

lavaan: an R package for structural equation modeling

Yves Rosseel
Department of Data Analysis
Ghent University – Belgium

Utrecht – April 24, 2012

Overview

1. software for SEM
2. what is lavaan? why do we need lavaan?
3. features of lavaan
4. lavaan and the (computational) history of SEM

Software for SEM (commercial)

The big four

- LISREL
- EQS
- AMOS
- MPLUS

Others

- CALIS/TCALIS (SAS/Stat)
- SEPATH (Statistica)
- RAMONA (Systat)
- Stata 12
- ...

Software for SEM: non-commercial

- outside the R ecosystem: Mx, gllamm (Stata), ...
- three R packages:
 1. sem
 - developer: John Fox (since 2001)
 - for a long time the only option in R
 2. OpenMx
 - Mx reborn – website: <http://openmx.psyc.virginia.edu/>
 - free, but the solver (NPSOL) is (currently) not open-source
 3. lavaan
 - first public release: May 2010 (version 0.3-1)
 - current version: 0.4-12
- interfaces between R and commercial packages:
 - REQS, MplusAutomation

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()`
 - (item response theory (IRT) models)
 - (latent class + mixture models)
 - (multilevel models)
- the **lavaan** package is developed to provide useRs, researchers and teachers a free, open-source, but commercial-quality package for latent variable modeling
- the long-term goal of **lavaan** is to implement all the state-of-the-art capabilities that are currently available in commercial packages

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`

- journal paper:

Rosseel (2012). lavaan: an R package for structural equation modeling. *Journal of Statistical Software*, (??), ??-??.

- **lavaan** development:

`https://github.com/yrosseel/lavaan`

Why do we need lavaan?

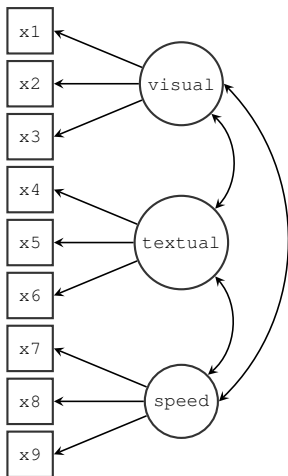
1. lavaan is for statisticians working in the field of SEM
 - it seems unfortunate that new developments in this field are hindered by the lack of open source software that researchers can use to implement their newest ideas
2. lavaan is for teachers
 - teaching these techniques to students was often complicated by the forced choice for one of the commercial packages
3. lavaan is for applied researchers
 - keep it simple, provide all the features they need

Features of lavaan

lavaan is reliable and robust

- extensive testing before a 'public' release on CRAN
- no convergence problems (for admissible models)
- numerical results are very close (if not identical) to commercial packages:
 - Mplus (if `mimic="Mplus"`, default)
 - EQS (if `mimic="EQS"`)

the lavaan model syntax



```
HS.model <- ' visual  =~ x1 + x2 + x3
             textual =~ x4 + x5 + x6
             speed   =~ x7 + x8 + x9
             '
```

```
fit <- cfa(model = HS.model,
           data = HolzingerSwineford1939)
```

```
summary(fit, fit.measures = TRUE,
        standardized = TRUE)
```

lavaan output

lavaan (0.4-13) converged normally after 41 iterations

Number of observations	301
Estimator	ML
Minimum Function Chi-square	85.306
Degrees of freedom	24
P-value	0.000

Chi-square test baseline model:

Minimum Function Chi-square	918.852
Degrees of freedom	36
P-value	0.000

Full model versus baseline model:

Comparative Fit Index (CFI)	0.931
Tucker-Lewis Index (TLI)	0.896

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-3737.745
Loglikelihood unrestricted model (H1)	-3695.092
Number of free parameters	21

Akaike (AIC)	7517.490
Bayesian (BIC)	7595.339
Sample-size adjusted Bayesian (BIC)	7528.739

Root Mean Square Error of Approximation:

RMSEA	0.092
90 Percent Confidence Interval	0.071 0.114
P-value RMSEA \leq 0.05	0.001

Standardized Root Mean Square Residual:

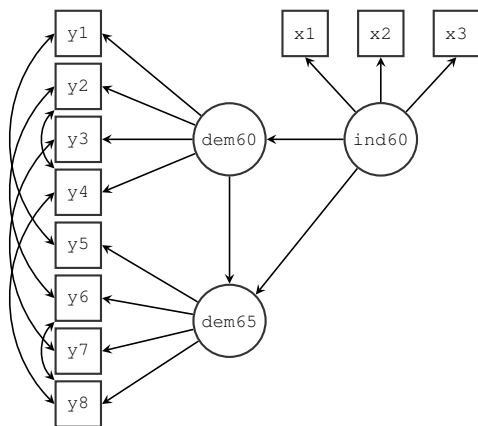
SRMR	0.065
------	-------

Parameter estimates:

