PA2-code and figures

```
library(readr)
library(haven)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(psych)
library(lavaan)
```

```
## This is lavaan 0.6-9
## lavaan is FREE software! Please report any bugs.
```

```
##
## Attaching package: 'lavaan'
```

```
## The following object is masked from 'package:psych':
##
##     cor2cov
```

```
library(semPlot)
library(texreg)
```

```
## Version:  1.37.5
## Date:     2020-06-17
## Author:   Philip Leifeld (University of Essex)
##
## Consider submitting praise using the praise or praise_interactive functions.
## Please cite the JSS article in your publications -- see citation("texreg").
```

```
data_60 <- read_csv("~/Downloads/data_60.csv")
```

```
## Rows: 635 Columns: 18
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (18): id, x1, x2, x3, x4, m1, m2, m3, m4, m5, y1, y2, y3, y4, y5, v15, v...

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
##Remove Unit non response
data_60 <- data_60[data_60$id != 145, ]
data_60 <- data_60[data_60$id != 188, ]
data_60 <- data_60[data_60$id != 190, ]
data_60 <- data_60[data_60$id != 258, ]
data_60 <- data_60[data_60$id != 268, ]
data_60 <- data_60[data_60$id != 299, ]
data_60 <- data_60[data_60$id != 314, ]
data_60 <- data_60[data_60$id != 361, ]
data_60 <- data_60[data_60$id != 453, ]
data_60 <- data_60[data_60$id != 549, ]


#Weird Cases

##Duplicates
data60_cl <- add_count(data_60, x1, x2, x3, x4, m1, m2, m3, m4, m5, y1, y2, y3, y4, y5,
                               name = "duplicate")

data60_cl<- data60_cl[order(-data60_cl$duplicate),]

data60_cl <- data60_cl[data60_cl$id != 13, ]
data60_cl <- data60_cl[data60_cl$id != 80, ]

data60_cl$weird <- 0
data60_cl$weird[data60_cl$duplicate == 2] <- 1

##Straight liners
data60_cl$sd <- apply(data60_cl[2:15], 1 , sd)

data60_cl$weird[data60_cl$sd == 0] <- 2
data60_cl <- data60_cl[order(-data60_cl$weird, data60_cl$id),]


##Outliers
summary(data60_cl)
```

```
##        id              x1              x2              x3              x4
##  Min.   :  1.0   Min.   :1.000   Min.   :1.000   Min.   :1.00   Min.   :1.000
##  1st Qu.:159.5   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.00   1st Qu.:2.000
##  Median :321.0   Median :3.000   Median :3.000   Median :3.00   Median :3.000
##  Mean   :319.1   Mean   :3.064   Mean   :3.051   Mean   :3.08   Mean   :3.021
##  3rd Qu.:478.5   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.00   3rd Qu.:4.000
##  Max.   :635.0   Max.   :9.000   Max.   :7.000   Max.   :9.00   Max.   :8.000
##        m1              m2              m3              m4
```

```
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :3.000   Median :3.000   Median :3.000   Median :3.000
##  Mean   :2.978   Mean   :3.079   Mean   :2.902   Mean   :2.987
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :9.000   Max.   :9.000   Max.   :8.000   Max.   :9.000
##        m5              y1              y2              y3
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :3.000   Median :3.000   Median :3.000   Median :3.000
##  Mean   :2.899   Mean   :2.992   Mean   :3.005   Mean   :3.048
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :9.000   Max.   :8.000   Max.   :9.000   Max.   :8.000
##        y4              y5             v15             v16
##  Min.   :1.000   Min.   :1.000   Min.   :0.000   Min.   :-3.174252
##  1st Qu.:2.000   1st Qu.:2.000   1st Qu.:0.000   1st Qu.:-0.690628
##  Median :3.000   Median :3.000   Median :0.000   Median : 0.019442
##  Mean   :3.074   Mean   :3.034   Mean   :0.488   Mean   :-0.008646
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:1.000   3rd Qu.: 0.615984
##  Max.   :9.000   Max.   :8.000   Max.   :1.000   Max.   : 2.941043
##       v17              duplicate        weird               sd
##  Min.   :  0.05405   Min.   :1.00   Min.   :0.00000   Min.   :0.3631
##  1st Qu.:  0.48987   1st Qu.:1.00   1st Qu.:0.00000   1st Qu.:0.8018
##  Median :  0.98386   Median :1.00   Median :0.00000   Median :0.9169
##  Mean   :  2.68932   Mean   :1.01   Mean   :0.00642   Mean   :0.9536
##  3rd Qu.:  1.96512   3rd Qu.:1.00   3rd Qu.:0.00000   3rd Qu.:1.0509
##  Max.   :150.77086   Max.   :3.00   Max.   :1.00000   Max.   :3.0751
```

```
data60_cl$x1[data60_cl$x1 > 5] <- NA
data60_cl$x2[data60_cl$x2 > 5] <- NA
data60_cl$x3[data60_cl$x3 > 5] <- NA
data60_cl$x4[data60_cl$x4 > 5] <- NA
data60_cl$m1[data60_cl$m1 > 5] <- NA
data60_cl$m2[data60_cl$m2 > 5] <- NA
data60_cl$m3[data60_cl$m3 > 5] <- NA
data60_cl$m4[data60_cl$m4 > 5] <- NA
data60_cl$m5[data60_cl$m5 > 5] <- NA
data60_cl$y1[data60_cl$y1 > 5] <- NA
data60_cl$y2[data60_cl$y2 > 5] <- NA
data60_cl$y3[data60_cl$y3 > 5] <- NA
data60_cl$y4[data60_cl$y4 > 5] <- NA
data60_cl$y5[data60_cl$y5 > 5] <- NA

summary(data60_cl)
```

```
##        id              x1              x2              x3
##  Min.   :  1.0   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:159.5   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :321.0   Median :3.000   Median :3.000   Median :3.000
##  Mean   :319.1   Mean   :3.032   Mean   :3.034   Mean   :3.048
##  3rd Qu.:478.5   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :635.0   Max.   :5.000   Max.   :5.000   Max.   :5.000
##                  NA's   :4       NA's   :3       NA's   :4
##        x4              m1              m2              m3              m4
```

```
##  Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:2.00   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :3.000   Median :3.00   Median :3.000   Median :3.000   Median :3.000
##  Mean   :3.006   Mean   :2.96   Mean   :3.069   Mean   :2.881   Mean   :2.977
##  3rd Qu.:4.000   3rd Qu.:4.00   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.000   Max.   :5.000
##  NA's   :2       NA's   :2      NA's   :1       NA's   :3       NA's   :1
##        m5              y1              y2              y3              y4
##  Min.   :1.000   Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.00   1st Qu.:2.000   1st Qu.:2.000
##  Median :3.000   Median :3.000   Median :3.00   Median :3.000   Median :3.000
##  Mean   :2.881   Mean   :2.977   Mean   :2.99   Mean   :3.035   Mean   :3.039
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.00   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.000
##  NA's   :2       NA's   :2       NA's   :2      NA's   :2       NA's   :5
##        y5              v15             v16                 v17
##  Min.   :1.000   Min.   :0.000   Min.   :-3.174252   Min.   :  0.05405
##  1st Qu.:2.000   1st Qu.:0.000   1st Qu.:-0.690628   1st Qu.:  0.48987
##  Median :3.000   Median :0.000   Median : 0.019442   Median :  0.98386
##  Mean   :3.013   Mean   :0.488   Mean   :-0.008646   Mean   :  2.68932
##  3rd Qu.:4.000   3rd Qu.:1.000   3rd Qu.: 0.615984   3rd Qu.:  1.96512
##  Max.   :5.000   Max.   :1.000   Max.   : 2.941043   Max.   :150.77086
##  NA's   :3
##    duplicate        weird               sd
##  Min.   :1.00   Min.   :0.00000   Min.   :0.3631
##  1st Qu.:1.00   1st Qu.:0.00000   1st Qu.:0.8018
##  Median :1.00   Median :0.00000   Median :0.9169
##  Mean   :1.01   Mean   :0.00642   Mean   :0.9536
##  3rd Qu.:1.00   3rd Qu.:0.00000   3rd Qu.:1.0509
##  Max.   :3.00   Max.   :1.00000   Max.   :3.0751
##
```

```
data60_cl$weird[data60_cl$id == 50] <- 3
data60_cl$weird[data60_cl$id == 370] <- 3
data60_cl$weird[data60_cl$id == 405] <- 3
data60_cl$weird[data60_cl$id == 431] <- 3
data60_cl$weird[data60_cl$id == 450] <- 3
data60_cl$weird[data60_cl$id == 497] <- 3

data60_cl <- data60_cl[order(-data60_cl$weird, data60_cl$id),]


skew_x1 <- round(skew(data60_cl$x1),2)
skew_x2 <- round(skew(data60_cl$x2),2)
skew_x3 <- round(skew(data60_cl$x3),2)
skew_x4 <- round(skew(data60_cl$x4),2)

skew_m1 <- round(skew(data60_cl$m1),2)
skew_m2 <- round(skew(data60_cl$m2),2)
skew_m3 <- round(skew(data60_cl$m3),2)
skew_m4 <- round(skew(data60_cl$m4),2)
skew_m5 <- round(skew(data60_cl$m5),2)

skew_y1 <- round(skew(data60_cl$y1),2)
```

```r
skew_y2 <- round(skew(data60_cl$y2),2)
skew_y3 <- round(skew(data60_cl$y3),2)
skew_y4 <- round(skew(data60_cl$y4),2)
skew_y5 <- round(skew(data60_cl$y5),2)

skew_v16 <- round(skew(data60_cl$v16),2)
skew_v17 <- round(skew(data60_cl$v17),2)


data60_sk <- data60_cl

data60_sk$Z_x1 <- scale(data60_sk$x1, center=TRUE, scale=TRUE)
data60_sk$Z_x2 <- scale(data60_sk$x2, center=TRUE, scale=TRUE)
data60_sk$Z_x3 <- scale(data60_sk$x3, center=TRUE, scale=TRUE)
data60_sk$Z_x4 <- scale(data60_sk$x4, center=TRUE, scale=TRUE)

data60_sk$Z_m1 <- scale(data60_sk$m1, center=TRUE, scale=TRUE)
data60_sk$Z_m2 <- scale(data60_sk$m2, center=TRUE, scale=TRUE)
data60_sk$Z_m3 <- scale(data60_sk$m3, center=TRUE, scale=TRUE)
data60_sk$Z_m4 <- scale(data60_sk$m4, center=TRUE, scale=TRUE)
data60_sk$Z_m5 <- scale(data60_sk$m5, center=TRUE, scale=TRUE)

data60_sk$Z_y1 <- scale(data60_sk$y1, center=TRUE, scale=TRUE)
data60_sk$Z_y2 <- scale(data60_sk$y2, center=TRUE, scale=TRUE)
data60_sk$Z_y3 <- scale(data60_sk$y3, center=TRUE, scale=TRUE)
data60_sk$Z_y4 <- scale(data60_sk$y4, center=TRUE, scale=TRUE)
data60_sk$Z_y5 <- scale(data60_sk$y5, center=TRUE, scale=TRUE)

summary(data60_sk)
```

```
