

Creating Path Model Diagrams

Appendix to *Latent Variable Modeling using R: A Step-by-Step Guide*

A. Alexander Beaujean

December 2013

Updated September 2015

Contents

1	L^AT_EX	2
2	Graphviz	2
3	R Packages	4
4	Graphical User Interfaces	5

There are variety of programs available to create path model diagrams.

1 \LaTeX

The path model diagrams used in this text were created in \LaTeX using the *PGF/TikZ* (Tantau, 2007), which is a very powerful graphics package. As an example, to create the diagram in Figure 2, I used the following syntax.

```
% This needs to be added to the LaTeX preamble
\usepackage{tikz}
\usetikzlibrary{positioning,shapes,arrows}
% This starts the function and defines the path model shapes
\begin{tikzpicture}[auto,>=latex,align=center,
latent/.style={circle,draw,thick,inner sep=0pt,minimum size=10mm},
manifest/.style={rectangle,draw,thick,inner sep=2pt,minimum size=10mm},
mean/.style={regular polygon,regular polygon sides=3,draw,thick,inner sep=0pt,minimum
size=10mm},
paths/.style={->,very thick,>=stealth'},
variance/.style={<->,thick,>=stealth',bend left=270,looseness=2},
]
% Draw, place, and label variables
\node [latent] (LV1) at (0,0) {g};
\node [manifest] (WR) [below =of LV1] {Word\\Reasoning};
\node [manifest] (SI) [left =of WR] {Similarities};
\node [manifest] (IN) [left =of SI] {Information};
\node [manifest] (MR) [right =of WR] {Matrix\\Reasoning};
\node [manifest] (PS) [right =of MR] {Picture\\Similarities};

\node [latent] (EIN) [below =of IN] {Error};
\node [latent] (ESI) [below =of SI] {Error};
\node [latent] (EWR) [below =of WR] {Error};
\node [latent] (EMR) [below =of MR] {Error};
\node [latent] (EPS) [below =of PS] {Error};

% Draw paths from LV to MV, with labels.
% The pos argument places the label up/down the path.
\draw [paths,above] (LV1) to node {a} (IN);
\draw [paths,above] (LV1) to node [pos=.75] {b} (SI);
\draw [paths] (LV1) to node {c} (WR);
\draw [paths] (LV1) to node [pos=.75] {d} (MR);
\draw [paths] (LV1) to node {e} (PS);

% Draw paths from residuals to MVs
\foreach \e in {IN, SI, WR, MR, PS}
\draw [paths] (E\e) to node {1} (\e);

% Draw residual error paths and label them
\foreach \x/\xlab in {EIN/f, ESI/g,EWR/h, EMR/i, EPS/j}
\draw [variance] (\x.south west) to node [swap] {\xlab} (\x.south east);
\end{tikzpicture}
```

Figure 1 *PGF/TikZ* syntax for the path diagram in Figure 2.

2 Graphviz

If you do not want to learn the \LaTeX language, another option is the Graphviz program (<http://www.graphviz.org>). An example Graphviz path diagram is shown in Figure 4.

```
digraph Simple_Factor {
// This makes all lines straight
```

