

2. An overview (vignette) of the *psych* package

3. Installing R and some useful packages

4. Using R and the *psych* package to find $omega_h$ and ω_t.

5. Using R and the *psych* for factor analysis and principal components analysis.

6. Using the scoreItems function to find scale scores and scale statistics.

7. Using mediate and setCor to do mediation, moderation and regression analysis (this document)

1.1 Jump starting the *psych* package–a guide for the impatient

You have installed *psych* and you want to use it without reading much more. What should you do?

1. Activate the *psych* and *psychTools* packages.

```
R code
> library(psych)
> library(psychTools)
```

2. Input your data. If your file name ends in .sav, .text, .txt, .csv, .xpt, .rds, .Rds, .rda, or .RDATA, then just read it in directly using read.file. Or you can go to your friendly text editor or data manipulation program (e.g., Excel) and copy the data to the clipboard. Include a first line that has the variable labels. Paste it into *psych* using the read.clipboard.tab command:

```
R code
myData <- read.file()      #this will open a search window on your machine
#                                 and read or load the file.
#or
#first copy your file to your clipboard and then
myData <- read.clipboard.tab()  #if you have an excel file
```

3. Make sure that what you just read is right. Describe it and perhaps look at the first and last few lines. If you want to "view" the first and last few lines using a spreadsheet like viewer, use quickView.

3

```
                                             ┌─ R code ─┐
  describe(myData)
  headTail(myData)
  #or
  quickView(myData)
```

4. Look at the patterns in the data. If you have fewer than about 10 variables, look at the
   SPLOM (Scatter Plot Matrix) of the data using `pairs.panels`.
   ```
                                             ┌─ R code ─┐
   pairs.panels(myData)
   ```

5. Find the correlations of all of your data.

   • Descriptively (just the values)
     ```
                                             ┌─ R code ─┐
     lowerCor(myData)
     ```

   • Graphically
     ```
                                             ┌─ R code ─┐
     corPlot(myData)    #does not show the values of the correlations
     corPlot(myData, numbers=TRUE)   #show the numbers,\
                             #scales the character size by "significance"
     corPlot(myData, numbers=TRUE,scale=FALSE)   #show the numbers,
                                 #  all characters the same size
     corPlot(lowerCor(myData), numbers =TRUE)    #print the correlations
                          # and show them graphically
     ```

1.2 For the not impatient

The following pages are meant to lead you through the use of the `setCor` and `mediate` func-
tions. The assumption is that you have already made *psych* active and want some example code.

2 Multiple regression and mediation

Mediation and moderation are merely different uses of the linear model $\hat{Y} = \mu + \beta_{y.x}X$ and are
implemented in *psych* with two functions: `setCor` and `mediate`.

Given a set of predictor variables, X and a set of criteria variables, Y, multiple regression solves the
equation $\hat{Y} = \mu + \beta_{y.x}X$ by finding $\beta_{y.x} = C_{xx}^{-1}C_{yx}$ where C_{xx} is the covariances of the X variables
and C_{yx} is the covariances of predictors and the criteria.

Although typically done using the raw data, clearly this can also be done by using the covariance
or correlation matrices. `setCor` was developed to handle the correlation matrix solution but has
been generalized to the case of raw data. In the later case, it assumes a Missing Completely at
Random (MCAR) structure, and thus uses all the data and finds pair.wise complete correlations.
For complete data sets, the results are identical to using `lm`. By default, `setCor` uses standardized
variables, but to compare with `lm`, it can use unstandardized variables.

3 Regression using `setCor`

Although typically done from a raw data matrix (using the `lm` function), it is sometimes useful to do the regression from a correlation or covariance matrix. `setCor` was developed for this purpose. From a correlation/covariance matrix, it will do normal regression as well as regression on partialled correlation matrices. With the raw data, it will also do moderated regression (centered or non-centered). In particular, for the raw data, it will work with missing data.

An interesting option, if using categorical or dichotomous data is first find the appropriate polychoric, tetrachoric, or poly-serial correlations using `mixedCor` and then use the resulting correlation matrix for analysis. The resulting correlations and multiple correlations will not match those of the `lm` analysis.

3.1 Comparison with `lm` on complete data

We use the `attitude` data set for our first example.

3.1.1 It is important to know your data by describing it first

```
R code
> psych::describe(attitude)
```

```
            vars  n  mean    sd median trimmed   mad min max range  skew kurtosis   se
rating         1 30 64.63 12.17   65.5   65.21 10.38  40  85    45 -0.36    -0.77 2.22
complaints     2 30 66.60 13.31   65.0   67.08 14.83  37  90    53 -0.22    -0.68 2.43
privileges     3 30 53.13 12.24   51.5   52.75 10.38  30  83    53  0.38    -0.41 2.23
learning       4 30 56.37 11.74   56.5   56.58 14.83  34  75    41 -0.05    -1.22 2.14
raises         5 30 64.63 10.40   63.5   64.50 11.12  43  88    45  0.20    -0.60 1.90
critical       6 30 74.77  9.89   77.5   75.83  7.41  49  92    43 -0.87     0.17 1.81
advance        7 30 42.93 10.29   41.0   41.83  8.90  25  72    47  0.85     0.47 1.88
```

3.1.2 Now do the regressions

```
R code
>   #do not standardize
> mod1 <- setCor(rating ~ complaints + privileges, data=attitude,std=FALSE)
> mod1
```

```
Call: setCor(y = rating ~ complaints + privileges, data = attitude,
    std = FALSE)

Multiple Regression from raw data

 DV =  rating
            slope   se     t        p lower.ci upper.ci  VIF   Vy.x
(Intercept) 15.33 7.16  2.14 4.1e-02     0.64    30.02 1.00   0.00
complaints   0.78 0.12  6.54 5.2e-07     0.54     1.03 1.45   0.70
privileges  -0.05 0.13 -0.39 7.0e-01    -0.32     0.22 1.45  -0.02

Residual Standard Error =  7.1  with  27  degrees of freedom

 Multiple Regression
```

5

```
             R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
rating 0.83 0.68 0.62 0.38         0.66       0.08       29.1   2  27 1.83e-07
```

Compare this solution with the results of the `lm` function.

```
┌─ R code ─────────────────────────────────────────────────────────────────┐
│ > summary(lm(rating ~ complaints + privileges, data=attitude))            │
└───────────────────────────────────────────────────────────────────────────┘

Call:
lm(formula = rating ~ complaints + privileges, data = attitude)

Residuals:
    Min      1Q   Median      3Q     Max
-12.7887  -5.6893  -0.0284   6.2745  9.9726

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 15.32762    7.16023   2.141   0.0415 *
complaints   0.78034    0.11939   6.536 5.22e-07 ***
privileges  -0.05016    0.12992  -0.386   0.7025
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.102 on 27 degrees of freedom
Multiple R-squared:  0.6831,     Adjusted R-squared:  0.6596
F-statistic:  29.1 on 2 and 27 DF,  p-value: 1.833e-07
```

The graphic for the standardized regression is shown in (Figure 1).

```
Call: setCor(y = rating ~ complaints + privileges, data = attitude)

Multiple Regression from raw data

 DV =  rating
            slope   se     t       p lower.ci upper.ci  VIF  Vy.x
(Intercept)  0.00 0.11  0.00 1.0e+00    -0.22     0.22 1.00  0.00
complaints   0.85 0.13  6.54 5.2e-07     0.59     1.12 1.45  0.70
privileges  -0.05 0.13 -0.39 7.0e-01    -0.32     0.22 1.45 -0.02

Residual Standard Error =  0.58  with  27  degrees of freedom

 Multiple Regression
            R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
rating 0.83 0.68 0.71 0.5          0.66       0.08       29.1   2  27 1.83e-07

pdf
  2
```

3.2   From a correlation matrix

Perhaps most usefully, `setCor` will find the beta weights between a set of X variables, and a set of Y variables. Consider seven variables in the `atttitude` data set. We first find the correlation matrix (normally, this could just be supplied by the user). Then we find the regressions from the correlation matrix. Compare this regression to the (standardized) solution shown above. By specifying the number of observations (n.obs), we are able to apply various inferential tests.
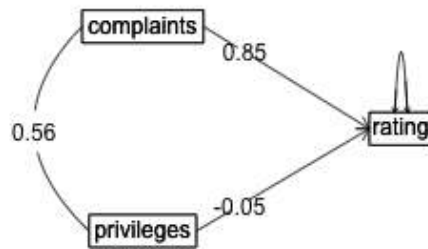
A simple regression model

Figure 1: A simple multiple regression using the attitude data set (standardized solution is shown).

```
                                    ┌─ R code ─┐
> R <- lowerCor(attitude)
```

```
           ratng cmpln prvlg lrnng raiss crtcl advnc
rating     1.00
complaints 0.83  1.00
privileges 0.43  0.56  1.00
learning   0.62  0.60  0.49  1.00
raises     0.59  0.67  0.45  0.64  1.00
critical   0.16  0.19  0.15  0.12  0.38  1.00
advance    0.16  0.22  0.34  0.53  0.57  0.28  1.00
```

```
                                    ┌─ R code ─┐
>  setCor(rating ~ complaints + privileges, data=R, n.obs =30)
```

```
Call: setCor(y = rating ~ complaints + privileges, data = R, n.obs = 30)

Multiple Regression from matrix input

 DV =  rating
            slope   se     t       p lower.ci upper.ci  VIF   Vy.x
complaints   0.85 0.13  6.54 5.2e-07     0.59     1.12 1.45   0.70
privileges  -0.05 0.13 -0.39 7.0e-01    -0.32     0.22 1.45  -0.02

Residual Standard Error =  0.58  with  27  degrees of freedom

 Multiple Regression
          R    R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
rating 0.83 0.68 0.71  0.5        0.66     0.08      29.1   2  27 1.83e-07
```

Compare this solution (from the correlation matrix) with the *standardized* solution for the raw data.
setCor does several things:

- Finds the regression weights (betas) between the predictor variables and each of the criterion variables.

- If the number of subjects is specified, or if the raw data are used, it also compares each of these betas to its standard error, finds a t statistic, and reports the probability of the $t > 0$.

- It reports the Multiple R and R^2 based upon these beta weights. In addition, following the tradition of the robust beauty of the improper linear models (Dawes, 1979) it also reports the unit weighted multiple correlations.