##        id              x1              x2              x3
##  Min.   :  1.0   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:159.5   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :321.0   Median :3.000   Median :3.000   Median :3.000
##  Mean   :319.1   Mean   :3.032   Mean   :3.034   Mean   :3.048
##  3rd Qu.:478.5   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :635.0   Max.   :5.000   Max.   :5.000   Max.   :5.000
##                  NA's   :4       NA's   :3       NA's   :4
##        x4              m1              m2              m3              m4
##  Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:2.00   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
##  Median :3.000   Median :3.00   Median :3.000   Median :3.000   Median :3.000
##  Mean   :3.006   Mean   :2.96   Mean   :3.069   Mean   :2.881   Mean   :2.977
##  3rd Qu.:4.000   3rd Qu.:4.00   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.000   Max.   :5.000
##  NA's   :2       NA's   :2      NA's   :1       NA's   :3       NA's   :1
##        m5              y1              y2              y3              y4
##  Min.   :1.000   Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.00   1st Qu.:2.000   1st Qu.:2.000
##  Median :3.000   Median :3.000   Median :3.00   Median :3.000   Median :3.000
##  Mean   :2.881   Mean   :2.977   Mean   :2.99   Mean   :3.035   Mean   :3.039
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.00   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :5.000   Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.000
```

```
##  NA's   :2       NA's   :2       NA's   :2      NA's   :2       NA's   :5
##        y5              v15             v16                 v17
##  Min.   :1.000   Min.   :0.000   Min.   :-3.174252   Min.   :  0.05405
##  1st Qu.:2.000   1st Qu.:0.000   1st Qu.:-0.690628   1st Qu.:  0.48987
##  Median :3.000   Median :0.000   Median : 0.019442   Median :  0.98386
##  Mean   :3.013   Mean   :0.488   Mean   :-0.008646   Mean   :  2.68932
##  3rd Qu.:4.000   3rd Qu.:1.000   3rd Qu.: 0.615984   3rd Qu.:  1.96512
##  Max.   :5.000   Max.   :1.000   Max.   : 2.941043   Max.   :150.77086
##  NA's   :3
##    duplicate        weird               sd               Z_x1.V1
##  Min.   :1.00   Min.   :0.00000   Min.   :0.3631   Min.   :-1.899616
##  1st Qu.:1.00   1st Qu.:0.00000   1st Qu.:0.8018   1st Qu.:-0.964908
##  Median :1.00   Median :0.00000   Median :0.9169   Median :-0.030201
##  Mean   :1.01   Mean   :0.03531   Mean   :0.9536   Mean   : 0.000000
##  3rd Qu.:1.00   3rd Qu.:0.00000   3rd Qu.:1.0509   3rd Qu.: 0.904507
##  Max.   :3.00   Max.   :3.00000   Max.   :3.0751   Max.   : 1.839215
##                                                    NA's   :4
##       Z_x2.V1             Z_x3.V1             Z_x4.V1
##  Min.   :-1.9287831   Min.   :-2.021582   Min.   :-1.9936613
##  1st Qu.:-0.9804520   1st Qu.:-1.034705   1st Qu.:-1.0000307
##  Median :-0.0321209   Median :-0.047829   Median :-0.0064002
##  Mean   : 0.0000000   Mean   : 0.000000   Mean   : 0.0000000
##  3rd Qu.: 0.9162102   3rd Qu.: 0.939047   3rd Qu.: 0.9872303
##  Max.   : 1.8645413   Max.   : 1.925923   Max.   : 1.9808609
##  NA's   :3            NA's   :4           NA's   :2
##       Z_m1.V1             Z_m2.V1             Z_m3.V1
##  Min.   :-1.8523246   Min.   :-1.9544331   Min.   :-1.9858209
##  1st Qu.:-0.9071368   1st Qu.:-1.0098664   1st Qu.:-0.9298955
##  Median : 0.0380510   Median :-0.0652996   Median : 0.1260298
##  Mean   : 0.0000000   Mean   : 0.0000000   Mean   : 0.0000000
##  3rd Qu.: 0.9832389   3rd Qu.: 0.8792671   3rd Qu.: 1.1819551
##  Max.   : 1.9284267   Max.   : 1.8238339   Max.   : 2.2378805
##  NA's   :2            NA's   :1            NA's   :3
##       Z_m4.V1             Z_m5.V1             Z_y1.V1
##  Min.   :-1.9128440   Min.   :-2.0094371   Min.   :-1.9083957
##  1st Qu.:-0.9455359   1st Qu.:-0.9410634   1st Qu.:-0.9433194
##  Median : 0.0217722   Median : 0.1273102   Median : 0.0217570
##  Mean   : 0.0000000   Mean   : 0.0000000   Mean   : 0.0000000
##  3rd Qu.: 0.9890803   3rd Qu.: 1.1956839   3rd Qu.: 0.9868333
##  Max.   : 1.9563885   Max.   : 2.2640576   Max.   : 1.9519096
##  NA's   :1            NA's   :2            NA's   :2
##       Z_y2.V1             Z_y3.V1             Z_y4.V1
##  Min.   :-1.8626183   Min.   :-1.9591454   Min.   :-2.025634
##  1st Qu.:-0.9267882   1st Qu.:-0.9966222   1st Qu.:-1.032109
##  Median : 0.0090418   Median :-0.0340990   Median :-0.038584
##  Mean   : 0.0000000   Mean   : 0.0000000   Mean   : 0.000000
##  3rd Qu.: 0.9448719   3rd Qu.: 0.9284241   3rd Qu.: 0.954942
##  Max.   : 1.8807020   Max.   : 1.8909473   Max.   : 1.948467
##  NA's   :2            NA's   :2            NA's   :5
##       Z_y5.V1
##  Min.   :-1.8511572
##  1st Qu.:-0.9315118
##  Median :-0.0118664
##  Mean   : 0.0000000
```

```
##  3rd Qu.: 0.9077790
##  Max.   : 1.8274244
##  NA's   :3
```

```
#Scale Construction
##Construct new Scales
data60_cl$X <- rowMeans(data60_cl[,c(2,3,4,5)])

data60_cl$M <- rowMeans(data60_cl[,c(6,8,9,10,22)])

data60_cl$Y <- rowMeans(data60_cl[,c(11,12,13,14,15)])


#Weird Variables

dfX <- data.frame (first_column  = (data60_cl$x1),
                   second_column = (data60_cl$x2),
                   third_column = (data60_cl$x3),
                   fourth_column = (data60_cl$x4)
                  )

dfX <- na.exclude(dfX)
names(dfX) = c("x1", "x2", "x3", "x4")
round(cor(dfX), 2)
```

```
##      x1   x2   x3   x4
## x1 1.00 0.35 0.34 0.33
## x2 0.35 1.00 0.41 0.42
## x3 0.34 0.41 1.00 0.34
## x4 0.33 0.42 0.34 1.00
```

```
alpha(dfX)
```

```
##
## Reliability analysis
## Call: alpha(x = dfX)
##
##   raw_alpha std.alpha G6(smc) average_r S/N  ase mean   sd median_r
##        0.7       0.7    0.64      0.37 2.3 0.02    3 0.75     0.35
##
##  lower alpha upper     95% confidence boundaries
## 0.66 0.7 0.74
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r S/N alpha se   var.r med.r
## x1      0.66      0.66    0.57      0.39 1.9    0.024 2.3e-03  0.41
## x2      0.60      0.60    0.50      0.34 1.5    0.028 2.5e-05  0.34
## x3      0.64      0.64    0.54      0.37 1.8    0.025 2.3e-03  0.35
## x4      0.64      0.64    0.54      0.37 1.8    0.025 1.5e-03  0.35
##
##  Item statistics
##      n raw.r std.r r.cor r.drop mean  sd
## x1 617  0.71  0.70  0.53   0.44  3.0 1.1
```

```
## x2 617  0.76  0.75  0.64   0.53  3.0 1.1
## x3 617  0.72  0.72  0.57   0.48  3.1 1.0
## x4 617  0.72  0.72  0.58   0.48  3.0 1.0
##
## Non missing response frequency for each item
##       1    2    3    4    5 miss
## x1 0.08 0.24 0.33 0.27 0.08    0
## x2 0.06 0.25 0.38 0.21 0.10    0
## x3 0.06 0.24 0.39 0.23 0.08    0
## x4 0.05 0.27 0.39 0.22 0.08    0
```

```
dfM <- data.frame (first_column  = (data60_cl$m1),
                   second_column = (data60_cl$m2),
                   third_column = (data60_cl$m3),
                   fourth_column = (data60_cl$m4),
                   fifth_column = (data60_cl$m5)
)
dfM <- na.exclude(dfM)
names(dfM) = c("m1", "m2", "m3", "m4", "m5")
round(cor(dfM), 2)
```

```
##       m1    m2    m3    m4    m5
## m1  1.00 -0.50  0.20  0.50  0.47
## m2 -0.50  1.00 -0.17 -0.48 -0.47
## m3  0.20 -0.17  1.00  0.16  0.13
## m4  0.50 -0.48  0.16  1.00  0.47
## m5  0.47 -0.47  0.13  0.47  1.00
```

```
alpha(dfM)
```

```
## Some items ( m2 ) were negatively correlated with the total scale and
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
##
## Reliability analysis
## Call: alpha(x = dfM)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N   ase mean   sd median_r
##      0.095      0.13    0.37      0.03 0.15 0.052    3 0.47     0.14
##
##  lower alpha upper     95% confidence boundaries
## -0.01 0.1 0.2
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r     S/N alpha se var.r   med.r
## m1    -0.380   -0.2983   0.069   -0.0609 -0.2297    0.088 0.144 -0.0241
## m2     0.659    0.6541   0.617    0.3210  1.8911    0.022 0.031  0.3324
## m3    -0.057   -0.0076   0.352   -0.0019 -0.0076    0.062 0.279 -0.0015
## m4    -0.360   -0.2848   0.091   -0.0587 -0.2217    0.086 0.153 -0.0241
## m5    -0.273   -0.2310   0.132   -0.0492 -0.1877    0.080 0.162 -0.0057
##
```

```
##  Item statistics
##      n raw.r std.r r.cor r.drop mean   sd
## m1 619  0.71  0.70  0.73   0.35  3.0 1.06
## m2 619 -0.24 -0.26 -0.88  -0.58  3.1 1.06
## m3 619  0.54  0.55  0.25   0.15  2.9 0.95
## m4 619  0.70  0.70  0.70   0.35  3.0 1.03
## m5 619  0.66  0.67  0.64   0.32  2.9 0.94
##
## Non missing response frequency for each item
##       1    2    3    4    5 miss
## m1 0.08 0.26 0.34 0.25 0.07    0
## m2 0.06 0.25 0.34 0.26 0.09    0
## m3 0.08 0.25 0.37 0.28 0.01    0
## m4 0.08 0.25 0.35 0.26 0.06    0
## m5 0.08 0.27 0.36 0.29 0.01    0
```

```
dfY <- data.frame (first_column  = (data60_cl$y1),
                   second_column = (data60_cl$y2),
                   third_column = (data60_cl$y3),
                   fourth_column = (data60_cl$y4),
                   fifth_column = (data60_cl$y5)
)
dfY <- na.exclude(dfY)
names(dfY) = c("1y", "y2", "y3", "y4", "y5")
round(cor(dfY), 2)
```

```
##      1y   y2   y3   y4   y5
## 1y 1.00 0.51 0.55 0.58 0.14
## y2 0.51 1.00 0.52 0.54 0.12
## y3 0.55 0.52 1.00 0.60 0.14
## y4 0.58 0.54 0.60 1.00 0.14
## y5 0.14 0.12 0.14 0.14 1.00
```

```
alpha(dfY)
```

```
##
## Reliability analysis
## Call: alpha(x = dfY)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.75      0.76    0.74      0.38 3.1 0.016    3 0.74     0.51
##
##  lower alpha upper     95% confidence boundaries
## 0.72 0.75 0.78
##
##  Reliability if an item is dropped:
##    raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## 1y      0.67      0.67    0.66      0.34 2.1    0.022 0.0535  0.33
## y2      0.68      0.69    0.67      0.36 2.2    0.021 0.0581  0.34
## y3      0.67      0.67    0.65      0.34 2.0    0.022 0.0515  0.32
## y4      0.66      0.66    0.64      0.33 2.0    0.023 0.0466  0.33
## y5      0.83      0.83    0.79      0.55 4.9    0.011 0.0012  0.54
##
```

```
