Software for mediation analysis

Yves Rosseel
Department of Data Analysis
Ghent University – Belgium

Symposium on Causal Mediation Analysis
January 28–29, 2013 – Ghent University
Software for mediation analysis – two traditions

- traditional software for mediation analysis
  - Baron and Kenny (1986) tradition
  - many applied researchers still follow these steps
  - using SPSS/SAS, often in combination with macros/scripts
  - modern approach: using SEM software
  - psychologists are very familiar with this approach

- modern software for mediation analysis:
  - based on the causal inference literature
  - custom macros/code available for SPSS, SAS, Stata, R, …
  - psychologists are very unfamiliar with this approach
    (and the accompanying software)
Some links and resources

- traditional approaches (mostly non-SEM)
  - Andrew F. Hayes website (‘My Macros and Code for SPSS and SAS’):
    http://www.afhayes.com/
  - Kristopher J. Preacher
    http://quantpsy.org/medn.htm
  - David A. Kenny
    http://davidakenny.net

- approaches based on the counterfactual framework
  - Stata: command paramed (Richard Emsley and Hanhua Liu), …
  - R: package ‘mediation’
Software for SEM

Commercial – closed-source

- LISREL, EQS, AMOS, MPLUS
- SAS/Stat: proc CALIS, proc TCALIS
- SEPATH (Statistica), RAMONA (Systat), Stata 12
- Mx (free, closed-source)

Non-commercial – open-source

- outside the R ecosystem: gllamm (Stata), …
- R packages:
  - sem
  - OpenMx
  - lavaan
What is lavaan?

- **lavaan** is an R package for latent variable analysis:
  - confirmatory factor analysis: function `cfa()`
  - structural equation modeling: function `sem()`
  - latent curve analysis / growth modeling: function `growth()`
  - general mean/covariance structure modeling: function `lavaan()`
  - support for continuous, binary and ordinal data

- under development, future plans:
  - multilevel SEM, mixture/latent-class SEM, Bayesian SEM

- the long-term goal of **lavaan** is
  1. to implement all the state-of-the-art capabilities that are currently available in commercial packages
  2. to provide a modular and extensible platform that allows for easy implementation and testing of new statistical and modeling ideas
Installing lavaan, finding documentation

- **lavaan** depends on the R project for statistical computing:

  http://www.r-project.org

- to install **lavaan**, simply start up an R session and type:

  > install.packages("lavaan")

- more information about **lavaan**:

  http://lavaan.org

- the lavaan paper:


- **lavaan** development:

  https://github.com/yrosseel/lavaan
the lavaan model syntax – a simple regression

```r
library(lavaan)
myData <- read.csv("c:/temp/myData.csv")

myModel <- ' y ~ x1 + x2 + x3 + x4 '

# fit model
fit <- sem(model = myModel, 
            data = myData)

# show results
summary(fit)
```

The standard linear model:

- \( y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \beta_4 x_{i4} + \epsilon_i \quad (i = 1, 2, \ldots, n) \)
- to ‘see’ the intercept, use

```
fit <- sem(model=myModel, data=myData, meanstructure=TRUE)
```
output (artificial data, N=100)

lavaan (0.5-12) converged normally after 1 iterations

Number of observations 100

Estimator ML
Minimum Function Test Statistic 0.000
Degrees of freedom 0
P-value (Chi-square) 1.000

Parameter estimates:

<table>
<thead>
<tr>
<th>Information</th>
<th>Expected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Errors</td>
<td>Standard</td>
</tr>
</tbody>
</table>

| Estimate | Std.err | Z-value | P(|z|) |
|----------|---------|---------|-------|
| Regressions: y ~                     |
| x1       | 5.047   | 0.048   | 105.425 | 0.000 |
| x2       | -2.070  | 0.048   | -43.198 | 0.000 |
| x3       | 10.029  | 0.049   | 204.811 | 0.000 |
| x4       | 0.119   | 0.052   | 2.308   | 0.021 |

Variances:

| y        | 0.240 | 0.034 |

Yves Rosseel  Software for mediation analysis
the lavaan model syntax – multivariate regression

\[
\text{myModel <- ' y1 ~ x1 + x2 + x3 + x4 y2 ~ x1 + x2 + x3 + x4 '}
\]
the lavaan model syntax – path analysis

\[
\text{myModel} \leftarrow \ ' x5 \sim x1 + x2 + x3 \\
x6 \sim x4 + x5 \\
x7 \sim x6 ',
\]
Simple mediation example

```r
model <- ' 
  Y ~ M + X 
  M ~ X 
', 

fit <- sem(model, data=myData) 
summary(fit)
```
## output

...  