- The canonical correlations between the two sets (Hotelling, 1936) is reported.

- Cohen's set correlation (Cohen, 1982) as well as the unweighted correlation between the two sets of variables are reported.

3.3 The Hotelling example

```
                                    ┌─ R code ─┐
> #the second Kelley data from Hotelling
> kelley <- structure(list(speed = c(1, 0.4248, 0.042, 0.0215, 0.0573), power = c(0.4248,
+ 1, 0.1487, 0.2489, 0.2843), words = c(0.042, 0.1487, 1, 0.6693,
+ 0.4662), symbols = c(0.0215, 0.2489, 0.6693, 1, 0.6915), meaningless = c(0.0573,
```

```
+ 0.2843, 0.4662, 0.6915, 1)), .Names = c("speed", "power", "words",
+ "symbols", "meaningless"), class = "data.frame", row.names = c("speed",
+ "power", "words", "symbols", "meaningless"))
> #first show the correlations
> lowerMat(kelley)
```

```
            speed power words symbl mnngl
speed        1.00
power        0.42  1.00
words        0.04  0.15  1.00
symbols      0.02  0.25  0.67  1.00
meaningless  0.06  0.28  0.47  0.69  1.00
```

R code

```
> #now find and draw the regression
> sc <- setCor(power + speed ~ words + symbols + meaningless,data=kelley)  #formula mode
> sc  #show it
```

```
Call: setCor(y = power + speed ~ words + symbols + meaningless, data = kelley)

Multiple Regression from matrix input

 DV =  power
            slope  VIF  Vy.x
words       -0.03 1.81 -0.01
symbols      0.12 2.72  0.03
meaningless  0.22 1.92  0.06

 Multiple Regression
         R    R2  Ruw R2uw
power 0.29 0.09 0.26 0.07

 DV =  speed
            slope  VIF Vy.x
words        0.05 1.81    0
symbols     -0.07 2.72    0
meaningless  0.08 1.92    0

 Multiple Regression
         R    R2  Ruw R2uw
speed 0.07 0.01 0.05    0

Various estimates of between set correlations
Squared Canonical Correlations
[1] 0.0946 0.0035

 Average squared canonical correlation =  0.05
 Cohen's Set Correlation R2 =  0.1
Unweighted correlation between the two sets =  0.18
```

A plot of the regression model is shown as well (Figure 2).

```
pdf
  2
```

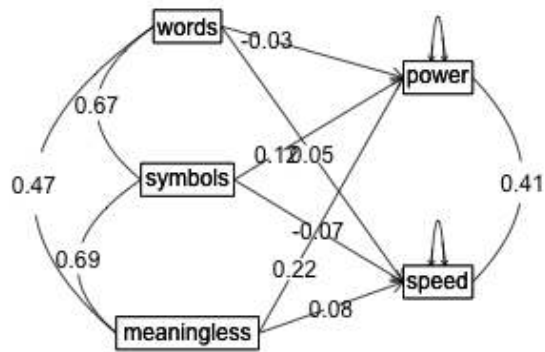But, we can also do multiple predictors *and* multiple criteria in the same call:

```
pdf
  2
```

Figure 2: The relationship between three predictors and two criteria from `setCor`. The data are from the Kelley data set reported by Hotelling (1936).
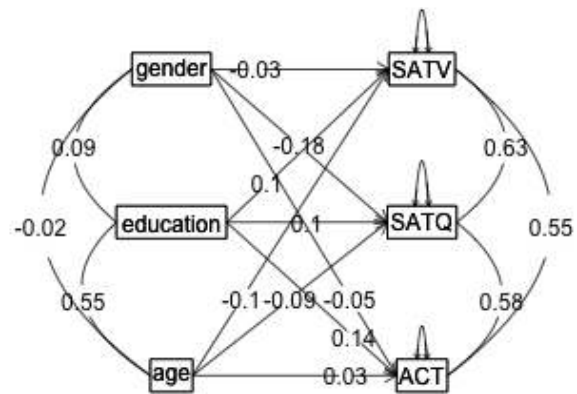
Regression Models

Figure 3: The relationship between three predictors and three criteria from `setCor`. The data are from the `sat.act` data set.

3.4 Moderated multiple regression

If we have the raw data, we can also find interactions (known as moderated multiple regression). This is done by zero centering the data (Cohen et al., 2003) and then multiplying the two terms of the interaction. As an option, we can not zero center the data (Hayes, 2013) which results in different "main effects" but the same interaction term. To show the equivalence of the interaction terms, we also must not standardize the results.

```
  ┌─ R code ─────────────────────────────────────────────────────────────┐
> mod <- setCor(SATQ ~ SATV*gender + ACT, data=sat.act, std =FALSE, plot=FALSE)
> mod
  └───────────────────────────────────────────────────────────────────────┘

Call: setCor(y = SATQ ~ SATV * gender + ACT, data = sat.act, std = FALSE,
    plot = FALSE)

Multiple Regression from raw data

 DV =  SATQ
              slope   se     t        p lower.ci upper.ci  VIF Vy.x
(Intercept) 610.19 3.06 199.50 0.0e+00   604.18   616.19 1.00 0.00
SATV          0.47 0.03  14.47 1.1e-41     0.41     0.54 1.46 0.30
gender      -35.08 6.40  -5.48 6.0e-08   -47.66   -22.51 1.00 0.02
ACT           7.72 0.77  10.05 2.8e-22     6.22     9.23 1.47 0.19
SATV*gender  -0.03 0.06  -0.47 6.4e-01    -0.14     0.08 1.01 0.00

Residual Standard Error =  80.91  with  695  degrees of freedom

 Multiple Regression
        R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2          p
SATQ 0.72 0.51 0.57 0.32        0.51     0.03   183.23   4 695 3.84e-107
```

```
  ┌─ R code ──────────────────────────────────────────────────────────────────┐
> mod0 <- setCor(SATQ ~ SATV*gender + ACT, data=sat.act, zero=FALSE, std=FALSE,plot=FALSE)
> mod0
  └────────────────────────────────────────────────────────────────────────────┘

Call: setCor(y = SATQ ~ SATV * gender + ACT, data = sat.act, std = FALSE,
    plot = FALSE, zero = FALSE)

Multiple Regression from raw data

 DV =  SATQ
              slope    se     t        p lower.ci upper.ci   VIF  Vy.x
(Intercept) 130.26 60.73   2.14 3.2e-02    11.02   249.50  1.00  0.00
SATV          0.52  0.10   5.20 2.7e-07     0.32     0.71 13.52  0.33
gender      -18.71 35.31  -0.53 6.0e-01   -88.03    50.61 30.44  0.01
ACT           7.72  0.77  10.05 2.8e-22     6.22     9.23  1.47  0.19
SATV*gender  -0.03  0.06  -0.47 6.4e-01    -0.14     0.08 41.50 -0.02

Residual Standard Error =  80.91  with  695  degrees of freedom

 Multiple Regression
        R   R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2          p
SATQ 0.72 0.51 0.42 0.17        0.51     0.03   183.23   4 695 3.84e-107

pdf
  2

pdf
```

Another example of moderated mediation

prot2

0.31

respappr   c = -2.81

1.12   c = 0.36

0.74   liking

0.04   c = 0.54

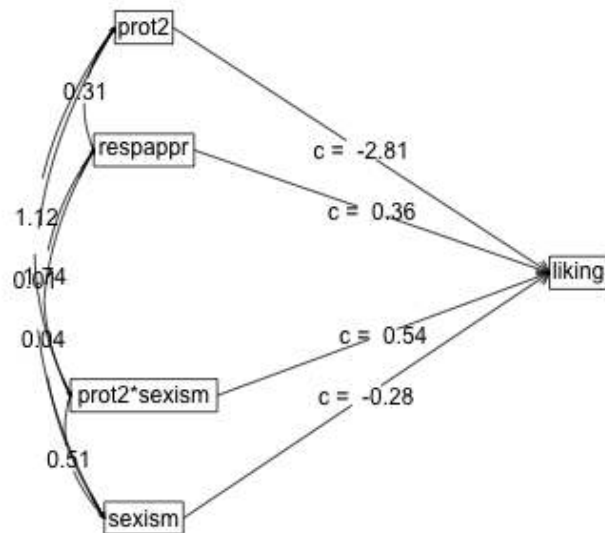prot2*sexism   c = -0.28

0.51

sexism

Figure 4: Showing a moderated multiple regression using `setCor`. The data are from the `sat.act` data set.

3.5 Plotting the interactions

To visualize the effect of zero (mean) centering, it is useful to plot the various elements that go into the linear model. `setCor` returns the product terms as well as the original data. I combine the two datasets to make it clearer. Note that the correlations of the centered SATQ, SATV, and gender with the uncentered are 1.0, but that the correlations with the product terms depend upon centering versus not. I drop the ACT variables from the figure for clarity (Figure 6).
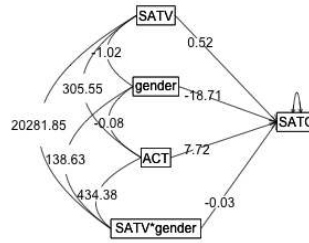
pdf
2
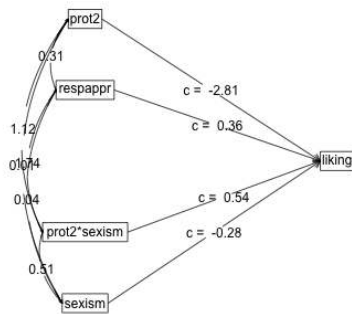
Figure 5: The difference between 0 and not 0 centering `setCor`. The data are from the `sat.act` data set. In both cases, the data are not standarized.