##  Item statistics
##      n raw.r std.r r.cor r.drop mean  sd
## 1y 616  0.78  0.78  0.72   0.62    3 1.0
## y2 616  0.76  0.76  0.67   0.58    3 1.1
## y3 616  0.78  0.79  0.73   0.63    3 1.0
## y4 616  0.80  0.80  0.76   0.65    3 1.0
## y5 616  0.44  0.43  0.18   0.16    3 1.1
##
## Non missing response frequency for each item
##       1    2    3    4    5 miss
## 1y 0.07 0.26 0.38 0.21 0.08    0
## y2 0.08 0.25 0.35 0.24 0.08    0
## y3 0.07 0.21 0.43 0.20 0.09    0
## y4 0.06 0.23 0.38 0.26 0.07    0
## y5 0.06 0.28 0.32 0.23 0.10    0
```

```
##Solve M2
data60_cl$m2r[data60_cl$m2 == 1] <- 5
data60_cl$m2r[data60_cl$m2 == 2] <- 4
data60_cl$m2r[data60_cl$m2 == 3] <- 3
data60_cl$m2r[data60_cl$m2 == 4] <- 2
data60_cl$m2r[data60_cl$m2 == 5] <- 1

dfM2 <- data.frame (first_column  = (data60_cl$m1),
                    second_column = (data60_cl$m2r),
                    third_column = (data60_cl$m3),
                    fourth_column = (data60_cl$m4),
                    fifth_column = (data60_cl$m5)
)
dfM2 <- na.exclude(dfM2)
names(dfM2) = c("m1", "m2r", "m3", "m4", "m5")
round(cor(dfM2), 2)
```

```
##       m1  m2r   m3   m4   m5
## m1  1.00 0.50 0.20 0.50 0.47
## m2r 0.50 1.00 0.17 0.48 0.47
## m3  0.20 0.17 1.00 0.16 0.13
## m4  0.50 0.48 0.16 1.00 0.47
## m5  0.47 0.47 0.13 0.47 1.00
```

```
alpha(dfM2)
```

```
##
## Reliability analysis
## Call: alpha(x = dfM2)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean  sd median_r
##       0.74      0.73    0.71      0.36 2.8 0.016  2.9 0.7     0.47
##
##  lower alpha upper     95% confidence boundaries
## 0.71 0.74 0.77
##
##  Reliability if an item is dropped:
```

```
##     raw_alpha std.alpha G6(smc) average_r S/N alpha se   var.r med.r
## m1       0.65      0.65    0.61      0.31 1.8    0.023 0.03047  0.32
## m2r      0.66      0.65    0.62      0.32 1.9    0.022 0.03092  0.33
## m3       0.79      0.79    0.74      0.48 3.7    0.014 0.00024  0.48
## m4       0.66      0.66    0.62      0.32 1.9    0.022 0.03065  0.33
## m5       0.68      0.67    0.63      0.34 2.0    0.021 0.03000  0.34
##
##  Item statistics
##       n raw.r std.r r.cor r.drop mean   sd
## m1  619  0.78  0.77  0.70   0.60  3.0 1.06
## m2r 619  0.76  0.75  0.68   0.58  2.9 1.06
## m3  619  0.46  0.48  0.24   0.21  2.9 0.95
## m4  619  0.75  0.75  0.67   0.58  3.0 1.03
## m5  619  0.72  0.73  0.64   0.55  2.9 0.94
##
## Non missing response frequency for each item
##        1    2    3    4    5 miss
## m1  0.08 0.26 0.34 0.25 0.07    0
## m2r 0.09 0.26 0.34 0.25 0.06    0
## m3  0.08 0.25 0.37 0.28 0.01    0
## m4  0.08 0.25 0.35 0.26 0.06    0
## m5  0.08 0.27 0.36 0.29 0.01    0
```

```
#Analyses
##Q1 & Q2
model.1 <- "
M ~ a*X
Y  ~ b*M + cp*X
indirect := a*b
direct   := cp
total    := a*b + cp
"
mediation <- sem(model.1, data = data60_cl, se = "bootstrap", bootstrap=1000)

summary(mediation, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 16 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         5
##
##                                                   Used       Total
##   Number of observations                            616         623
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Parameter Estimates:
##
##   Standard errors                             Bootstrap
##   Number of requested bootstrap draws              1000
```

```
##   Number of successful bootstrap draws            1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.327    0.039    8.432    0.000    0.252    0.404
##   Y ~
##     M          (b)    0.394    0.075    5.260    0.000    0.253    0.554
##     X         (cp)    0.105    0.065    1.614    0.106   -0.026    0.225
##    Std.lv  Std.all
##
##     0.327    0.408
##
##     0.394    0.321
##     0.105    0.106
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.301    0.019   15.632    0.000    0.262    0.337
##    .Y                 0.468    0.032   14.795    0.000    0.399    0.527
##    Std.lv  Std.all
##     0.301    0.834
##     0.468    0.858
##
## R-Square:
##                    Estimate
##     M                 0.166
##     Y                 0.142
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect          0.129    0.036    3.604    0.000    0.070    0.208
##     direct            0.105    0.065    1.614    0.107   -0.026    0.225
##     total             0.234    0.043    5.468    0.000    0.149    0.316
##    Std.lv  Std.all
##     0.129    0.131
##     0.105    0.106
##     0.234    0.237
```

```
##Q3 & Q4
model.2 <- "
M ~ a*X + d1*v15 +f1*v16
Y  ~ b*M + cp*X + d2*v15 +f2*v16
indirect := a*b
direct   := cp
total    := a*b + cp
"
covariates <- sem(model.2, data = data60_cl, se = "bootstrap", bootstrap=1000)

summary(covariates, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 20 iterations
##
##   Estimator                                         ML
```

```
##   Optimization method                           NLMINB
##   Number of model parameters                         9
##
##                                                  Used       Total
##   Number of observations                          616         623
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Parameter Estimates:
##
##   Standard errors                             Bootstrap
##   Number of requested bootstrap draws              1000
##   Number of successful bootstrap draws             1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.333    0.039    8.439    0.000    0.254    0.412
##     v15       (d1)    0.126    0.043    2.915    0.004    0.041    0.211
##     v16       (f1)    0.119    0.020    5.824    0.000    0.079    0.158
##   Y ~
##     M          (b)    0.302    0.074    4.080    0.000    0.159    0.456
##     X         (cp)    0.146    0.065    2.250    0.024    0.019    0.265
##     v15       (d2)    0.266    0.054    4.921    0.000    0.164    0.370
##     v16       (f2)    0.173    0.028    6.167    0.000    0.117    0.227
##    Std.lv  Std.all
##
##     0.333    0.415
##     0.126    0.105
##     0.119    0.197
##
##     0.302    0.246
##     0.146    0.148
##     0.266    0.180
##     0.173    0.232
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.284    0.018   15.896    0.000    0.245    0.318
##    .Y                 0.425    0.027   15.568    0.000    0.365    0.471
##    Std.lv  Std.all
##     0.284    0.786
##     0.425    0.780
##
## R-Square:
##                    Estimate
##     M                 0.214
##     Y                 0.220
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
```

```
##     indirect          0.101    0.034    3.003    0.003    0.044    0.175
##     direct            0.146    0.065    2.249    0.025    0.019    0.265
##     total             0.246    0.042    5.806    0.000    0.166    0.328
##    Std.lv  Std.all
##     0.101    0.102
##     0.146    0.148
##     0.246    0.250
```

```
##Q5
model.3 <- "
M ~ a*X + d1*v15 +f1*v16
Y  ~ b*M + cp*X + d2*v15 +f2*v16
v17 ~ g1*M + g2*Y
indirect := a*b
direct   := cp
total    := a*b + cp
"
covariate17 <- sem(model.3, data = data60_cl, se = "bootstrap", bootstrap=1000)
summary(covariate17, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 32 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        12
##
##                                                   Used       Total
##   Number of observations                           616         623
##
## Model Test User Model:
##
##   Test statistic                               104.945
##   Degrees of freedom                                 3
##   P-value (Chi-square)                           0.000
##
## Parameter Estimates:
##
##   Standard errors                            Bootstrap
##   Number of requested bootstrap draws             1000
##   Number of successful bootstrap draws            1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X           (a)    0.333    0.038    8.749    0.000    0.262    0.410
##     v15         (d1)   0.126    0.044    2.904    0.004    0.041    0.213
##     v16         (f1)   0.119    0.020    6.010    0.000    0.079    0.161
##   Y ~
##     M           (b)    0.302    0.073    4.124    0.000    0.173    0.456
##     X           (cp)   0.146    0.065    2.252    0.024    0.011    0.264
##     v15         (d2)   0.266    0.051    5.185    0.000    0.161    0.369
##     v16         (f2)   0.173    0.028    6.177    0.000    0.113    0.226
##   v17 ~
##     M           (g1)  -5.151    1.745   -2.952    0.003   -8.439   -1.890
```

```
##     Y         (g2)    4.650    1.651    2.817    0.005    1.498    7.746
##    Std.lv  Std.all
##
##     0.333    0.415
##     0.126    0.105
##     0.119    0.197
##
##     0.302    0.246
##     0.146    0.148
##     0.266    0.180
##     0.173    0.232
##
##    -5.151   -0.319
##     4.650    0.354
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.284    0.018   15.673    0.000    0.247    0.316
##    .Y                 0.425    0.028   15.038    0.000    0.365    0.474
##    .v17              80.743   33.291    2.425    0.015   21.413  147.493
##    Std.lv  Std.all
##     0.284    0.786
##     0.425    0.780
##    80.743    0.855
##
## R-Square:
##                    Estimate
##     M                 0.214
##     Y                 0.220
##     v17               0.145
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect           0.101    0.033    3.042    0.002    0.050    0.178
##     direct             0.146    0.065    2.251    0.024    0.011    0.264
##     total              0.246    0.042    5.852    0.000    0.167    0.325
##    Std.lv  Std.all
##     0.101    0.102
##     0.146    0.148
##     0.246    0.250
```

```r
#Weird
data60_weird <- data60_cl
data60_weird1 <- data60_cl
data60_weird3 <- data60_cl
data60_weird <- data60_weird[order(-data60_weird$weird, data60_weird$id),]