Information Standard Errors	Expected Standard						
		Estimate	Std.err	Z-value	P(> z)	Std.lv	Std.all
Latent variables:							
visual =~							
x1		1.000				0.900	0.772
x2		0.553	0.100	5.554	0.000	0.498	0.424
x3		0.729	0.109	6.685	0.000	0.656	0.581
textual =~							
x4		1.000				0.990	0.852
x5		1.113	0.065	17.014	0.000	1.102	0.855

x6	0.926	0.055	16.703	0.000	0.917	0.838
speed =~						
x7	1.000				0.619	0.570
x8	1.180	0.165	7.152	0.000	0.731	0.723
x9	1.082	0.151	7.155	0.000	0.670	0.665
Covariances:						
visual ~~						
textual	0.408	0.074	5.552	0.000	0.459	0.459
speed	0.262	0.056	4.660	0.000	0.471	0.471
textual ~~						
speed	0.173	0.049	3.518	0.000	0.283	0.283
Variances:						
x1	0.549	0.114			0.549	0.404
x2	1.134	0.102			1.134	0.821
x3	0.844	0.091			0.844	0.662
x4	0.371	0.048			0.371	0.275
x5	0.446	0.058			0.446	0.269
x6	0.356	0.043			0.356	0.298
x7	0.799	0.081			0.799	0.676
x8	0.488	0.074			0.488	0.477
x9	0.566	0.071			0.566	0.558
visual	0.809	0.145			1.000	1.000
textual	0.979	0.112			1.000	1.000
speed	0.384	0.086			1.000	1.000

lavaan model syntax (2)



lavaan model syntax

```
# latent variable definitions
ind60 =~ x1 + x2 + x3
dem60 =~ y1 + y2 + y3 + y4
dem65 =~ y5 + y6 + y7 + y8

# regressions
dem60 ~ ind60
dem65 ~ ind60 + dem60

# residual covariances
y1 ~~ y5
y2 ~~ y6
y3 ~~ y7
y4 ~~ y8
y6 ~~ y8
```

lavaan features (0.4)

support for non-normal continuous data

- asymptotically distribution-free (ADF) estimation (Browne 1984)
- Satorra-Bentler scaled test statistic and robust standard errors
- Yuan-Bentler scaled test statistic and robust standard errors when data are both non-normal and missing (at random)
- bootstrapping: the naïve bootstrap and the Bollen-Stine bootstrap

support for missing data (fiml)

multiple groups and measurement invariance

linear and nonlinear equality and inequality constraints

defined parameters and mediation analysis

bootstrapping

what to expect in lavaan 0.5

- expected release: 11 May 2012
- support for categorical data (three-step/WLS approach)
- modularity + access to all lavaan internals

what to expect in lavaan 0.6-0.7

- Bayesian estimation
- support for multilevel data
- speed!
- ...
- technical documentation

The history of SEM, from a computational point of view

- several traditions in the SEM (software) world:
 - LISREL (Karl Jöreskog)
 - EQS (Peter Bentler)
 - Mplus (Bengt Muthén)
 - RAM-based approaches (AMOS, Mx, sem, OpenMx, ...)
- superficially, all SEM software packages produce the same results
- there are some subtle (and less subtle) differences in the output
- looking deeper, there are many computational differences

some differences

- matrix representation
 - standard number of matrices: LISREL: 8; Mplus: 4, EQS: 3, RAM: 2
- optimization algorithm
 - quasi-Newton, gradient-only + quasi-Newton, Gauss-Newton, ...
- variances constrained (strictly positive) versus unrestricted
- constrained optimization algorithm
 - mostly undocumented
 - a Lagrangian-multiplier variant, simple slacks, ...
- normal likelihood versus Wishart likelihood, ML versus GLS-ML (RLS)
 - N versus $N - 1$
 - GLS-ML based chi-square test statistic influences fit measures (CFI!)

some differences (2)

- Satorra-Bentler/Yuan-Bentler scaled test statistic
 - each program seems to use a different implementation
 - often asymptotically equivalent; but large differences in small samples
- categorical data using the limited information approach
 - Muthén 1984; Jöreskog 1994; Lee, Poon, Bentler (1992)
 - many ways to compute the asymptotic covariance matrix (needed for WLS)
- naive bootstrapping, Bollen-Stine bootstrapping
 - mostly undocumented; one-iteration bootstrap?
 - Bollen-Stine with missing data
- ...

lavaan and the history of SEM

- lavaan is in many areas still trying to catch up with commercial software; but instead of trying to implement one tradition (based on one program), lavaan tries to implement several traditions
- all fitting functions in lavaan have a `mimic` argument which can be set to "EQS" or "Mplus" respectively; "LISREL" is under development
- this was originally intended to convince users that lavaan could produce 'identical' results as the (commercial) competition
- it is now one of the main design goals of lavaan

lavaan and the future of SEM?

- we need to (re)evaluate old/new/unexplored computational methods in many areas (optimization, constrained inference, Bayesian techniques, limited information estimation, ...)
- lavaan should 'by default' implement best practices in all areas

End of part I