3.6 Comparisons to `lm`

The `setCor` function duplicates the functionality of the `lm` function for complete data, although `lm` does not zero center and `setCor` will (by default). In addition, `setCor` finds correlations based upon pair.wise deletion of missing data, while `lm` does case.wise deletion. We compare the `lm` and `setCor` results for complete data by setting the `use = "complete"` option.

```
R code
> lm(SATQ ~ SATV*gender + ACT, data=sat.act)


Call:
lm(formula = SATQ ~ SATV * gender + ACT, data = sat.act)

Coefficients:
(Intercept)          SATV        gender          ACT   SATV:gender
  138.52395       0.50280     -22.24995      7.71702      -0.01984
```

```
R code
> mod <- setCor(SATQ ~ SATV*gender + ACT, data=(sat.act), zero=FALSE, std=FALSE,use="complete")
> print(mod,digits=5)


Call: setCor(y = SATQ ~ SATV * gender + ACT, data = (sat.act), use = "complete",
    std = FALSE, zero = FALSE)

Multiple Regression from raw data

 DV =  SATQ
              slope        se        t          p  lower.ci  upper.ci       VIF      Vy.x
(Intercept) 138.72572 60.61339  2.28870 2.2396e-02  19.71841 257.73303   1.00000   0.00000
SATV          0.50280  0.09936  5.06050 5.3589e-07   0.30772   0.69788  13.43994   0.31739
gender      -22.24995 35.25783 -0.63106 5.2821e-01 -91.47457  46.97468  30.29663   0.01525
```
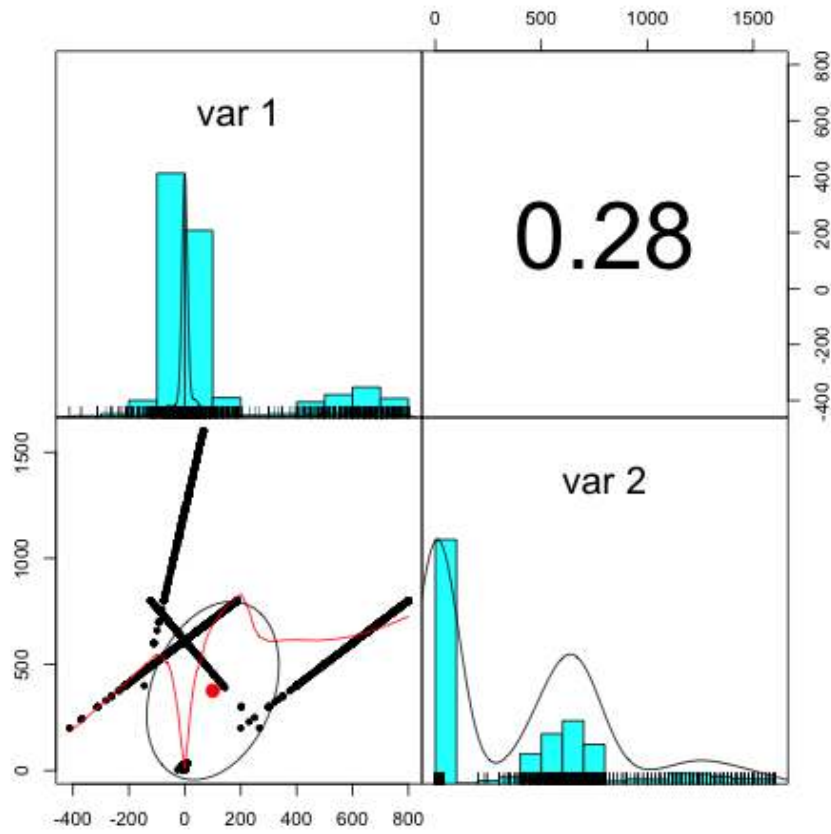
14

Figure 6: The effect of not mean centering versus mean centering on the product terms. The first four variables were not zero centered, the second four were.

```
ACT            7.71702  0.76977 10.02511 3.4878e-22   6.20567   9.22837  1.46678  0.18928
SATV*gender   -0.01984  0.05652 -0.35105 7.2566e-01  -0.13082   0.09114 41.25607 -0.01191

Residual Standard Error =  81.18031  with  695  degrees of freedom

 Multiple Regression
          R    R2      Ruw    R2uw Shrunken R2 SE of R2 overall F df1 df2          p
SATQ 0.71414 0.51 0.41706 0.17394     0.50718  0.02621  180.8401   4 695 3.93405e-106
```

4   Mediation using the `mediate` function

Mediation analysis is just linear regression reorganized slightly to show the direct effects of an X variable upon Y, partialling out the effect of a "mediator" (Figure 7). Although the statistical "significance" of the (c) path and the (c') path are both available from standard regression, the mediation effect (ab) is best found by boot strapping the regression model and displaying the empirical confidence intervals.
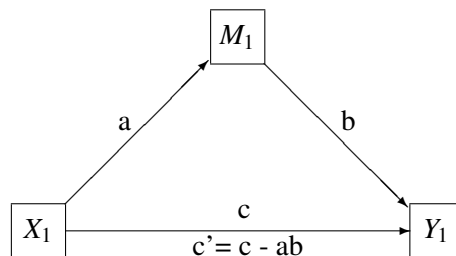
Figure 7: The classic mediation model. The Direct Path from X -> Y (c) is said to be mediated by the indirect path (a) to the mediator (X -> M) and (b) from the mediator to Y (M -> Y). The mediation effect is (ab).

A number of papers discuss how to test for the effect of mediation and there are some very popular 'macros' for SPSS and SAS to do so (Hayes, 2013; Preacher and Hayes, 2004; Preacher et al., 2007; Preacher, 2015). A useful discussion of mediation and moderation with sample data sets is found in Hayes (2013). More recently, the *processR* package (Moon, 2020) has been released with these data sets. Although these data used to be be available from http://www.afhayes.com/public/hayes2018data.zip

this now longer seems to be case.[1]. I use these for comparisons with the results in Hayes (2013). Four of these data sets are now included in the *psych* package with the kind permission of their authors: `Garcia` is from Garcia et al. (2010), and `Tal_Or` is from Tal-Or et al. (2010), The `Pollack` correlation matrix is taken from an article by **?**. The `globalWarm` data set is the `glbwarm` data set in the *processR* package and added to *psychTools* with the kind permission of the original author, Erik Nisbet.

To find the confidence intervals of the effect of mediation (the reduction between the c and c' paths, where c' = c - ab), I bootstrap the results by randomly sampling from the data with replacement (n.iter = 5000) times.

For these examples, the data files `Garcia` (Garcia et al., 2010) and `Tal_Or` (Tal-Or et al., 2010) are included in the `psych` package. The `estrss` data set and `glbwarm` were originally downloaded from the Hayes (2013) data seta and stored in a local directory on my computer. They are now available from the *processR* package Moon (2020).

The syntax is that $y \sim x + (m)$ where m is the mediating variable. By default the output is to two decimals, as is the graphic output. This can be increased by returning the output to an object and then printing that object with the desired number of decimals.

4.1 Simple mediation

The first example (Hayes, 2013, mod.4.5) is taken from (Tal-Or et al., 2010) and examines the mediating effect of "Presumed Media Influence" (pmi) on the intention to act (reaction) based upon the importance of a message (import). The data are in the `Tal_Or` data set in *psych* (with the kind permission of Nurit Tal-Or, Jonanathan Cohen, Yariv Tasfati, and Albert Gunther). In the Hayes (2013) book, this is the `pmi` data set.

```
 R code
> data(Tal.Or)
> psych::describe(Tal_Or)   #descriptive statistics
```

```
          vars   n   mean    sd median trimmed  mad min max range  skew kurtosis   se
cond         1 123   0.47 0.50   0.00    0.46 0.00   0   1     1  0.11    -2.00 0.05
pmi          2 123   5.60 1.32   6.00    5.78 1.48   1   7     6 -1.17     1.30 0.12
import       3 123   4.20 1.74   4.00    4.26 1.48   1   7     6 -0.26    -0.89 0.16
reaction     4 123   3.48 1.55   3.25    3.44 1.85   1   7     6  0.21    -0.90 0.14
gender       5 123   1.65 0.48   2.00    1.69 0.00   1   2     1 -0.62    -1.62 0.04
age          6 123  24.63 5.80  24.00   23.76 1.48  18  61    43  4.71    24.76 0.52
```

```
 R code
> mod4.4 <- mediate(reaction ~ cond + (pmi), data =Tal_Or)
> mod4.4
```

```
Mediation/Moderation Analysis
Call: mediate(y = reaction ~ cond + (pmi), data = Tal_Or)
```

---

[1]The Hayes data sets (2018) do not correspond exactly with those from the 2013 book. Those data files were at http://www.afhayes.com/public/hayes2013data.zip.

```
The DV (Y) was  reaction . The IV (X) was  cond . The mediating variable(s) =  pmi .

Total effect(c) of  cond  on  reaction  =  0.5    S.E. =  0.28  t  =  1.79  df= 121    with p =  0.077
Direct effect (c') of  cond  on  reaction  removing  pmi  =  0.25    S.E. =  0.26  t  =  0.99  df= 120    with
Indirect effect (ab) of  cond  on  reaction  through  pmi    =  0.24
Mean bootstrapped indirect effect =  0.25  with standard error =  0.13  Lower CI =  0.01     Upper CI =  0.52
R = 0.45 R2 = 0.21   F = 15.56 on 2 and 120 DF   p-value:  1.31e-08

 To see the longer output, specify short = FALSE in the print statement or ask for the summary
```
┌────────┐
│ R code │
└────────┘
```
> #print(mod4.4, digits = 4) # in order to get the precision of the Hayes (2013) p 99 example

pdf
  2
```
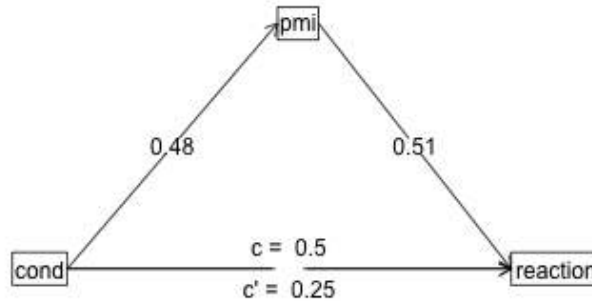
Mediation model

Figure 8: A simple mediation model (Hayes, 2013, p 99) with data derived from Tal-Or et al. (2010). The effect of a salience manipulation (cond) on the intention to buy a product (reaction) is mediated through the presumed media influence (pmi).