data60_weird <- data60_weird[data60_weird$id != 22, ]
data60_weird <- data60_weird[data60_weird$id != 217, ]
data60_weird <- data60_weird[data60_weird$id != 345, ]
data60_weird <- data60_weird[data60_weird$id != 425, ]

data60_weird <- data60_weird[data60_weird$id != 50, ]
data60_weird <- data60_weird[data60_weird$id != 370, ]
```

```r
data60_weird <- data60_weird[data60_weird$id != 405, ]
data60_weird <- data60_weird[data60_weird$id != 431, ]
data60_weird <- data60_weird[data60_weird$id != 450, ]
data60_weird <- data60_weird[data60_weird$id != 497, ]

data60_weird1 <- data60_weird1[data60_weird1$id != 50, ]
data60_weird1 <- data60_weird1[data60_weird1$id != 370, ]
data60_weird1 <- data60_weird1[data60_weird1$id != 405, ]
data60_weird1 <- data60_weird1[data60_weird1$id != 431, ]
data60_weird1 <- data60_weird1[data60_weird1$id != 450, ]
data60_weird1 <- data60_weird1[data60_weird1$id != 497, ]

data60_weird3 <- data60_weird3[data60_weird3$id != 22, ]
data60_weird3 <- data60_weird3[data60_weird3$id != 217, ]
data60_weird3 <- data60_weird3[data60_weird3$id != 345, ]
data60_weird3 <- data60_weird3[data60_weird3$id != 425, ]


mediation.W <- sem(model.1, data = data60_weird, se = "bootstrap", bootstrap=1000)
summary(mediation.W, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 17 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         5
##
##                                                   Used       Total
##   Number of observations                           612         613
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Parameter Estimates:
##
##   Standard errors                             Bootstrap
##   Number of requested bootstrap draws             1000
##   Number of successful bootstrap draws            1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.370    0.036   10.312    0.000    0.300    0.441
##   Y ~
##     M          (b)    0.488    0.073    6.664    0.000    0.352    0.634
##     X         (cp)    0.019    0.066    0.288    0.773   -0.116    0.141
##    Std.lv  Std.all
##
##     0.370    0.456
##
##     0.488    0.401
```

```
##     0.019    0.019
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.281    0.017   16.495    0.000    0.249    0.313
##    .Y                 0.438    0.030   14.790    0.000    0.375    0.495
##    Std.lv  Std.all
##     0.281    0.792
##     0.438    0.832
##
## R-Square:
##                    Estimate
##     M                 0.208
##     Y                 0.168
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect          0.181    0.039    4.577    0.000    0.111    0.264
##     direct            0.019    0.066    0.288    0.773   -0.116    0.141
##     total             0.200    0.042    4.714    0.000    0.110    0.277
##    Std.lv  Std.all
##     0.181    0.183
##     0.019    0.019
##     0.200    0.202
```

```
covariates.W <- sem(model.2, data = data60_weird, se = "bootstrap", bootstrap=1000)
summary(covariates.W, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 20 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         9
##
##                                                   Used       Total
##   Number of observations                           612         613
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Parameter Estimates:
##
##   Standard errors                            Bootstrap
##   Number of requested bootstrap draws             1000
##   Number of successful bootstrap draws            1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.372    0.036   10.402    0.000    0.301    0.442
##     v15       (d1)    0.106    0.042    2.525    0.012    0.017    0.186
##     v16       (f1)    0.109    0.021    5.316    0.000    0.067    0.148
```

```
##   Y ~
##     M          (b)    0.397    0.073    5.454    0.000    0.263    0.537
##     X         (cp)    0.058    0.062    0.924    0.355   -0.075    0.171
##     v15       (d2)    0.283    0.051    5.603    0.000    0.181    0.377
##     v16       (f2)    0.176    0.028    6.214    0.000    0.121    0.229
##    Std.lv  Std.all
##
##     0.372    0.459
##     0.106    0.089
##     0.109    0.182
##
##     0.397    0.326
##     0.058    0.058
##     0.283    0.195
##     0.176    0.241
##
## Variances:
##                   Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                0.267    0.017   15.685    0.000    0.231    0.298
##    .Y                0.391    0.025   15.440    0.000    0.339    0.436
##    Std.lv  Std.all
##     0.267    0.752
##     0.391    0.744
##
## R-Square:
##                   Estimate
##     M                0.248
##     Y                0.256
##
## Defined Parameters:
##                   Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect          0.148    0.037    4.034    0.000    0.086    0.224
##     direct            0.058    0.062    0.924    0.356   -0.075    0.171
##     total             0.205    0.039    5.243    0.000    0.122    0.279
##    Std.lv  Std.all
##     0.148    0.150
##     0.058    0.058
##     0.205    0.208
```

```
covariate17.W <- sem(model.3, data = data60_weird, se = "bootstrap", bootstrap=1000)
summary(covariate17.W, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 33 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        12
##
##                                                   Used       Total
##   Number of observations                           612         613
##
## Model Test User Model:
##
##   Test statistic                                67.784
```

```
##   Degrees of freedom                                3
##   P-value (Chi-square)                          0.000
##
## Parameter Estimates:
##
##   Standard errors                             Bootstrap
##   Number of requested bootstrap draws              1000
##   Number of successful bootstrap draws             1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.372    0.036   10.310    0.000    0.300    0.438
##     v15       (d1)    0.106    0.042    2.553    0.011    0.025    0.186
##     v16       (f1)    0.109    0.020    5.573    0.000    0.073    0.149
##   Y ~
##     M          (b)    0.397    0.074    5.371    0.000    0.253    0.539
##     X         (cp)    0.058    0.063    0.909    0.363   -0.071    0.173
##     v15       (d2)    0.283    0.052    5.421    0.000    0.177    0.389
##     v16       (f2)    0.176    0.027    6.426    0.000    0.121    0.234
##   v17 ~
##     M         (g1)   -2.750    1.387   -1.983    0.047   -5.538   -0.204
##     Y         (g2)    2.440    1.365    1.788    0.074   -0.029    5.136
##    Std.lv  Std.all
##
##     0.372    0.459
##     0.106    0.089
##     0.109    0.182
##
##     0.397    0.326
##     0.058    0.058
##     0.283    0.195
##     0.176    0.241
##
##    -2.750   -0.245
##     2.440    0.264
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.267    0.017   15.402    0.000    0.231    0.302
##    .Y                 0.391    0.026   15.232    0.000    0.339    0.439
##    .v17              41.420   19.711    2.101    0.036    6.195   82.136
##    Std.lv  Std.all
##     0.267    0.752
##     0.391    0.744
##    41.420    0.923
##
## R-Square:
##                    Estimate
##     M                 0.248
##     Y                 0.256
##     v17               0.077
##
## Defined Parameters:
```

```
##                   Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect         0.148    0.037    4.001    0.000    0.080    0.227
##     direct           0.058    0.063    0.909    0.363   -0.071    0.173
##     total            0.205    0.040    5.164    0.000    0.126    0.283
##    Std.lv  Std.all
##     0.148    0.150
##     0.058    0.058
##     0.205    0.208
```

```
#Omitted Variable Bias

screenreg(list(mediation, mediation.W),
          custom.model.name =
            c("Model 1: Normal",
              "Model 2: Adjusted for Weird"),
          single.row = TRUE, digits = 3)
```

```
##
## ========================================================================
##                   Model 1: Normal       Model 2: Adjusted for Weird
## ------------------------------------------------------------------------
## M ~ X                0.327 (0.039) ***     0.370 (0.036) ***
## Y ~ M                0.394 (0.075) ***     0.488 (0.073) ***
## Y ~ X                0.105 (0.065)         0.019 (0.066)
## M ~~ M               0.301 (0.019) ***     0.281 (0.017) ***
## Y ~~ Y               0.468 (0.032) ***     0.438 (0.030) ***
## X ~~ X               0.560 (0.000)         0.538 (0.000)
## indirect := a*b      0.129 (0.036) ***     0.181 (0.039) ***
## direct := cp         0.105 (0.065)         0.019 (0.066)
## total := a*b+cp      0.234 (0.043) ***     0.200 (0.042) ***
## ------------------------------------------------------------------------
## agfi                 1.000                 1.000
## AIC               2299.374              2200.727
## BIC               2321.491              2222.811
## cfi                  1.000                 1.000
## chisq                0.000                 0.000
## npar                 5.000                 5.000
## rmsea                0.000                 0.000
## rmsea.conf.high      0.000                 0.000
## srmr                 0.000                 0.000
## tli                  1.000                 1.000
## converged            1                     1
## ngroups              1                     1
## nobs               616                   612
## norig              623                   613
## nexcluded            7                     1
## ========================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