### Parameter estimates:

| Information | Expected Standard Errors | Standard
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimate</td>
<td>Std.err</td>
<td>Z-value</td>
</tr>
</tbody>
</table>

#### Regressions:

Y ~

<table>
<thead>
<tr>
<th>M</th>
<th>0.597</th>
<th>0.102</th>
<th>5.880</th>
<th>0.000</th>
</tr>
</thead>
</table>

X

<table>
<thead>
<tr>
<th>M</th>
<th>2.594</th>
<th>1.080</th>
<th>2.403</th>
<th>0.016</th>
</tr>
</thead>
</table>

M ~

<table>
<thead>
<tr>
<th>X</th>
<th>2.739</th>
<th>1.027</th>
<th>2.666</th>
<th>0.008</th>
</tr>
</thead>
</table>

#### Variances:

<table>
<thead>
<tr>
<th>Y</th>
<th>108.700</th>
<th>15.372</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>105.408</td>
<td>14.907</td>
</tr>
</tbody>
</table>
Simple mediation example – add labels

```
model <- '  
  Y ~ b*M + c*X
  M ~ a*X
' 

fit <- sem(model, data=myData)
summary(fit)
```
Parameter estimates:

<table>
<thead>
<tr>
<th>Information</th>
<th>Expected Standard Errors</th>
<th>Standard Errors</th>
</tr>
</thead>
</table>

| Regressions: | Estimate | Std.err | Z-value | P(>|z|) |
|--------------|----------|---------|---------|--------|
| Y ~ M        | 0.597    | 0.102   | 5.880   | 0.000  |
| Y ~ X        | 2.594    | 1.080   | 2.403   | 0.016  |
| M ~ X        | 2.739    | 1.027   | 2.666   | 0.008  |

<table>
<thead>
<tr>
<th>Variances:</th>
<th>Y</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>108.700</td>
<td>15.372</td>
</tr>
<tr>
<td>M</td>
<td>105.408</td>
<td>14.907</td>
</tr>
</tbody>
</table>
Simple mediation example – define indirect/total effect

\[ M \sim a \times X \]

\[ Y \sim b \times M + c \times X \]

\[ \text{indirect} := a \times b \]

\[ \text{total} := c + (a \times b) \]

\[
\text{model} \leftarrow 'Y \sim b \times M + c \times X\\
M \sim a \times X\\
\text{indirect} := a \times b\\
\text{total} := c + (a \times b)'\\
\]

fit <- sem(model, data=myData)
summary(fit)
## Parameter estimates:

<table>
<thead>
<tr>
<th>Information</th>
<th>Expected Standard Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Estimate  Std.err  Z-value  P(&gt;</td>
</tr>
<tr>
<td>Regressions:</td>
<td></td>
</tr>
<tr>
<td>$Y \sim$</td>
<td></td>
</tr>
<tr>
<td>$M$ (b)</td>
<td>0.597  0.102  5.880  0.000</td>
</tr>
<tr>
<td>$X$ (c)</td>
<td>2.594  1.080  2.403  0.016</td>
</tr>
<tr>
<td>$M \sim$</td>
<td></td>
</tr>
<tr>
<td>$X$ (a)</td>
<td>2.739  1.027  2.666  0.008</td>
</tr>
<tr>
<td>Variances:</td>
<td></td>
</tr>
<tr>
<td>$Y$</td>
<td>108.700  15.372</td>
</tr>
<tr>
<td>$M$</td>
<td>105.408  14.907</td>
</tr>
<tr>
<td>Defined parameters:</td>
<td></td>
</tr>
<tr>
<td>indirect</td>
<td>1.636  0.674  2.428  0.015</td>
</tr>
<tr>
<td>total</td>
<td>4.230  1.210  3.495  0.000</td>
</tr>
</tbody>
</table>
Simple mediation example – bootstrapping

\[
\begin{align*}
  \text{model} & <- ' \\
  & Y \sim b \ast M + c \ast X \\
  & M \sim a \ast X \\
  & \text{indirect} := a \ast b \\
  & \text{total} := c + (a \ast b) \\
  \end{align*}
\]

fit <- sem(model, data=myData, se="bootstrap")
summary(fit)
output

...  