A second example from (Hayes, 2013) is an example of moderated mediated effect. The data are from (Garcia et al., 2010) and report on the effect of protest on reactions to a case of sexual discrimination.

```
R code
> data(GSBE)  #alias to Garcia data set
> #compare two models  (bootstrapping n.iter set to 50 for speed
> # 1) mean center the variables prior to taking product terms
> mod1 <- mediate(respappr ~ prot2 * sexism +(sexism),data=Garcia,n.iter=50
+ ,main="Moderated mediation (mean centered)")
> # 2) do not mean center
> mod2 <- mediate(respappr ~ prot2 * sexism +(sexism),data=Garcia,zero=FALSE, n.iter=50,
+     main="Moderated  mediation (not centered")
> summary(mod1)
```

```
Call: mediate(y = respappr ~ prot2 * sexism + (sexism), data = Garcia,
    n.iter = 50, main = "Moderated mediation (mean centered)")

Direct effect estimates (traditional regression)    (c') X + M on Y
            respappr   se     t   df     Prob
Intercept      -0.01 0.10 -0.12 125 9.07e-01
prot2           1.46 0.22  6.73 125 5.52e-10
prot2*sexism    0.81 0.28  2.87 125 4.78e-03
sexism          0.02 0.13  0.18 125 8.56e-01

R = 0.54 R2 = 0.3   F = 17.53 on 3 and 125 DF   p-value:  1.46e-09

 Total effect estimates (c) (X on Y)
            respappr   se     t   df     Prob
Intercept      -0.01 0.10 -0.12 126 9.06e-01
prot2           1.46 0.22  6.77 126 4.43e-10
prot2*sexism    0.81 0.28  2.89 126 4.49e-03

 'a'  effect estimates (X on M)
            sexism   se     t  df  Prob
Intercept     0.00 0.07 -0.02 126 0.986
prot2         0.07 0.15  0.47 126 0.642
prot2*sexism  0.09 0.19  0.44 126 0.661

 'b'  effect estimates (M on Y controlling for X)
       respappr   se    t  df  Prob
sexism     0.02 0.13 0.18 125 0.856

 'ab'  effect estimates (through all  mediators)
            respappr boot   sd lower upper
prot2              0 0.00 0.02 -0.07  0.04
prot2*sexism       0 0.01 0.04 -0.07  0.04
```

```
R code
> summary(mod2)
```

```
Call: mediate(y = respappr ~ prot2 * sexism + (sexism), data = Garcia,
    n.iter = 50, zero = FALSE, main = "Moderated  mediation (not centered")

Direct effect estimates (traditional regression)    (c') X + M on Y
            respappr   se     t   df     Prob
Intercept       6.57 1.21  5.43 125 2.83e-07
prot2          -2.69 1.45 -1.85 125 6.65e-02
prot2*sexism    0.81 0.28  2.87 125 4.78e-03
```

19

```
sexism          -0.53 0.24 -2.24 125 2.67e-02

R = 0.54 R2 = 0.3   F = 17.53 on 3 and 125 DF   p-value:  1.46e-09

 Total effect estimates (c) (X on Y)
            respappr   se      t   df      Prob
Intercept       3.88 0.18 21.39 126 9.14e-44
prot2           0.00 0.84  0.00 126 9.96e-01
prot2*sexism    0.28 0.16  1.79 126 7.56e-02

 'a'  effect estimates (X on M)
            sexism   se      t   df      Prob
Intercept     5.07 0.07  75.12 126 1.69e-106
prot2        -5.07 0.31 -16.33 126  6.81e-33
prot2*sexism  1.00 0.06  17.15 126  9.41e-35

 'b'  effect estimates (M on Y controlling for X)
      respappr   se      t   df   Prob
sexism    -0.53 0.24 -2.24 125 0.0267

 'ab'  effect estimates (through all  mediators)
            respappr  boot    sd lower upper
prot2           2.68  2.96 1.28  0.87  5.45
prot2*sexism   -0.53 -0.59 0.25  0.87  5.45
```

4.2   Multiple mediators

It is trivial to show the effect of multiple mediators. I do this by adding the second (or third) mediator into our equation. I use the `Tal_Or` data set (Tal-Or et al., 2010) again. I show the graphical representation in Figure 9.

```
┌─ R code ─────────────────────────────────────────────────────────────
> mod5.4 <- mediate(reaction ~ cond + (import) + (pmi), data = Tal_Or)
> print(mod5.4, digits=4)  #to compare with Hayes
└──────────────────────────────────────────────────────────────────────

Mediation/Moderation Analysis
Call: mediate(y = reaction ~ cond + (import) + (pmi), data = Tal_Or)

The DV (Y) was  reaction . The IV (X) was  cond . The mediating variable(s) =  import pmi .

Total effect(c) of  cond  on  reaction  =  0.4957   S.E. =  0.2775  t  =  1.786  df=  121   with p =  0.0766
Direct effect (c') of  cond  on  reaction  removing  import pmi  =  0.1034   S.E. =  0.2391  t  =  0.4324  d
Indirect effect (ab) of  cond  on  reaction  through  import pmi   =  0.3923
Mean bootstrapped indirect effect =  0.3953  with standard error =  0.1654  Lower CI =  0.0893    Upper CI =
R = 0.5702 R2 = 0.3251   F = 19.1118 on 3 and 119 DF   p-value:  3.6636e-12

 To see the longer output, specify short = FALSE in the print statement or ask for the summary
┌─ R code ─────────────────────────────────────────────────────────────
>
└──────────────────────────────────────────────────────────────────────

pdf
  2
```
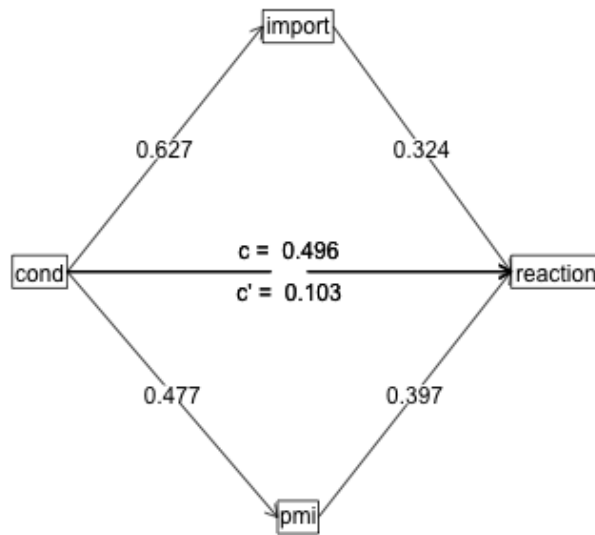
Figure 9: A mediation model with two mediators (Hayes, 2013, p 131). The data are data derived from Tal-Or et al. (2010). The effect of a salience manipulation (cond) on the intention to buy a product (reaction) is mediated through the presumed media influence (pmi) and importance of the message (import).

4.3   Serial mediators

The example from Hayes (2013) for two mediators, where one effects the second, is a bit more complicated and currently can be done by combining two separate analyses. The first is just model 5.4, the second is the effect of cond on pmi mediated by import.

Combining the two results leads to the output found on (Hayes, 2013, page 153).

```
pdf
  2
```

R code

```
> #model 5.4 + mod5.7 is the two chained mediator model
> mod5.7 <- mediate(pmi ~ cond + (import) , data = Tal_Or)
> summary(mod5.7, digits=4)
```

```
Call: mediate(y = pmi ~ cond + (import), data = Tal_Or)

Direct effect estimates (traditional regression)    (c') X + M on Y
            pmi     se       t   df      Prob
Intercept 4.6104 0.3057 15.0836 120 1.7543e-29
cond      0.3536 0.2325  1.5207 120 1.3096e-01
import    0.1961 0.0671  2.9228 120 4.1467e-03

R = 0.3114 R2 = 0.097   F = 6.4428 on 2 and 120 DF   p-value:  0.0021989

 Total effect estimates (c)  (X on Y)
            pmi     se       t   df      Prob
Intercept 5.3769 0.1618 33.2222 121 1.1593e-62
cond      0.4765 0.2357  2.0218 121 4.5401e-02

 'a'  effect estimates  (X on M)
         import     se       t   df      Prob
Intercept 3.9077 0.2127 18.3704 121 8.3936e-37
cond      0.6268 0.3098  2.0234 121 4.5235e-02

 'b'  effect estimates  (M on Y controlling for X)
         pmi     se      t   df      Prob
import 0.1961 0.0671 2.9228 120 0.0041467

 'ab'  effect estimates  (through all  mediators)
       pmi   boot     sd  lower  upper
cond 0.1229 0.1273 0.0855 -0.002 0.3253
```

4.4   Single mediators, multiple covariates

The `Pollack` data set (Pollack et al., 2012) is used as an example of multiple covariates (included in *psychTools* as a correlation matrix). The raw data are available from the *processR* package as `estress`. Confidence in executive decision making ("Entrepeneurial self-effiicacy), gender (sex), and length of time in business (tenure) are used as covariates. There are two ways of doing this: enter them as predictors of the criterion or to partial them out. The first approach estimates their effects, the second just removes them.

R code

```
> lowerMat(Pollack)
```

```
                 sex   age   tenur slf.f cmptn scl.t ecnm. dprss wthdr
sex              1.00
age              0.07  1.00
tenure           0.03  0.32  1.00
self.efficacy   -0.02 -0.09 -0.06  1.00
competence       0.08  0.01  0.02  0.22  1.00
social.ties      0.07 -0.06  0.01  0.19  0.13  1.00
economic.stress -0.15  0.09  0.07 -0.16 -0.09 -0.07  1.00
depression      -0.05 -0.02 -0.07 -0.25  0.06 -0.05  0.34  1.00
withdrawal      -0.03 -0.05 -0.04 -0.24 -0.09  0.01  0.06  0.42  1.00
```

```r
> mod6.2 <- mediate(withdrawal ~ economic.stress + self.efficacy + sex + tenure + (depression),
+       data=Pollack, n.obs=262)
> summary(mod6.2)
```

```
Call: mediate(y = withdrawal ~ economic.stress + self.efficacy + sex +
    tenure + (depression), data = Pollack, n.obs = 262)

Direct effect estimates (traditional regression)    (c') X + M on Y
                withdrawal   se     t   df     Prob
Intercept            0.00 0.06  0.00 256 1.00e+00
economic.stress     -0.11 0.06 -1.82 256 6.99e-02
self.efficacy       -0.15 0.06 -2.67 256 8.01e-03
sex                 -0.03 0.06 -0.50 256 6.15e-01
tenure              -0.01 0.06 -0.21 256 8.37e-01
depression           0.42 0.06  6.83 256 6.05e-11

R = 0.45 R2 = 0.21   F = 13.35 on 5 and 256 DF   p-value:  1.45e-11

 Total effect estimates (c)  (X on Y)
                withdrawal   se     t   df     Prob
Intercept            0.00 0.06  0.00 257 1.000000
economic.stress      0.02 0.06  0.34 257 0.737000
self.efficacy       -0.24 0.06 -3.92 257 0.000113
sex                 -0.03 0.06 -0.49 257 0.624000
tenure              -0.05 0.06 -0.91 257 0.366000

 'a'  effect estimates (X on M)
                depression   se     t   df     Prob
Intercept            0.00 0.06  0.00 257 1.00e+00
economic.stress      0.31 0.06  5.36 257 1.88e-07
self.efficacy       -0.21 0.06 -3.56 257 4.36e-04
sex                  0.00 0.06 -0.07 257 9.46e-01
tenure              -0.10 0.06 -1.82 257 6.98e-02

 'b'  effect estimates (M on Y controlling for X)
           withdrawal   se    t   df     Prob
depression       0.42 0.06 6.83 256 6.05e-11

 'ab'  effect estimates (through all  mediators)
                withdrawal  boot    sd lower upper
economic.stress       0.13  0.08 0.02  0.04  0.13
self.efficacy        -0.09 -0.08 0.02  0.04  0.13
sex                   0.00  0.01 0.02  0.04  0.13
tenure               -0.04 -0.05 0.02  0.04  0.13

pdf
  2
```