```
screenreg(list(covariates, covariates.W),
          custom.model.name =
            c("Model 1: Normal",
              "Model 2: Adjusted for Weird"),
          single.row = TRUE, digits = 3)
```

```
##
## ========================================================================
##                   Model 1: Normal       Model 2: Adjusted for Weird
## ------------------------------------------------------------------------
## M ~ X               0.333 (0.039) ***     0.372 (0.036) ***
## M ~ v15             0.126 (0.043) **      0.106 (0.042) *
## M ~ v16             0.119 (0.020) ***     0.109 (0.021) ***
## Y ~ M               0.302 (0.074) ***     0.397 (0.073) ***
## Y ~ X               0.146 (0.065) *       0.058 (0.062)
## Y ~ v15             0.266 (0.054) ***     0.283 (0.051) ***
## Y ~ v16             0.173 (0.028) ***     0.176 (0.028) ***
## M ~~ M              0.284 (0.018) ***     0.267 (0.017) ***
## Y ~~ Y              0.425 (0.027) ***     0.391 (0.025) ***
## X ~~ X              0.560 (0.000)         0.538 (0.000)
## X ~~ v15           -0.016 (0.000)        -0.010 (0.000)
## X ~~ v16           -0.010 (0.000)         0.001 (0.000)
## v15 ~~ v15          0.250 (0.000)         0.250 (0.000)
## v15 ~~ v16         -0.018 (0.000)        -0.020 (0.000)
## v16 ~~ v16          0.988 (0.000)         0.989 (0.000)
## indirect := a*b     0.101 (0.034) **      0.148 (0.037) ***
## direct := cp        0.146 (0.065) *       0.058 (0.062)
## total := a*b+cp     0.246 (0.042) ***     0.205 (0.039) ***
## ------------------------------------------------------------------------
## agfi                1.000                 1.000
## AIC              2211.480              2108.802
## BIC              2251.290              2148.553
## cfi                 1.000                 1.000
## chisq               0.000                 0.000
## npar                9.000                 9.000
## rmsea               0.000                 0.000
## rmsea.conf.high     0.000                 0.000
## srmr                0.000                 0.000
## tli                 1.000                 1.000
## converged           1                     1
## ngroups             1                     1
## nobs              616                   612
## norig             623                   613
## nexcluded           7                     1
## ========================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

```
screenreg(list(covariate17, covariate17.W),
          custom.model.name =
            c("Model 1: Normal",
              "Model 2: Adjusted for Weird"),
          single.row = TRUE, digits = 3)
```

```
##
## ========================================================================
##                   Model 1: Normal        Model 2: Adjusted for Weird
## ------------------------------------------------------------------------
```

```
## M ~ X              0.333  (0.038) ***     0.372  (0.036) ***
## M ~ v15            0.126  (0.044) **      0.106  (0.042) *
## M ~ v16            0.119  (0.020) ***     0.109  (0.020) ***
## Y ~ M              0.302  (0.073) ***     0.397  (0.074) ***
## Y ~ X              0.146  (0.065) *       0.058  (0.063)
## Y ~ v15            0.266  (0.051) ***     0.283  (0.052) ***
## Y ~ v16            0.173  (0.028) ***     0.176  (0.027) ***
## v17 ~ M           -5.151  (1.745) **     -2.750  (1.387) *
## v17 ~ Y            4.650  (1.651) **      2.440  (1.365)
## M ~~ M             0.284  (0.018) ***     0.267  (0.017) ***
## Y ~~ Y             0.425  (0.028) ***     0.391  (0.026) ***
## v17 ~~ v17        80.743 (33.291) *      41.420 (19.711) *
## X ~~ X             0.560  (0.000)         0.538  (0.000)
## X ~~ v15          -0.016  (0.000)        -0.010  (0.000)
## X ~~ v16          -0.010  (0.000)         0.001  (0.000)
## v15 ~~ v15         0.250  (0.000)         0.250  (0.000)
## v15 ~~ v16        -0.018  (0.000)        -0.020  (0.000)
## v16 ~~ v16         0.988  (0.000)         0.989  (0.000)
## indirect := a*b    0.101  (0.033) **      0.148  (0.037) ***
## direct := cp       0.146  (0.065) *       0.058  (0.063)
## total := a*b+cp    0.246  (0.042) ***     0.205  (0.040) ***
## ----------------------------------------------------------------------
## agfi               0.364                 0.563
## AIC             6670.633              6130.530
## BIC             6723.712              6183.530
## cfi                0.792                 0.859
## chisq            104.945                67.784
## npar              12.000                12.000
## rmsea              0.235                 0.188
## rmsea.conf.high    0.274                 0.228
## srmr               0.076                 0.061
## tli                0.170                 0.436
## converged          1                     1
## ngroups            1                     1
## nobs             616                   612
## norig            623                   613
## nexcluded          7                     1
## ======================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