Parameter estimates:

<table>
<thead>
<tr>
<th>Information</th>
<th>Observed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Errors</td>
<td>Bootstrap</td>
</tr>
<tr>
<td>Number of requested bootstrap draws</td>
<td>1000</td>
</tr>
<tr>
<td>Number of successful bootstrap draws</td>
<td>1000</td>
</tr>
</tbody>
</table>

| Estimate | Std.err | Z-value | P(>|z|) |
|----------|---------|---------|--------|
| Y ~ M | (b) | 0.597 | 0.098 | 6.068 | 0.000 |
| X | (c) | 2.594 | 1.210 | 2.145 | 0.032 |
| Y ~ M | (a) | 2.739 | 0.999 | 2.741 | 0.006 |

Variances:

| Y | 108.700 | 17.747 |
| M | 105.408 | 16.556 |

Defined parameters:

| indirect | 1.636 | 0.645 | 2.535 | 0.011 |
| total | 4.230 | 1.383 | 3.059 | 0.002 |
Multiple mediator model

\[
\begin{align*}
model & \leftarrow ' \\
Y & \sim M1 + M2 + X \\
M1 & \sim X \\
M2 & \sim X
\end{align*}
\]

fit <- sem(model, data=myData, se="bootstrap")
summary(fit)
Longitudinal mediation models

```
model <- ' X2  ~  X1
     X3  ~  X2
     M2  ~  X1 + M1
     M3  ~  X2 + M2
     Y2  ~  X1 + M1 + Y1
     Y3  ~  X2 + M2 + Y2
',

fit <- sem(model, data=myData, se="bootstrap")
summary(fit)
```
Mediation with latent variables

\[ \text{model} <- \]
\[
# \text{latent variable definitions}
X \sim x_1 + x_2 + x_3 + x_4 \\
M \sim m_1 + m_2 + m_3 + m_4 \\
Y \sim y_1 + y_2 + y_3 \\

# regressions
M \sim X \\
Y \sim X + M \\
, \\
fit <- \text{sem}(\text{model, data=...})\]
Mediation with an ordinal mediation, and a binary outcome

\[
\text{model} <- ' \\
\text{ciguse} \sim c \times \text{intervention} + b \times \text{intention} \\
\text{intention} \sim a \times \text{intervention} \\
\text{naive.indirect} := a \times b \\
\text{naive.direct} := c,
\]

\[
\text{fit} <- \text{sem(model, data=myData,}
\text{ ordered=c("ciguse","intention")})
\]

\text{summary(fit)}

Muthén, B. Applications of Causally Defined Direct and Indirect Effects in Mediation Analysis using SEM in Mplus. (retrieved from www.statmodel.com)

**output**

Parameter estimates:

<table>
<thead>
<tr>
<th>Information</th>
<th>Expected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Errors</td>
<td>Robust.sem</td>
</tr>
</tbody>
</table>

| Estimate | Std.err | Z-value | P(>|z|) |
|----------|---------|---------|--------|

**Regressions:**

- ciguse ˜ tx (c)
  - tx: -0.130 (0.093), -1.402, 0.161
  - intent: 0.631 (0.042), 15.105, 0.000

- intent ˜ tx (a)
  - tx: -0.246 (0.089), -2.758, 0.006

**Thresholds:**

- cigus|t1: 0.760 (0.072), 10.491, 0.000
- intnt|t1: 0.525 (0.067), 7.844, 0.000
- intnt|t2: 0.970 (0.071), 13.572, 0.000
- intnt|t3: 1.378 (0.082), 16.710, 0.000

**Variances:**

- ciguse: 1.000
- intent: 1.000

**Defined parameters:**

- naive.indirct: -0.155 (0.057), -2.713, 0.007
- naive.direct: -0.130 (0.093), -1.402, 0.161
Computing ‘causal’ direct and indirect effects

- Imai et al. (2010) showed in a simulation study that the naive approach (taking the product of coefficients) results in biased estimates of the direct and indirect effects

- But we can compute the correct estimates using the parameter values as computed in a SEM

- References:

  Muthén, B. Applications of Causally Defined Direct and Indirect Effects in Mediation Analysis using SEM in Mplus.