The graphical output (Figure 10) looks a bit more complicated than the figure in (Hayes, 2013, p 177) because I are showing the covariates as causal paths.
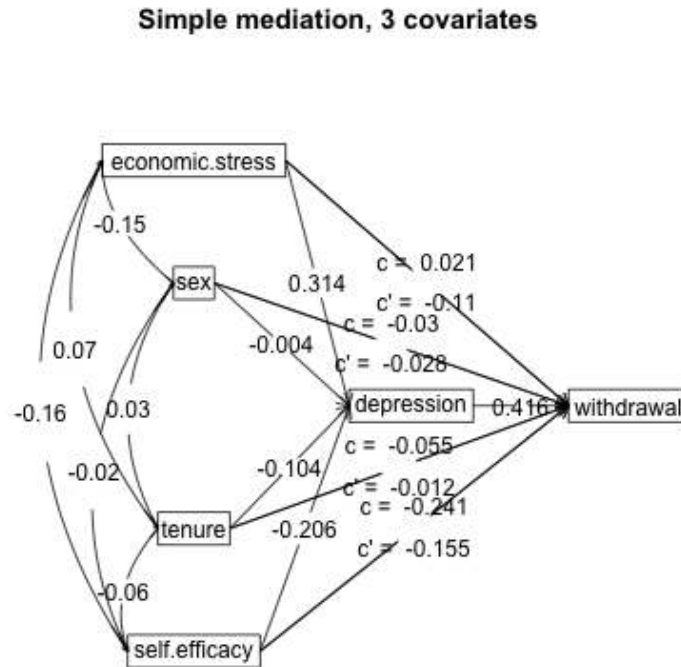
Simple mediation, 3 covariates

Figure 10: A mediation model with three covariates (Hayes, 2013, p 177). Compare this to the solution in which they are partialled out. (Figure 11).

4.5 Single predictor, single criterion, multiple covariates

An alternative way to display the previous results is to remove the three covariates from the mediation model. I do this by partialling out the covariates. This is represented in the `mediate` code by a negative sign. (Figure 11)

```
R code
> mod6.2a <- mediate(withdrawal ~ economic.stress -self.efficacy - sex - tenure + (depression),
+       data=Pollack, n.obs=262)
> summary(mod6.2a)
```

```
Call: mediate(y = withdrawal ~ economic.stress - self.efficacy - sex -
    tenure + (depression), data = Pollack, n.obs = 262)

Direct effect estimates (traditional regression)    (c') X + M on Y
                withdrawal*   se    t    df     Prob
Intercept             0.00  0.06  0.00  256  1.00e+00
economic.stress      -0.11  0.06 -1.80  256  7.23e-02
depression            0.42  0.06  6.78  256  8.50e-11

R = 0.39 R2 = 0.15   F = 23.41 on 2 and 256 DF   p-value:  4.6e-10

 Total effect estimates (c) (X on Y)
                withdrawal*   se    t   df  Prob
Intercept             0.00  0.06  0.00 257 1.000
economic.stress       0.02  0.06  0.34 257 0.737

 'a'  effect estimates (X on M)
                depression   se    t   df     Prob
Intercept             0.00  0.06  0.00 257  1.00e+00
economic.stress       0.31  0.06  5.36 257  1.88e-07

 'b'  effect estimates (M on Y controlling for X)
            withdrawal*   se    t   df     Prob
depression        0.42  0.06  6.83 256  6.05e-11

 'ab'  effect estimates (through all  mediators)
                withdrawal* boot   sd lower upper
economic.stress       0.13 0.15 0.03  0.09  0.22

pdf
  2
```

4.6   Multiple predictors, single criterion

It is straightforward to use multiple predictors see (Hayes, 2013, p196) and in fact I did so in
the previous example where the predictors were treated as *covariates*. mediate also allows for
multiple criteria.

5   Mediation and moderation

We already saw how to do moderation in the discussion of setCor. Combining the concepts of
mediation with moderation is done in mediate. That is, I can find the linear model of prod-
uct terms as they are associated with dependent variables and regressed on the mediating vari-
ables.

The Garcia data set (Garcia et al., 2010) can be used for an example of moderation. (This was
taken from (Hayes, 2013) but is used with kind permission of Donna M. Garcia, Michael T. Schmitt,
Nyla R. Branscombe, and Naomi Ellemers.) Just as setCor and lm will find the interaction term
by forming a product, so will mediate. Notice that by default, setCor reports zero centered
and standardized regressions, mediate reports zero centered but not standardized regressions, and
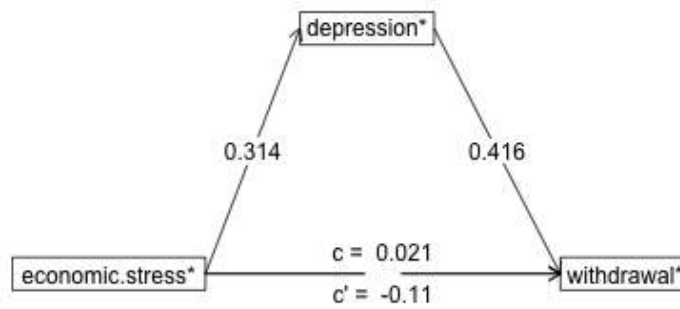
Simple mediation, 3 covariates (partialled out)

Figure 11: Show the mediation model from Figure 10 with the covariates (ese, sex, tenure) removed.

some of the examples from Hayes (2013) do not zero center the data. Thus, I specify zero=FALSE to get the Hayes (2013) results.

It is important to note that the `protest` data set discussed here is from the 2013 examples and not the more recent 2018 examples available from afhayes.com. The 2013 data have a dichotomous protest variable, while the 2018 data set has three levels for the protest variable. The `Garcia` data set is composed of the 2018 data set with the addition of a dichotomous variable (prot2) to match the 2013 exampes.

We consider how the interaction of sexism with protest affects the mediation effect of sexism (Hayes, 2013, p 362), I contrast the `lm`, `setCor` and `mediate` approaches. For reasons to be discussed in the next section, I do not zero center the variables. The graphic output is in Figure 12 and the output is below. For comparison purposes, I show the results from the `lm` as well as `setCor` and `mediate`.

```
R code
> summary(lm(respappr ~ prot2 * sexism,data = Garcia))  #show the lm results for comparison


Call:
lm(formula = respappr ~ prot2 * sexism, data = Garcia)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4984 -0.7540  0.0801  0.8301  3.1853

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    6.5667     1.2095   5.429 2.83e-07 ***
prot2         -2.6866     1.4515  -1.851  0.06654 .
sexism        -0.5290     0.2359  -2.243  0.02668 *
prot2:sexism   0.8100     0.2819   2.873  0.00478 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.144 on 125 degrees of freedom
Multiple R-squared:  0.2962,        Adjusted R-squared:  0.2793
F-statistic: 17.53 on 3 and 125 DF,  p-value: 1.456e-09
```

```
R code
> #show the setCor analysis
> setCor(respappr ~ prot2* sexism ,data=Garcia,zero=FALSE,main="Moderation",std=FALSE)


Call: setCor(y = respappr ~ prot2 * sexism, data = Garcia, std = FALSE,
    main = "Moderation", zero = FALSE)

Multiple Regression from raw data

 DV =  respappr
             slope   se     t        p lower.ci upper.ci   VIF   Vy.x
(Intercept)   6.57 1.21  5.43 2.8e-07     4.17     8.96  1.00   0.00
prot2        -2.69 1.45 -1.85 6.7e-02    -5.56     0.19 44.99  -0.47
sexism       -0.53 0.24 -2.24 2.7e-02    -1.00    -0.06  3.34  -0.01
prot2*sexism  0.81 0.28  2.87 4.8e-03     0.25     1.37 48.14   0.77

Residual Standard Error =  1.14  with  125  degrees of freedom
```

27

```
 Multiple Regression
            R  R2  Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2         p
respappr 0.54 0.3 0.41 0.17          0.28     0.06     17.53   3 125 1.46e-09
```

```
┌─ R code ─────────────────────────────────────────────────────────────────────
│ > #then show the mediate results
│ >
│ > modgarcia <-mediate(respappr ~ prot2 * sexism +(sexism),data=Garcia,zero=FALSE,main="Moderated mediation")
│ > summary(modgarcia)
└──────────────────────────────────────────────────────────────────────────────
```

```
Call: mediate(y = respappr ~ prot2 * sexism + (sexism), data = Garcia,
    zero = FALSE, main = "Moderated mediation")

Direct effect estimates (traditional regression)    (c') X + M on Y
            respappr   se     t   df     Prob
Intercept       6.57 1.21   5.43 125 2.83e-07
prot2          -2.69 1.45  -1.85 125 6.65e-02
prot2*sexism    0.81 0.28   2.87 125 4.78e-03
sexism         -0.53 0.24  -2.24 125 2.67e-02

R = 0.54 R2 = 0.3   F = 17.53 on 3 and 125 DF   p-value:  1.46e-09


 Total effect estimates (c) (X on Y)
            respappr   se     t   df     Prob
Intercept       3.88 0.18 21.39 126 9.14e-44
prot2           0.00 0.84  0.00 126 9.96e-01
prot2*sexism    0.28 0.16  1.79 126 7.56e-02


 'a'  effect estimates (X on M)
            sexism   se     t   df     Prob
Intercept     5.07 0.07  75.12 126 1.69e-106
prot2        -5.07 0.31 -16.33 126  6.81e-33
prot2*sexism  1.00 0.06  17.15 126  9.41e-35


 'b'  effect estimates (M on Y controlling for X)
      respappr   se     t   df   Prob
sexism   -0.53 0.24 -2.24 125 0.0267


 'ab'  effect estimates (through all  mediators)
            respappr  boot    sd lower upper
prot2           2.68  2.70  1.60 -0.59  5.67
prot2*sexism   -0.53 -0.53  0.32 -0.59  5.67
```

```
┌─ R code ─────────────────────────────────────────────────────────────────────
│ >
└──────────────────────────────────────────────────────────────────────────────
```

```
pdf
  2
```

5.1   To center or not to center, that is the question

We have discussed the difference between zero centering and not zero centering. Although Hayes
(2013) seems to prefer not centering, some of his examples are in fact centered. So, when we
examine Table 8.2 and try to replicate the regression, we need to zero center the data.