```
##Omitted Var Bias 1

mediation.W1 <- sem(model.1, data = data60_weird1, se = "bootstrap", bootstrap=1000)
summary(mediation.W1, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 16 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         5
##
##                                                   Used       Total
##   Number of observations                           616         617
##
```

```
## Model Test User Model:
##
##   Test statistic                                0.000
##   Degrees of freedom                                0
##
## Parameter Estimates:
##
##   Standard errors                            Bootstrap
##   Number of requested bootstrap draws             1000
##   Number of successful bootstrap draws            1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.327    0.041    8.062    0.000    0.247    0.408
##   Y ~
##     M          (b)    0.394    0.078    5.069    0.000    0.251    0.562
##     X         (cp)    0.105    0.067    1.561    0.119   -0.032    0.229
##    Std.lv  Std.all
##
##     0.327    0.408
##
##     0.394    0.321
##     0.105    0.106
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.301    0.019   15.857    0.000    0.261    0.340
##    .Y                 0.468    0.031   15.263    0.000    0.404    0.523
##    Std.lv  Std.all
##     0.301    0.834
##     0.468    0.858
##
## R-Square:
##                    Estimate
##     M                 0.166
##     Y                 0.142
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect          0.129    0.038    3.428    0.001    0.067    0.222
##     direct            0.105    0.067    1.560    0.119   -0.032    0.229
##     total             0.234    0.043    5.435    0.000    0.153    0.317
##    Std.lv  Std.all
##     0.129    0.131
##     0.105    0.106
##     0.234    0.237
```

```
covariates.W1 <- sem(model.2, data = data60_weird1, se = "bootstrap", bootstrap=1000)
summary(covariates.W1, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 20 iterations
##
##   Estimator                                         ML
```

```
##   Optimization method                           NLMINB
##   Number of model parameters                         9
##
##                                                  Used       Total
##   Number of observations                          616         617
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Parameter Estimates:
##
##   Standard errors                             Bootstrap
##   Number of requested bootstrap draws              1000
##   Number of successful bootstrap draws             1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.333    0.038    8.692    0.000    0.256    0.409
##     v15       (d1)    0.126    0.041    3.082    0.002    0.051    0.208
##     v16       (f1)    0.119    0.021    5.727    0.000    0.079    0.159
##   Y ~
##     M          (b)    0.302    0.075    4.054    0.000    0.173    0.460
##     X         (cp)    0.146    0.066    2.215    0.027    0.006    0.263
##     v15       (d2)    0.266    0.054    4.938    0.000    0.155    0.370
##     v16       (f2)    0.173    0.028    6.094    0.000    0.117    0.228
##    Std.lv  Std.all
##
##     0.333    0.415
##     0.126    0.105
##     0.119    0.197
##
##     0.302    0.246
##     0.146    0.148
##     0.266    0.180
##     0.173    0.232
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.284    0.018   15.946    0.000    0.250    0.318
##    .Y                 0.425    0.028   15.043    0.000    0.364    0.475
##    Std.lv  Std.all
##     0.284    0.786
##     0.425    0.780
##
## R-Square:
##                    Estimate
##     M                 0.214
##     Y                 0.220
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
```

```
##     indirect          0.101    0.034    3.000    0.003    0.049    0.180
##     direct            0.146    0.066    2.214    0.027    0.006    0.263
##     total             0.246    0.043    5.760    0.000    0.159    0.325
##    Std.lv  Std.all
##     0.101    0.102
##     0.146    0.148
##     0.246    0.250
```

```
covariate17.W1 <- sem(model.3, data = data60_weird1, se = "bootstrap", bootstrap=1000)
summary(covariate17.W1, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 32 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        12
##
##                                                   Used       Total
##   Number of observations                           616         617
##
## Model Test User Model:
##
##   Test statistic                               104.945
##   Degrees of freedom                                 3
##   P-value (Chi-square)                           0.000
##
## Parameter Estimates:
##
##   Standard errors                            Bootstrap
##   Number of requested bootstrap draws             1000
##   Number of successful bootstrap draws            1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.333    0.039    8.491    0.000    0.258    0.412
##     v15       (d1)    0.126    0.044    2.887    0.004    0.044    0.212
##     v16       (f1)    0.119    0.020    5.854    0.000    0.079    0.158
##   Y ~
##     M          (b)    0.302    0.075    4.010    0.000    0.162    0.454
##     X         (cp)    0.146    0.065    2.261    0.024    0.017    0.268
##     v15       (d2)    0.266    0.052    5.149    0.000    0.163    0.363
##     v16       (f2)    0.173    0.029    5.907    0.000    0.118    0.234
##   v17 ~
##     M         (g1)   -5.151    1.699   -3.032    0.002   -8.679   -1.811
##     Y         (g2)    4.650    1.602    2.902    0.004    1.523    7.915
##    Std.lv  Std.all
##
##     0.333    0.415
##     0.126    0.105
##     0.119    0.197
##
##     0.302    0.246
##     0.146    0.148
```

```
##     0.266    0.180
##     0.173    0.232
##
##    -5.151   -0.319
##     4.650    0.354
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.284    0.018   16.174    0.000    0.248    0.317
##    .Y                 0.425    0.027   15.716    0.000    0.368    0.473
##    .v17              80.743   32.478    2.486    0.013   22.351  145.047
##    Std.lv  Std.all
##     0.284    0.786
##     0.425    0.780
##    80.743    0.855
##
## R-Square:
##                    Estimate
##     M                 0.214
##     Y                 0.220
##     v17               0.145
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect           0.101    0.034    2.991    0.003    0.046    0.177
##     direct             0.146    0.065    2.260    0.024    0.017    0.268
##     total              0.246    0.041    5.969    0.000    0.167    0.326
##    Std.lv  Std.all
##     0.101    0.102
##     0.146    0.148
##     0.246    0.250
```

```
screenreg(list(mediation.W, mediation.W1),
          custom.model.name =
            c("Model 1: All Weird removed",
              "Model 2: Flatliners removed"),
          single.row = TRUE, digits = 3)
```

```
##
## ============================================================================
##                   Model 1: All Weird removed  Model 2: Flatliners removed
## ----------------------------------------------------------------------------
## M ~ X               0.370 (0.036) ***           0.327 (0.041) ***
## Y ~ M               0.488 (0.073) ***           0.394 (0.078) ***
## Y ~ X               0.019 (0.066)               0.105 (0.067)
## M ~~ M              0.281 (0.017) ***           0.301 (0.019) ***
## Y ~~ Y              0.438 (0.030) ***           0.468 (0.031) ***
## X ~~ X              0.538 (0.000)               0.560 (0.000)
## indirect := a*b     0.181 (0.039) ***           0.129 (0.038) ***
## direct := cp        0.019 (0.066)               0.105 (0.067)
## total := a*b+cp     0.200 (0.042) ***           0.234 (0.043) ***
## ----------------------------------------------------------------------------
## agfi                1.000                       1.000
## AIC              2200.727                    2299.374
```