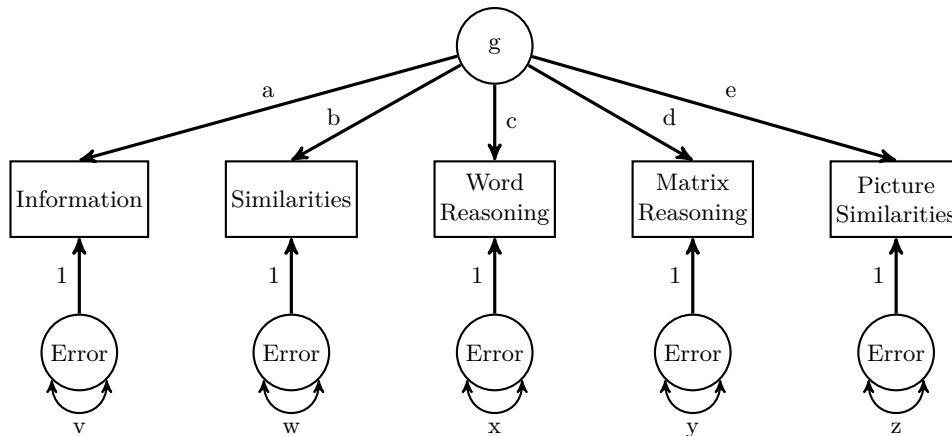


Figure 2 Latent variable model path diagram created with $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

```

splines=false;
node [shape="box"];

// Paths from g
g-> I[label="a"]
g-> S[label="b"]
g-> W[label=" c"]
g-> M[label="d"]
g-> P[label="e"]

// Paths from MVs to residuals
I->ui [dir=back, label=" 1"];
S->us [dir=back, label=" 1"]
W->uw [dir=back, label=" 1"]
M->um [dir=back, label=" 1"]
P->up [dir=back, label=" 1"]

// Define the shape of the latent variables
g[shape=circle]
ui[shape=circle, label="Error", fixedsize=TRUE, width=.6in]
us[shape=circle, label="Error", fixedsize=TRUE, width=.6in]
uw[shape=circle, label="Error", fixedsize=TRUE, width=.6in]
um[shape=circle, label="Error", fixedsize=TRUE, width=.6in]
up[shape=circle, label="Error", fixedsize=TRUE, width=.6in]

// Label the residuals' variance
ui->ui[dir=both, label=" f"]
us->us[dir=both, label=" g"]
uw->uw[dir=both, label=" h"]
um->um[dir=both, label=" i"]
up->up[dir=both, label=" j"]

// Label the MVs
I[label="Information"]
S[label="Similarities"]
W[label="Word \n Reasoning"]
M[label="Matrix \n Reasoning"]
P[label="Picture \n Similarities"]
}

```

Figure 3 *Graphviz* syntax for the path diagram in Figure 4.

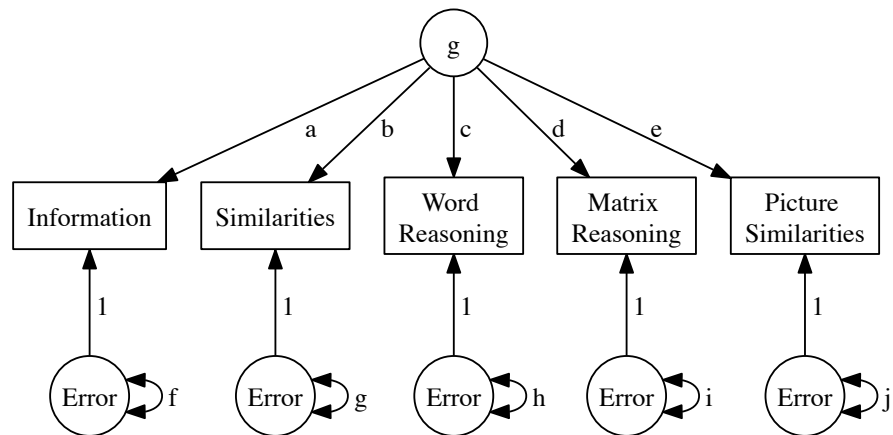


Figure 4 Latent variable model path diagram created with *Graphviz*.

3 R Packages

The `psych` package can create rough path diagrams via the `lavaan.diagram()` function. Example `lavaan.diagram()` syntax for the function is shown below, and the resulting output is shown in Figure 5.

```
library(psych)
lavaan.diagram(WiscIV.fit)
```

```
semPaths()
```

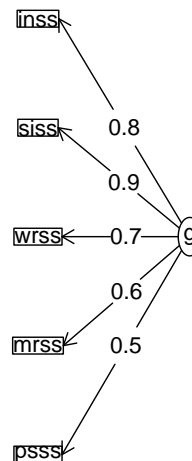


Figure 5 Example of a latent variable model created with `lavaan.diagram()` function in the `psych` package

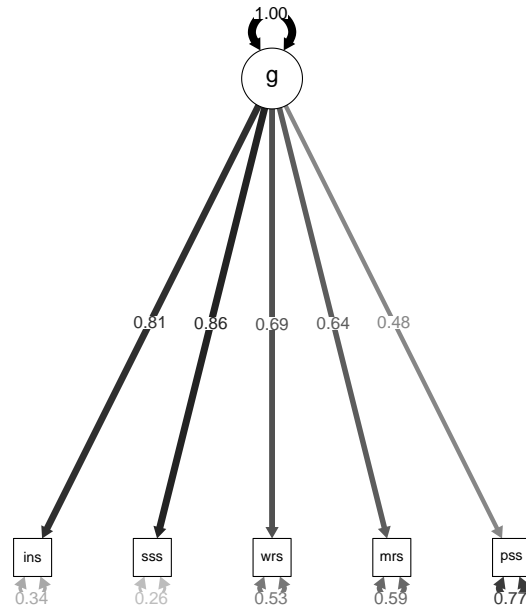


Figure 6 Example of a latent variable model created with `semPlot`.

The `semPlot` package can also create path diagrams using the `semPaths()` function. Example syntax is given below and the resulting diagram is given in Figure 6.

```

# path diagram via the semPlot package
library(semPlot)
semPaths(object=WiscIV.fit,what="std",edge.label.cex=1, curvePivot = TRUE,
fixedStyle=c("black",1), freeStyle=c("black",1), edge.color="black")
  
```

4 Graphical User Interfaces

There are many commercial programs with a graphical user interface (GUI) to create path models, such as PowerPoint, Keynote, OpenOffice's *Draw*, and OmniGraffle (<http://www.omnigroup.com/products/omnigraffle/>). *Onyx* (<http://onyx.brandmaier.de>) is a freeware program that takes a drawn path model as passes it to `OpenMx` package in **R** for analysis.

References

Tantau, T. (2007). The TikZ and PGF packages (Version 2.10) [Computer software]. Lübeck, Germany: University of Lübeck.