With the global warming data from Hayes (2013), the default (uncentered) regression does not

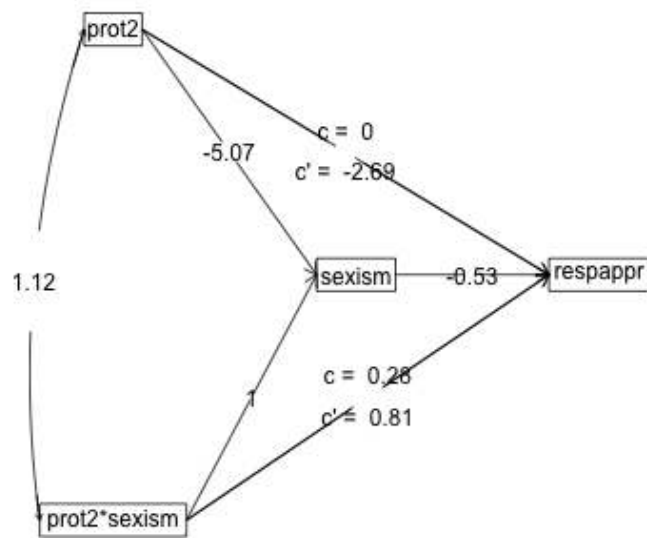**An example of moderated mediation**

Figure 12: Moderated mediation from (Hayes, 2013, p 362). The data are from Garcia et al. (2010).

reproduce his Table, but zero centering does. To this in `lm` requires two steps, but we can do this in `setCor` with the zero=TRUE or zero=FALSE option.

```
─ R code ─
> lm(govact ~ age * negemot + posemot + ideology + sex, data=globalWarm)
```

```
Call:
lm(formula = govact ~ age * negemot + posemot + ideology + sex,
    data = globalWarm)

Coefficients:
(Intercept)          age      negemot      posemot     ideology          sex  age:negemot
   5.173849    -0.023879     0.119583    -0.021419    -0.211515    -0.011191     0.006331
```

```
─ R code ─
> # but  zero center and try again
> glbwarmc <-data.frame(scale(globalWarm,scale=FALSE))
> lm(govact ~ age * negemot + posemot + ideology + sex, data=glbwarmc)
```

```
Call:
lm(formula = govact ~ age * negemot + posemot + ideology + sex,
    data = glbwarmc)

Coefficients:
(Intercept)          age      negemot      posemot     ideology          sex  age:negemot
   0.008979    -0.001354     0.433184    -0.021419    -0.211515    -0.011191     0.006331
```

```
─ R code ─
> mod.glb <- setCor(govact ~ age * negemot + posemot + ideology + sex, data=globalWarm,zero=FALSE,std=FALSE)
> print(mod.glb,digits=6)
```

```
Call: setCor(y = govact ~ age * negemot + posemot + ideology + sex,
    data = globalWarm, std = FALSE, zero = FALSE)

Multiple Regression from raw data

 DV =  govact
                slope         se          t            p   lower.ci   upper.ci         VIF        Vy.x
(Intercept)  5.173849   0.338451  15.286838  1.58157e-46   4.509502   5.838197    1.000000    0.000000
age         -0.023879   0.005980  -3.992944  7.12038e-05  -0.035618  -0.012140    6.949401    0.027844
negemot      0.119583   0.082535   1.448881  1.47759e-01  -0.042425   0.281591   11.594520    0.077620
posemot     -0.021419   0.027904  -0.767597  4.42951e-01  -0.076193   0.033354    1.028663   -0.000912
ideology    -0.211515   0.026833  -7.882678  1.03603e-14  -0.264185  -0.158845    1.198910    0.098323
sex         -0.011191   0.076003  -0.147240  8.82979e-01  -0.160378   0.137997    1.052907    0.000406
age*negemot  0.006331   0.001543   4.103542  4.48155e-05   0.003302   0.009359   16.455422    0.197526

Residual Standard Error =  1.056984  with  808  degrees of freedom

 Multiple Regression
              R       R2      Ruw      R2uw Shrunken R2 SE of R2 overall F df1 df2          p
govact 0.633093 0.400806 0.14033 0.019693    0.396357 0.026299  90.07983   6 808 1.824604e-86
```

```
─ R code ─
> mod.glb0 <- setCor(govact ~ age * negemot + posemot + ideology + sex, data=globalWarm,std=FALSE)
> print(mod.glb0,digits=6)
```

```
Call: setCor(y = govact ~ age * negemot + posemot + ideology + sex,
    data = globalWarm, std = FALSE)

Multiple Regression from raw data
```

```
 DV =  govact
                slope       se         t          p  lower.ci  upper.ci      VIF      Vy.x
(Intercept)  4.595973 0.037089 123.916910 0.00000e+00  4.523171  4.668776 1.000000  0.000000
age         -0.001354 0.002348  -0.576864 5.64192e-01 -0.005963  0.003254 1.071058  0.001579
negemot      0.433184 0.026243  16.506679 5.75775e-53  0.381671  0.484696 1.172207  0.281175
posemot     -0.021419 0.027904  -0.767597 4.42951e-01 -0.076193  0.033354 1.028663 -0.000912
ideology    -0.211515 0.026833  -7.882678 1.03603e-14 -0.264185 -0.158845 1.198910  0.098323
sex         -0.011191 0.076003  -0.147240 8.82979e-01 -0.160378  0.137997 1.052907  0.000406
age*negemot  0.006331 0.001543   4.103542 4.48155e-05  0.003302  0.009359 1.014744  0.020236

Residual Standard Error =  1.056984  with  808  degrees of freedom

 Multiple Regression
               R        R2       Ruw     R2uw Shrunken R2 SE of R2 overall F df1 df2            p
govact 0.633093 0.400806 0.355865 0.12664    0.396357 0.026299  90.07983   6 808 1.824604e-86
```

So, when we do the mediated moderation model, we need to use the zero centered option to match the Hayes (2013) results from Figure 8.5.

```
┌─ R code ─────────────────────────────────────────────────────────────────────────────────┐
> #by default, mediate zero centers before finding the products
>  mod.glb <- mediate(govact ~ age * negemot + posemot + ideology + sex + (age), data=globalWarm,zero=TRUE)
> summary(mod.glb,digits=4)
└────────────────────────────────────────────────────────────────────────────────────────────┘


Call: mediate(y = govact ~ age * negemot + posemot + ideology + sex +
    (age), data = globalWarm, zero = TRUE)

Direct effect estimates (traditional regression)    (c') X + M on Y
             govact      se       t   df       Prob
Intercept    0.0090  0.0371   0.2421  808 8.0876e-01
negemot      0.4332  0.0262  16.5067  808 5.7578e-53
posemot     -0.0214  0.0279  -0.7676  808 4.4295e-01
ideology    -0.2115  0.0268  -7.8827  808 1.0360e-14
sex         -0.0112  0.0760  -0.1472  808 8.8298e-01
age*negemot  0.0063  0.0015   4.1035  808 4.4816e-05
age         -0.0014  0.0023  -0.5769  808 5.6419e-01

R = 0.6331 R2 = 0.4008   F = 90.0798 on 6 and 808 DF   p-value:  1.8246e-86


 Total effect estimates (c) (X on Y)
             govact      se       t   df       Prob
Intercept    0.0090  0.0371   0.2420  809 8.0881e-01
negemot      0.4328  0.0262  16.5043  809 5.8181e-53
posemot     -0.0220  0.0279  -0.7890  809 4.3036e-01
ideology    -0.2145  0.0263  -8.1510  809 1.3690e-15
sex         -0.0173  0.0752  -0.2304  809 8.1783e-01
age*negemot  0.0063  0.0015   4.1025  809 4.4999e-05


 'a'  effect estimates (X on M)
               age      se      t   df       Prob
Intercept    0.0044  0.5554  0.0079  809 9.9366e-01
negemot      0.2757  0.3929  0.7017  809 4.8305e-01
posemot      0.4232  0.4176  1.0135  809 3.1112e-01
ideology     2.2079  0.3943  5.6002  809 2.9334e-08
sex          4.5345  1.1269  4.0238  809 6.2643e-05
age*negemot  0.0031  0.0231  0.1346  809 8.9294e-01


 'b'  effect estimates (M on Y controlling for X)
```

31

```
     govact      se       t df    Prob
age -0.0014 0.0023 -0.5769 808 0.56419


 'ab'  effect estimates (through all  mediators)
             govact    boot      sd   lower   upper
negemot      -0.0004 -0.0004 0.0011 -0.0033 0.0016
posemot      -0.0006 -0.0006 0.0014 -0.0033 0.0016
ideology     -0.0030 -0.0030 0.0050 -0.0033 0.0016
sex          -0.0061 -0.0061 0.0104 -0.0033 0.0016
age*negemot   0.0000  0.0000 0.0001 -0.0033 0.0016
```

Compare this output to that of Table 8.2 and Figure 8.5 (p 258 - 259).