```
## BIC              2222.811                    2321.491
## cfi                 1.000                       1.000
## chisq               0.000                       0.000
## npar                5.000                       5.000
## rmsea               0.000                       0.000
## rmsea.conf.high     0.000                       0.000
## srmr                0.000                       0.000
## tli                 1.000                       1.000
## converged           1                           1
## ngroups             1                           1
## nobs              612                         616
## norig             613                         617
## nexcluded           1                           1
## ================================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

```
screenreg(list(covariates.W, covariates.W1),
          custom.model.name =
            c("Model 1: All Weird removed",
              "Model 2: Flatliners removed"),
          single.row = TRUE, digits = 3)
```

```
##
## ================================================================================
##                   Model 1: All Weird removed  Model 2: Flatliners removed
## --------------------------------------------------------------------------------
## M ~ X                0.372 (0.036) ***           0.333 (0.038) ***
## M ~ v15              0.106 (0.042) *             0.126 (0.041) **
## M ~ v16              0.109 (0.021) ***           0.119 (0.021) ***
## Y ~ M                0.397 (0.073) ***           0.302 (0.075) ***
## Y ~ X                0.058 (0.062)               0.146 (0.066) *
## Y ~ v15              0.283 (0.051) ***           0.266 (0.054) ***
## Y ~ v16              0.176 (0.028) ***           0.173 (0.028) ***
## M ~~ M               0.267 (0.017) ***           0.284 (0.018) ***
## Y ~~ Y               0.391 (0.025) ***           0.425 (0.028) ***
## X ~~ X               0.538 (0.000)               0.560 (0.000)
## X ~~ v15            -0.010 (0.000)              -0.016 (0.000)
## X ~~ v16             0.001 (0.000)              -0.010 (0.000)
## v15 ~~ v15           0.250 (0.000)               0.250 (0.000)
## v15 ~~ v16          -0.020 (0.000)              -0.018 (0.000)
## v16 ~~ v16           0.989 (0.000)               0.988 (0.000)
## indirect := a*b      0.148 (0.037) ***           0.101 (0.034) **
## direct := cp         0.058 (0.062)               0.146 (0.066) *
## total := a*b+cp      0.205 (0.039) ***           0.246 (0.043) ***
## --------------------------------------------------------------------------------
## agfi                 1.000                       1.000
## AIC               2108.802                    2211.480
## BIC               2148.553                    2251.290
## cfi                  1.000                       1.000
## chisq                0.000                       0.000
## npar                 9.000                       9.000
## rmsea                0.000                       0.000
## rmsea.conf.high      0.000                       0.000
## srmr                 0.000                       0.000
```

```
## tli                 1.000                       1.000
## converged           1                           1
## ngroups             1                           1
## nobs              612                         616
## norig             613                         617
## nexcluded           1                           1
## ============================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

```
screenreg(list(covariate17.W, covariate17.W1),
          custom.model.name =
            c("Model 1: All Weird removed",
              "Model 2: Flatliners removed"),
          single.row = TRUE, digits = 3)
```

```
##
## ============================================================================
##                   Model 1: All Weird removed  Model 2: Flatliners removed
## ----------------------------------------------------------------------------
## M ~ X                0.372  (0.036) ***          0.333  (0.039) ***
## M ~ v15              0.106  (0.042) *            0.126  (0.044) **
## M ~ v16              0.109  (0.020) ***          0.119  (0.020) ***
## Y ~ M                0.397  (0.074) ***          0.302  (0.075) ***
## Y ~ X                0.058  (0.063)              0.146  (0.065) *
## Y ~ v15              0.283  (0.052) ***          0.266  (0.052) ***
## Y ~ v16              0.176  (0.027) ***          0.173  (0.029) ***
## v17 ~ M             -2.750  (1.387) *           -5.151  (1.699) **
## v17 ~ Y              2.440  (1.365)              4.650  (1.602) **
## M ~~ M               0.267  (0.017) ***          0.284  (0.018) ***
## Y ~~ Y               0.391  (0.026) ***          0.425  (0.027) ***
## v17 ~~ v17          41.420 (19.711) *           80.743 (32.478) *
## X ~~ X               0.538  (0.000)              0.560  (0.000)
## X ~~ v15            -0.010  (0.000)             -0.016  (0.000)
## X ~~ v16             0.001  (0.000)             -0.010  (0.000)
## v15 ~~ v15           0.250  (0.000)              0.250  (0.000)
## v15 ~~ v16          -0.020  (0.000)             -0.018  (0.000)
## v16 ~~ v16           0.989  (0.000)              0.988  (0.000)
## indirect := a*b      0.148  (0.037) ***          0.101  (0.034) **
## direct := cp         0.058  (0.063)              0.146  (0.065) *
## total := a*b+cp      0.205  (0.040) ***          0.246  (0.041) ***
## ----------------------------------------------------------------------------
## agfi                 0.563                       0.364
## AIC               6130.530                    6670.633
## BIC               6183.530                    6723.712
## cfi                  0.859                       0.792
## chisq               67.784                     104.945
## npar                12.000                      12.000
## rmsea                0.188                       0.235
## rmsea.conf.high      0.228                       0.274
## srmr                 0.061                       0.076
## tli                  0.436                       0.170
## converged            1                           1
## ngroups              1                           1
## nobs               612                         616
```