  http://www.statmodel.com/examples/penn.shtml
lavaan syntax - computing probits and probabilities

```r
model <- ' 
ciguse ~ c*intervention +
       b*intention
intention ~ a*intervention

# label threshold for ciguse
ciguse | b0*t1

probit11 := (-b0+c+b*a)/sqrt(b^2+1)
probit10 := (-b0+c )/sqrt(b^2+1)
probit00 := (-b0 )/sqrt(b^2+1)

indirect := pnorm(probit11) - pnorm(probit10)
direct := pnorm(probit10) - pnorm(probit00)

OR.indirect := (pnorm(probit11)/(1-pnorm(probit11)))/
              (pnorm(probit10)/(1-pnorm(probit10)))

OR.direct := (pnorm(probit10)/(1-pnorm(probit10)))/
              (pnorm(probit00)/(1-pnorm(probit00)))
',
```
Defined parameters:

| Parameter   | Estimate | Std. Error | z value  | Pr(>|z|) |
|-------------|----------|------------|----------|----------|
| probit11    | -0.884   | 0.062      | -14.182  | 0.000    |
| probit10    | -0.752   | 0.070      | -10.714  | 0.000    |
| probit00    | -0.643   | 0.063      | -10.184  | 0.000    |
| indirect    | -0.037   | 0.014      | -2.647   | 0.008    |
| direct      | -0.034   | 0.024      | -1.403   | 0.161    |
| OR.indirect | 0.796    | 0.066      | 12.030   | 0.000    |
| OR.direct   | 0.829    | 0.111      | 7.450    | 0.000    |
The R package ‘mediation’

- References:
  


- ‘General’ approach to causal mediation analysis, based on the counterfactual framework

- accommodates linear and nonlinear relationships, parametric and nonparametric models, continuous and discrete mediators, various types of outcomes

- allows for a sensitivity analysis
The jobs example - linear model for outcome and mediator

- \textbf{job\_seek}: level of job-search self-efficacy
- \textbf{econ\_hard}: level of economic hardship pre-treatment

\begin{verbatim}
# model for M
model.M <- lm(job\_seek ~ treat +
              econ\_hard + sex + age,
data=jobs)

# model for Y
model.Y <- lm(depress2 ~ treat + job\_seek +
              econ\_hard + sex + age,
data=jobs)

# Estimation via quasi-Bayesian approx.
out <- mediate(model.M,
               model.Y,
              treat = "treat",
             mediator = "job\_seek")
summary(out)
\end{verbatim}
## Output

**Causal Mediation Analysis**

**Quasi-Bayesian Confidence Intervals**

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>95% CI Lower</th>
<th>95% CI Upper</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mediation Effect</td>
<td>-0.01600</td>
<td>-0.04191</td>
<td>0.00673</td>
<td>0.21</td>
</tr>
<tr>
<td>Direct Effect</td>
<td>-0.04091</td>
<td>-0.12296</td>
<td>0.04441</td>
<td>0.34</td>
</tr>
<tr>
<td>Total Effect</td>
<td>-0.05690</td>
<td>-0.14009</td>
<td>0.03313</td>
<td>0.22</td>
</tr>
<tr>
<td>Proportion via Mediation</td>
<td>0.23156</td>
<td>-1.70527</td>
<td>3.26928</td>
<td>0.44</td>
</tr>
</tbody>
</table>

**Sample Size Used:** 899

**Simulations:** 1000
The jobs example - binary outcome and ordered (K=4) mediator

- **job_disc**: The job_seek measure recoded into four categories from lowest to highest
- **work1**: Indicator variable for employment. 1 = employed

```r
# model for M
model.M <- polr(job_disc ~ treat +
                econ_hard + sex + age,
                data=jobs, method="probit", Hess=TRUE)

# model for Y
model.Y <- glm(work1 ~ treat + job_disc +
               econ_hard + sex + age,
               data=jobs,
               family=binomial(link="probit"))

out <- mediate(model.M, model.Y, treat = "treat", mediator = "job_disc")
summary(out)
```
Conclusions

- traditional software for mediation analysis: SEM
  - psychologists are very familiar with this approach
  - many results from the causal inference literature can be computed using SEM software (with care)
  - open-source software is available (R package ‘lavaan’)

- modern software for mediation analysis:
  - based on the causal inference literature
  - R package ‘mediation’
  - sensitivity analysis

- towards an integration of both approaches in one single coherent software package?
Thank you!