5.2   Another example of moderated medation

The `Garcia` data set (`protest` in Hayes (2013)) is another example of a moderated analysis. We can use either `setCor` or `mediate` to examine this data set. The defaults for these two differ, in that `setCor` assumes we want to zero center *and* standardize, while `mediate` defaults to not standardizing but also defaults to zero (mean) centering. Note that in the next examples we specify we do not want to standardize nor to mean center.

---- R code ----
```
> psych::describe(Garcia)


          vars   n mean   sd median trimmed  mad  min max range  skew kurtosis   se
protest      1 129 1.03 0.82   1.00    1.04 1.48 0.00   2  2.00 -0.06    -1.52 0.07
sexism       2 129 5.12 0.78   5.12    5.10 0.74 2.87   7  4.13  0.12    -0.32 0.07
anger        3 129 2.12 1.66   1.00    1.84 0.00 1.00   7  6.00  1.29     0.26 0.15
liking       4 129 5.64 1.05   5.83    5.73 0.99 1.00   7  6.00 -1.15     2.48 0.09
respappr     5 129 4.87 1.35   5.25    4.98 1.11 1.50   7  5.50 -0.75    -0.18 0.12
prot2        6 129 0.68 0.47   1.00    0.72 0.00 0.00   1  1.00 -0.77    -1.41 0.04
```
---- R code ----
```
> lm(liking ~ prot2* sexism + respappr, data=Garcia)


Call:
lm(formula = liking ~ prot2 * sexism + respappr, data = Garcia)


Coefficients:
 (Intercept)          prot2        sexism       respappr  prot2:sexism
      5.3471        -2.8075       -0.2824         0.3593        0.5426
```
---- R code ----
```
> setCor(liking ~ prot2* sexism + respappr, data = Garcia, zero=FALSE,std=FALSE)


Call: setCor(y = liking ~ prot2 * sexism + respappr, data = Garcia,
    std = FALSE, zero = FALSE)


Multiple Regression from raw data


 DV =  liking
             slope   se     t       p lower.ci upper.ci   VIF   Vy.x
(Intercept)   5.35 1.06  5.04 1.6e-06     3.25     7.45  1.00   0.00
prot2        -2.81 1.16 -2.42 1.7e-02    -5.10    -0.51 46.22  -0.27
sexism       -0.28 0.19 -1.49 1.4e-01    -0.66     0.09  3.47  -0.02
respappr      0.36 0.07  5.09 1.3e-06     0.22     0.50  1.42   0.23
prot2*sexism  0.54 0.23  2.36 2.0e-02     0.09     1.00 51.32   0.34
```

32

```
Residual Standard Error =  0.9  with  124  degrees of freedom

 Multiple Regression
          R    R2   Ruw R2uw Shrunken R2 SE of R2 overall F df1 df2        p
liking 0.53 0.28 0.34 0.12        0.26     0.06      12.26   4 124 1.99e-08
```

R code

```
> mod7.4 <- mediate(liking ~ prot2 * sexism +respappr, data = Garcia, zero=FALSE)
> summary(mod7.4)
```

```
Call: mediate(y = liking ~ prot2 * sexism + respappr, data = Garcia,
    zero = FALSE)

No mediator specified leads to traditional regression
            liking   se     t  df     Prob
Intercept     5.35 1.06  5.04 124 1.60e-06
prot2        -2.81 1.16 -2.42 124 1.70e-02
sexism       -0.28 0.19 -1.49 124 1.39e-01
respappr      0.36 0.07  5.09 124 1.28e-06
prot2*sexism  0.54 0.23  2.36 124 1.97e-02

R = 0.53 R2 = 0.28   F = 12.26 on 4 and 124 DF   p-value:  1.99e-08

pdf
  2
```

5.3   Graphic Displays of Interactions

In order to graphically display interactions, particularly if one of the variable is categorical, we can plot separate regression lines for each value of the categorical variable. We do this for the `Garcia` data set to show the interaction of protest with sexism. (see Figure 14). This is just an example of how to use Core-R to do graphics and is not a feature of *psych*.

```
Garcia$protest: 0
NULL
------------------------------------------------------------------------------------------------------------------------
Garcia$protest: 1
NULL
------------------------------------------------------------------------------------------------------------------------
Garcia$protest: 2
NULL

pdf
  2
```

6   Partial Correlations

Although not strickly speaking part of mediation or moderation, the use of *partial correlations* can be addressed here.

Another example of moderated mediation

prot2

0.31

respappr c = -2.81

1.12 c = 0.36

01.74

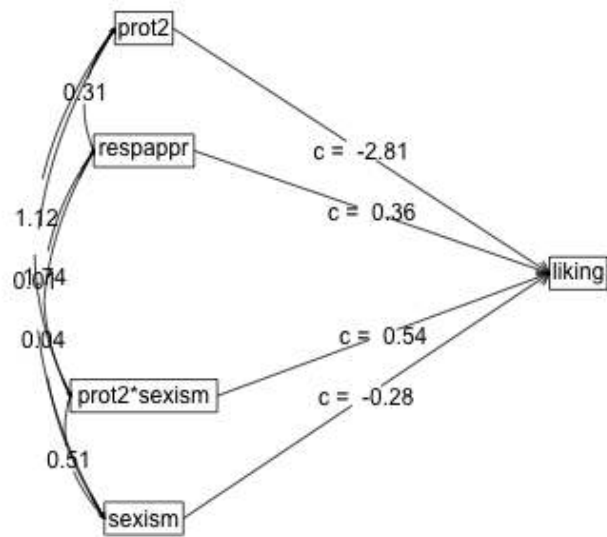0.04 liking

c = 0.54

prot2*sexism c = -0.28

0.51

sexism

Figure 13: A simple moderated regression analysis of the `protest` data set. The data were not zero centered. This shows the strength of the three regressions. Figure 14 shows the actual data and the three regression lines.
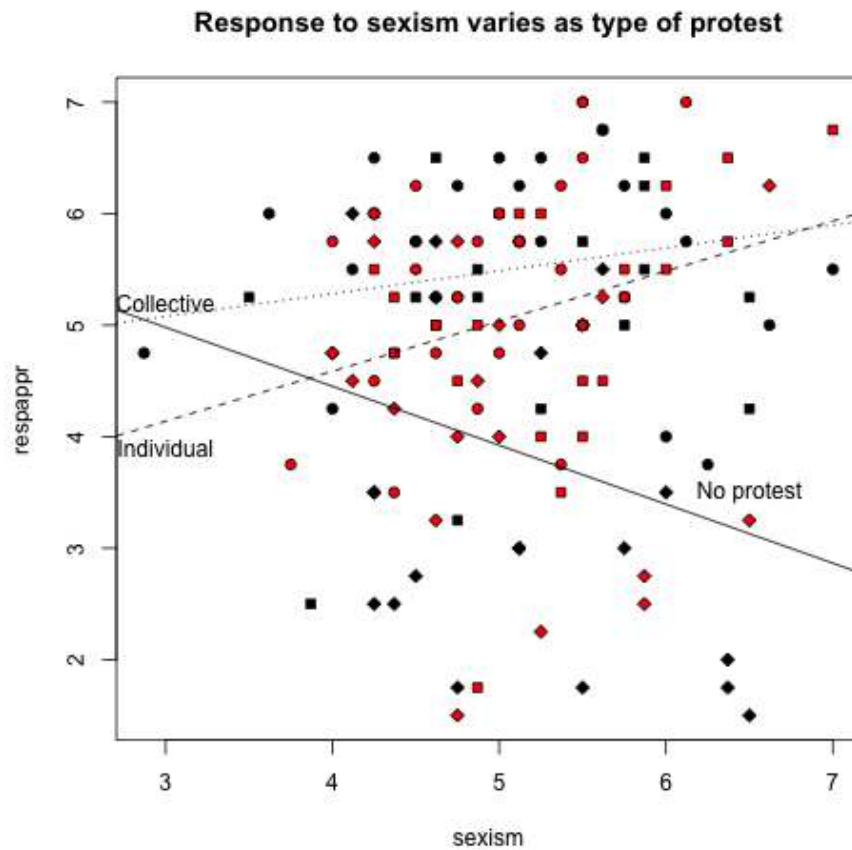
Figure 14: Showing the interaction between type of protest and sexism from the `Garcia` data set. The strength of the regression effects is shown in Fig 13.

6.1 Partial some variables from the rest of the variables

Given a set of X variables and a set of Y variables, we can control for an additional set of Z variables when we find the correlations between X and Y. This is effectively what happens when we want to add covariates into a model. We see this when we compare the regression model for government action as a function of the iteraction of ideology and age with some covariates, or when we partial them out first.

```
> #first,  the more complicated model
> mod.glb <- setCor(govact ~ age * negemot + posemot + ideology + sex,
+                   data=globalWarm,std=FALSE)
> print(mod.glb,digits=3)
```

```
Call: setCor(y = govact ~ age * negemot + posemot + ideology + sex,
    data = globalWarm, std = FALSE)

Multiple Regression from raw data

 DV =  govact
             slope    se       t        p lower.ci upper.ci   VIF    Vy.x
(Intercept)  4.596 0.037 123.917 0.00e+00    4.523    4.669 1.000   0.000
age         -0.001 0.002  -0.577 5.64e-01   -0.006    0.003 1.071   0.002
negemot      0.433 0.026  16.507 5.76e-53    0.382    0.485 1.172   0.281
posemot     -0.021 0.028  -0.768 4.43e-01   -0.076    0.033 1.029  -0.001
ideology    -0.212 0.027  -7.883 1.04e-14   -0.264   -0.159 1.199   0.098
sex         -0.011 0.076  -0.147 8.83e-01   -0.160    0.138 1.053   0.000
age*negemot  0.006 0.002   4.104 4.48e-05    0.003    0.009 1.015   0.020

Residual Standard Error =  1.057  with  808  degrees of freedom

 Multiple Regression
           R    R2   Ruw  R2uw Shrunken R2 SE of R2 overall F df1 df2         p
govact 0.633 0.401 0.356 0.127       0.396    0.026    90.08   6 808 1.825e-86
```

```
> # compare this to the partialled model
>
> mod.glb.partialled <- setCor(govact ~ age * negemot - posemot - ideology - sex,data = globalWarm)
>
```

Note how the beta weights for the age, negemot and interaction terms are identical.