```
## norig             613                           617
## nexcluded           1                             1
## ============================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

```
##Omitted Variable Bias Weird3


mediation.W3 <- sem(model.1, data = data60_weird3, se = "bootstrap", bootstrap=1000)
summary(mediation.W3, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 17 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         5
##
##                                                   Used       Total
##   Number of observations                           612         619
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Parameter Estimates:
##
##   Standard errors                            Bootstrap
##   Number of requested bootstrap draws             1000
##   Number of successful bootstrap draws            1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X           (a)    0.370    0.037   10.121    0.000    0.303    0.446
##   Y ~
##     M           (b)    0.488    0.074    6.581    0.000    0.358    0.648
##     X          (cp)    0.019    0.064    0.296    0.767   -0.118    0.133
##    Std.lv  Std.all
##
##     0.370    0.456
##
##     0.488    0.401
##     0.019    0.019
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.281    0.017   16.787    0.000    0.244    0.311
##    .Y                 0.438    0.029   14.887    0.000    0.376    0.490
##    Std.lv  Std.all
##     0.281    0.792
##     0.438    0.832
##
## R-Square:
```

```
##                    Estimate
##     M                 0.208
##     Y                 0.168
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect          0.181    0.040    4.564    0.000    0.116    0.267
##     direct            0.019    0.064    0.296    0.767   -0.118    0.133
##     total             0.200    0.040    4.972    0.000    0.118    0.279
##    Std.lv  Std.all
##     0.181    0.183
##     0.019    0.019
##     0.200    0.202
```

```
covariates.W3 <- sem(model.2, data = data60_weird3, se = "bootstrap", bootstrap=1000)
summary(covariates.W3, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 20 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         9
##
##                                                   Used       Total
##   Number of observations                           612         619
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Parameter Estimates:
##
##   Standard errors                            Bootstrap
##   Number of requested bootstrap draws             1000
##   Number of successful bootstrap draws            1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##   M ~
##     X          (a)    0.372    0.034   11.009    0.000    0.304    0.439
##     v15       (d1)    0.106    0.041    2.618    0.009    0.027    0.192
##     v16       (f1)    0.109    0.020    5.536    0.000    0.071    0.148
##   Y ~
##     M          (b)    0.397    0.066    5.982    0.000    0.270    0.532
##     X         (cp)    0.058    0.058    0.988    0.323   -0.055    0.168
##     v15       (d2)    0.283    0.053    5.363    0.000    0.179    0.390
##     v16       (f2)    0.176    0.027    6.409    0.000    0.121    0.226
##    Std.lv  Std.all
##
##     0.372    0.459
##     0.106    0.089
##     0.109    0.182
##
```

```
##     0.397    0.326
##     0.058    0.058
##     0.283    0.195
##     0.176    0.241
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.267    0.017   16.025    0.000    0.232    0.299
##    .Y                 0.391    0.024   16.255    0.000    0.341    0.433
##    Std.lv  Std.all
##     0.267    0.752
##     0.391    0.744
##
## R-Square:
##                    Estimate
##     M                 0.248
##     Y                 0.256
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect           0.148    0.033    4.468    0.000    0.089    0.217
##     direct             0.058    0.058    0.987    0.323   -0.055    0.168
##     total              0.205    0.039    5.261    0.000    0.127    0.280
##    Std.lv  Std.all
##     0.148    0.150
##     0.058    0.058
##     0.205    0.208
```

```
covariate17.W3 <- sem(model.3, data = data60_weird3, se = "bootstrap", bootstrap=1000)
summary(covariate17.W3, ci=T, standardized=T, rsquare=T, fit.measures=F)
```

```
## lavaan 0.6-9 ended normally after 33 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        12
##
##                                                   Used       Total
##   Number of observations                           612         619
##
## Model Test User Model:
##
##   Test statistic                                67.784
##   Degrees of freedom                                 3
##   P-value (Chi-square)                           0.000
##
## Parameter Estimates:
##
##   Standard errors                             Bootstrap
##   Number of requested bootstrap draws              1000
##   Number of successful bootstrap draws             1000
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
```

```
##   M ~
##     X          (a)    0.372    0.035   10.781    0.000    0.304    0.439
##     v15       (d1)    0.106    0.041    2.566    0.010    0.025    0.189
##     v16       (f1)    0.109    0.020    5.536    0.000    0.071    0.148
##   Y ~
##     M          (b)    0.397    0.070    5.678    0.000    0.268    0.543
##     X         (cp)    0.058    0.060    0.966    0.334   -0.064    0.169
##     v15       (d2)    0.283    0.052    5.461    0.000    0.181    0.381
##     v16       (f2)    0.176    0.027    6.579    0.000    0.125    0.229
##   v17 ~
##     M         (g1)   -2.750    1.326   -2.073    0.038   -5.387   -0.199
##     Y         (g2)    2.440    1.313    1.859    0.063   -0.040    5.064
##    Std.lv  Std.all
##
##     0.372    0.459
##     0.106    0.089
##     0.109    0.182
##
##     0.397    0.326
##     0.058    0.058
##     0.283    0.195
##     0.176    0.241
##
##    -2.750   -0.245
##     2.440    0.264
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##    .M                 0.267    0.017   15.760    0.000    0.235    0.299
##    .Y                 0.391    0.025   15.744    0.000    0.341    0.439
##    .v17              41.420   18.724    2.212    0.027    7.123   80.542
##    Std.lv  Std.all
##     0.267    0.752
##     0.391    0.744
##    41.420    0.923
##
## R-Square:
##                    Estimate
##     M                 0.248
##     Y                 0.256
##     v17               0.077
##
## Defined Parameters:
##                    Estimate  Std.Err  z-value  P(>|z|) ci.lower ci.upper
##     indirect           0.148    0.035    4.222    0.000    0.089    0.223
##     direct             0.058    0.060    0.965    0.334   -0.064    0.169
##     total              0.205    0.040    5.127    0.000    0.128    0.284
##    Std.lv  Std.all
##     0.148    0.150
##     0.058    0.058
##     0.205    0.208
```

```
screenreg(list(mediation.W, mediation.W3),
          custom.model.name =
```

```
            c("Model 1: All Weird removed",
              "Model 2: Partly Item Non Response removed"),
          single.row = TRUE, digits = 3)
```

```
##
## ============================================================================================
##                   Model 1: All Weird removed  Model 2: Partly Item Non Response removed
## --------------------------------------------------------------------------------------------
## M ~ X                0.370 (0.036) ***           0.370 (0.037) ***
## Y ~ M                0.488 (0.073) ***           0.488 (0.074) ***
## Y ~ X                0.019 (0.066)               0.019 (0.064)
## M ~~ M               0.281 (0.017) ***           0.281 (0.017) ***
## Y ~~ Y               0.438 (0.030) ***           0.438 (0.029) ***
## X ~~ X               0.538 (0.000)               0.538 (0.000)
## indirect := a*b      0.181 (0.039) ***           0.181 (0.040) ***
## direct := cp         0.019 (0.066)               0.019 (0.064)
## total := a*b+cp      0.200 (0.042) ***           0.200 (0.040) ***
## --------------------------------------------------------------------------------------------
## agfi                    1.000                       1.000
## AIC                  2200.727                    2200.727
## BIC                  2222.811                    2222.811
## cfi                     1.000                       1.000
## chisq                   0.000                       0.000
## npar                    5.000                       5.000
## rmsea                   0.000                       0.000
## rmsea.conf.high         0.000                       0.000
## srmr                    0.000                       0.000
## tli                     1.000                       1.000
## converged               1                           1
## ngroups                 1                           1
## nobs                  612                         612
## norig                 613                         619
## nexcluded               1                           7
## ============================================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

```
screenreg(list(covariates.W, covariates.W3),
          custom.model.name =
            c("Model 1: All Weird removed",
              "Model 2: Partly Item Non Response removed"),
          single.row = TRUE, digits = 3)
```

```
##
## ============================================================================================
##                   Model 1: All Weird removed  Model 2: Partly Item Non Response removed
## --------------------------------------------------------------------------------------------
## M ~ X                0.372 (0.036) ***           0.372 (0.034) ***
## M ~ v15              0.106 (0.042) *             0.106 (0.041) **
## M ~ v16              0.109 (0.021) ***           0.109 (0.020) ***
## Y ~ M                0.397 (0.073) ***           0.397 (0.066) ***
## Y ~ X                0.058 (0.062)               0.058 (0.058)
## Y ~ v15              0.283 (0.051) ***           0.283 (0.053) ***
```

```
## Y ~ v16             0.176 (0.028) ***           0.176 (0.027) ***
## M ~~ M              0.267 (0.017) ***           0.267 (0.017) ***
## Y ~~ Y              0.391 (0.025) ***           0.391 (0.024) ***
## X ~~ X              0.538 (0.000)               0.538 (0.000)
## X ~~ v15           -0.010 (0.000)              -0.010 (0.000)
## X ~~ v16            0.001 (0.000)               0.001 (0.000)
## v15 ~~ v15          0.250 (0.000)               0.250 (0.000)
## v15 ~~ v16         -0.020 (0.000)              -0.020 (0.000)
## v16 ~~ v16          0.989 (0.000)               0.989 (0.000)
## indirect := a*b     0.148 (0.037) ***           0.148 (0.033) ***
## direct := cp        0.058 (0.062)               0.058 (0.058)
## total := a*b+cp     0.205 (0.039) ***           0.205 (0.039) ***
## ------------------------------------------------------------------------------------
## agfi                1.000                       1.000
## AIC              2108.802                    2108.802
## BIC              2148.553                    2148.553
## cfi                 1.000                       1.000
## chisq               0.000                       0.000
## npar                9.000                       9.000
## rmsea               0.000                       0.000
## rmsea.conf.high     0.000                       0.000
## srmr                0.000                       0.000
## tli                 1.000                       1.000
## converged           1                           1
## ngroups             1                           1
## nobs              612                         612
## norig             613                         619
## nexcluded           1                           7
## ====================================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

```
screenreg(list(covariate17.W, covariate17.W3),
          custom.model.name =
            c("Model 1: All Weird removed",
              "Model 2: Partly Item Non Response removed"),
          single.row = TRUE, digits = 3)
```

```
##
## ====================================================================================
##                   Model 1: All Weird removed  Model 2: Partly Item Non Response removed
## ------------------------------------------------------------------------------------
## M ~ X               0.372  (0.036) ***          0.372  (0.035) ***
## M ~ v15             0.106  (0.042) *            0.106  (0.041) *
## M ~ v16             0.109  (0.020) ***          0.109  (0.020) ***
## Y ~ M               0.397  (0.074) ***          0.397  (0.070) ***
## Y ~ X               0.058  (0.063)              0.058  (0.060)
## Y ~ v15             0.283  (0.052) ***          0.283  (0.052) ***
## Y ~ v16             0.176  (0.027) ***          0.176  (0.027) ***
## v17 ~ M            -2.750  (1.387) *           -2.750  (1.326) *
## v17 ~ Y             2.440  (1.365)              2.440  (1.313)
## M ~~ M              0.267  (0.017) ***          0.267  (0.017) ***
## Y ~~ Y              0.391  (0.026) ***          0.391  (0.025) ***
## v17 ~~ v17         41.420 (19.711) *           41.420 (18.724) *
## X ~~ X              0.538  (0.000)              0.538  (0.000)
```

```
## X ~~ v15           -0.010  (0.000)             -0.010  (0.000)
## X ~~ v16            0.001  (0.000)              0.001  (0.000)
## v15 ~~ v15          0.250  (0.000)              0.250  (0.000)
## v15 ~~ v16         -0.020  (0.000)             -0.020  (0.000)
## v16 ~~ v16          0.989  (0.000)              0.989  (0.000)
## indirect := a*b     0.148  (0.037) ***          0.148  (0.035) ***
## direct := cp        0.058  (0.063)              0.058  (0.060)
## total := a*b+cp     0.205  (0.040) ***          0.205  (0.040) ***
## ------------------------------------------------------------------------------------
## agfi                0.563                      0.563
## AIC              6130.530                   6130.530
## BIC              6183.530                   6183.530
## cfi                 0.859                      0.859
## chisq              67.784                     67.784
## npar               12.000                     12.000
## rmsea               0.188                      0.188
## rmsea.conf.high     0.228                      0.228
## srmr                0.061                      0.061
## tli                 0.436                      0.436
## converged           1                          1
## ngroups             1                          1
## nobs              612                        612
## norig             613                        619
## nexcluded           1                          7
## ====================================================================================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```