6.2 Partial everything from everything

Sometimes we want to examine just the independent effects of all our variables. That is to say, we want to partial all the variables from all the other variables. I do this with the `partial.r` function. To show the results, I compare the partialed rs to the original rs. I show the lower off diagonal matrix using `lowerMat`. Then to compare the partial matrix to the original matrix, I form the square matrix where the lower off diagonal is the original matrix and the upper off diagonal is the partial matrix.

```
                                 ┌─ R code ─┐
> upper <-partial.r(globalWarm)
> lowerMat(upper)   #show it


         govct posmt negmt idlgy age    sex    prtyd
govact    1.00
posemot  -0.03  1.00
negemot   0.50  0.13  1.00
ideology -0.19  0.00 -0.07  1.00
age      -0.02  0.04  0.03  0.14  1.00
sex       0.00  0.08 -0.07  0.04  0.14  1.00
partyid  -0.08 -0.01 -0.09  0.53  0.03  0.02  1.00
                                 ┌─ R code ─┐
> lower <- lowerCor(globalWarm)


         govct posmt negmt idlgy age    sex    prtyd
govact    1.00
posemot   0.04  1.00
negemot   0.58  0.13  1.00
ideology -0.42 -0.03 -0.35  1.00
age      -0.10  0.04 -0.06  0.21  1.00
sex      -0.10  0.07 -0.12  0.13  0.17  1.00
partyid  -0.36 -0.04 -0.32  0.62  0.15  0.11  1.00
                                 ┌─ R code ─┐
> lowup <- lowerUpper(lower,upper)
>


pdf
  2
```

7   Related packages

`mediate` and `setCor` are just two functions in the *psych* package. There are several additional packages available in R to do mediation. The *mediation* package (Tingley et al., 2014) seems the most powerful, in that is tailor made for mediation. *MBESS* (Kelley, 2017) has a mediation function. Steven Short has a nice tutorial on mediation analysis available for download that discusses how to use R for mediation. And, of course, the *lavaan* package (Rosseel, 2012) is the recommended package to do SEM and path models.
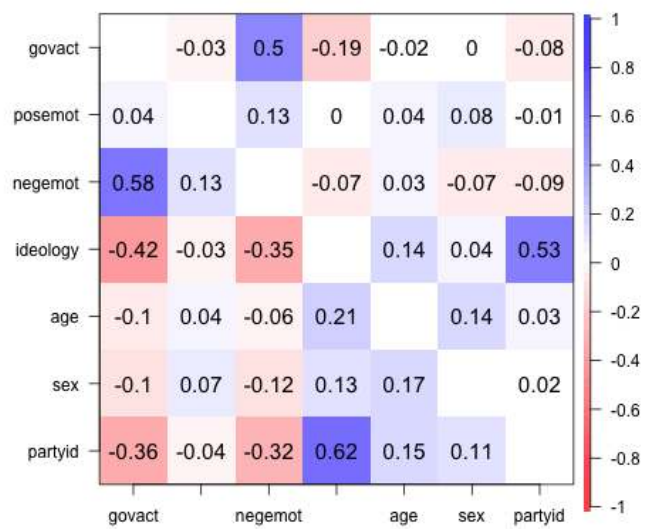
Figure 15: Correlations (below diagonal) and partial correlations (above the diagonal)

8 Development version and a users guide

The *psych* package is available from the CRAN repository. However, the most recent development version of the *psych* package is available as a source file at the repository maintained at http://personality-project.org/r. That version will have removed the most recently discovered bugs (but perhaps introduced other, yet to be discovered ones). To install this development version, either for PCs or Macs,

```R code
install.packages("psych", repos =  "http://personality-project.org/r", type = "source")
```

After doing this, it is important to restart R to get the new package.

Although the individual help pages for the *psych* package are available as part of R and may be accessed directly (e.g. ?psych) , the full manual for the psych package is also available as a pdf at http://personality-project.org/r/psych_manual.pdf

News and a history of changes are available in the NEWS and CHANGES files in the source files. To view the most recent news,

```R code
> news(Version >-= "2.0.12",package="psych")
```

9 Psychometric Theory

The *psych* package has been developed to help psychologists do basic research. Many of the functions were developed to supplement a book (http://personality-project.org/r/book An introduction to Psychometric Theory with Applications in R (Revelle, prep) More information about the use of some of the functions may be found in the book .

For more extensive discussion of the use of *psych* in particular and R in general, consult http://personality-project.org/r/r.guide.html A short guide to R.

10 SessionInfo

This document was prepared using the following settings.

```R code
> sessionInfo()
```

```
R version 4.2.1 (2022-06-23)
Platform: aarch64-apple-darwin20 (64-bit)
Running under: macOS Monterey 12.6

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.2-arm64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.2-arm64/Resources/lib/libRlapack.dylib
```

```
locale:
[1] C

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

other attached packages:
[1] psychTools_2.2.9 psych_2.2.9

loaded via a namespace (and not attached):
 [1] compiler_4.2.1      parallel_4.2.1      tools_4.2.1        foreign_0.8-82      lavaan_0.6-12
 [8] mnormt_2.0.2        grid_4.2.1          pbivnorm_0.6.0     knitr_1.39          xfun_0.30
[15] lattice_0.20-45
```

References

Cohen, J. (1982). Set correlation as a general mulitivariate data-analytic method. *Multivariate Behavioral Research*, 17(3):301–341.

Cohen, J., Cohen, P., West, S. G., and Aiken, L. S. (2003). *Applied multiple regression/correlation analysis for the behavioral sciences*. L. Erlbaum Associates, Mahwah, N.J., 3rd ed edition.

Dawes, R. M. (1979). The robust beauty of improper linear models in decision making. *American Psychologist*, 34(7):571–582.

Garcia, D. M., Schmitt, M. T., Branscombe, N. R., and Ellemers, N. (2010). Women's reactions to ingroup members who protest discriminatory treatment: The importance of beliefs about inequality and response appropriateness. *European Journal of Social Psychology*, 40(5):733–745.

Hayes, A. F. (2013). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach*. Guilford Press, New York.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika*, 28(3/4):321–377.

Kelley, K. (2017). *MBESS: The MBESS R Package*. R package version 4.4.1.

Pollack, J. M., Vanepps, E. M., and Hayes, A. F. (2012). The moderating role of social ties on entrepreneurs' depressed affect and withdrawal intentions in response to economic stress. *Journal of Organizational Behavior*, 33(6):789–810.

Preacher, K. J. (2015). Advances in mediation analysis: A survey and synthesis of new developments. *Annual Review of Psychology*, 66:825–852.

Preacher, K. J. and Hayes, A. F. (2004). SPSS and SAS procedures for estimating indirect effects in simple mediation models. *Behavior Research Methods, Instruments, & Computers*, 36(4):717–731.

Preacher, K. J., Rucker, D. D., and Hayes, A. F. (2007). Addressing moderated mediation hypotheses: Theory, methods, and prescriptions. *Multivariate behavioral research*, 42(1):185–227.

R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Revelle, W. (2020). *psych: Procedures for Personality and Psychological Research*. Northwestern University, Evanston, https://CRAN.r-project.org/package=psych. R package version 2.0.9.

Revelle, W. (in prep). *An introduction to psychometric theory with applications in R*. Springer.

Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, 48(2):1–36.

Tal-Or, N., Cohen, J., Tsfati, Y., and Gunther, A. C. (2010). Testing causal direction in the influence of presumed media influence. *Communication Research*, 37(6):801–824.

Tingley, D., Yamamoto, T., Hirose, K., Keele, L., and Imai, K. (2014). mediation: R package for causal mediation analysis. *Journal of Statistical Software*, 59(5):1–38.

References

Cohen, J. (1982). Set correlation as a general mulitivariate data-analytic method. *Multivariate Behavioral Research*, 17(3):301–341.

Cohen, J., Cohen, P., West, S. G., and Aiken, L. S. (2003). *Applied multiple regression/correlation analysis for the behavioral sciences*. L. Erlbaum Associates, Mahwah, N.J., 3rd ed edition.

Dawes, R. M. (1979). The robust beauty of improper linear models in decision making. *American Psychologist*, 34(7):571–582.

Garcia, D. M., Schmitt, M. T., Branscombe, N. R., and Ellemers, N. (2010). Women's reactions to ingroup members who protest discriminary treatment: The importance of beliefs about inequality and response appropriateness. *European Journal of Social Psychology*, 40(5):733–745.

Hayes, A. F. (2013). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach*. Guilford Press, New York.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika*, 28(3/4):321–377.

Kelley, K. (2017). *MBESS: The MBESS R Package*. R package version 4.4.1.

Pollack, J. M., Vanepps, E. M., and Hayes, A. F. (2012). The moderating role of social ties on entrepreneurs' depressed affect and withdrawal intentions in response to economic stress. *Journal of Organizational Behavior*, 33(6):789–810.

Preacher, K. J. (2015). Advances in mediation analysis: A survey and synthesis of new developments. *Annual Review of Psychology*, 66:825–852.

Preacher, K. J. and Hayes, A. F. (2004). SPSS and SAS procedures for estimating indirect effects in simple mediation models. *Behavior Research Methods, Instruments, & Computers*, 36(4):717–731.

Preacher, K. J., Rucker, D. D., and Hayes, A. F. (2007). Addressing moderated mediation hypotheses: Theory, methods, and prescriptions. *Multivariate behavioral research*, 42(1):185–227.

R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Keon-Woong Moon (2020). *processR: Implementation of the 'PROCESS' Macro*. https://CRAN.R-project.org/package=processR

Revelle, W. (2020). *psych: Procedures for Personality and Psychological Research*. Northwestern University, Evanston, https://CRAN.r-project.org/package=psych. R package version 2.0.12.

Revelle, W. (in prep). *An introduction to psychometric theory with applications in R*. Springer.

Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, 48(2):1–36.

Tal-Or, N., Cohen, J., Tsfati, Y., and Gunther, A. C. (2010). Testing causal direction in the influence of presumed media influence. *Communication Research*, 37(6):801–824.

Tingley, D., Yamamoto, T., Hirose, K., Keele, L., and Imai, K. (2014). mediation: R package for causal mediation analysis. *Journal of Statistical Software*, 59(5):1–38.