science for a changing world

Composites: The Equivalence of Various Specifications

Jim Grace

U.S. Department of the Interior
U.S. Geological Survey

1

This module provides a little more depth related to “Composites and Formative Indicators”

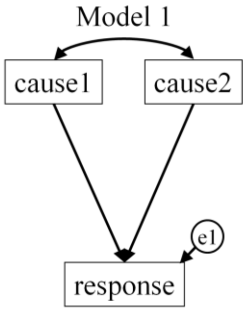
An appropriate citation for this material is

Grace, J.B. and Bollen, KA. 2008. Representing general theoretical concepts in structural equation models: the role of composite variables. Environmental and Ecological Statistics 15:191-213.

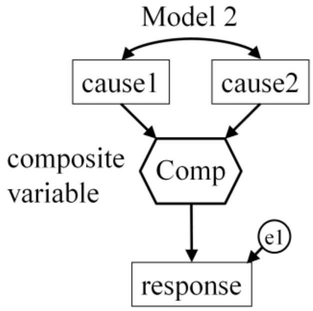
(http://www.odum.unc.edu/content/pdf/Bollen%20Grace%20Bollen%20(preprint%202008)%20Environ%20and%20Ecol%20Stats.pdf)

Notes: IP-056512; Support provided by the USGS Climate & Land Use R&D and Ecosystems Programs. I would like to acknowledge formal review of this material by Jesse Miller and Phil Hahn, University of Wisconsin. Many helpful informal comments have contributed to the final version of this presentation. The use of trade names is for descriptive purposes only and does not imply endorsement by the U.S. Government. Last revised 20141216. Questions about this material can be sent to sem@usgs.gov.

One use of composite variables is to represent the collective effects of a set of variables.



First step in compositing process.



Second step in compositing process.

USGS

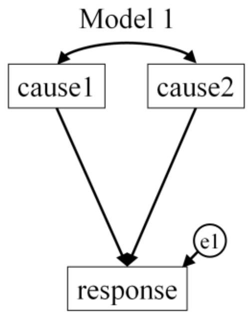
2

The inclusion of formative indicators in SE models is actually a complex topic. Here we only deal with the special case where we wish to represent “collective effects”. In this context, our simplest example is one where we have some model (Model 1 in this case) and wish to represent the collective effects of cause1 and cause2 on response using a composite variable “Comp”. We sometimes refer to this as “compositing”, which implies a two-step approach of (1) testing to see if both cause1 and cause2 contribute to the model and then (2) adding the composite to the model.

Note: Typically we don’t show both steps in our publications, only the results of the final model.

Note: In this example the label for the composite, “Comp” is a placeholder. In practice, we might label the composite in a more informative way that reflects what it is about the causes that influences the response.

Specification for Model 1




```
# Specification of Model 1

mod.1 <- 'response ~ cause1 + cause2'
```

USGS

3

We start in this example with the model omitting composites.

Select results for Model 1

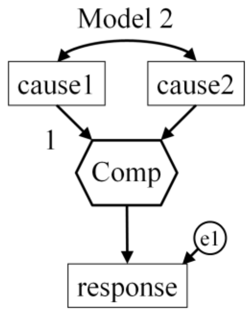
```
> summary(mod.1.fit, rsq= T, standardized=T)

lavaan (0.5-15) converged normally after   1 iterations

  Number of observations                            50

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

                   Estimate  Std.err  Z-value  P(>|z|)   std.all
Regressions:
  response ~
    cause1            0.838    0.117    7.163    0.000   0.684
    cause2            0.590    0.249    2.368    0.018   0.226

R-Square:
    response          0.699
```

4

USGS

Here are the basic results for the uncomposited model. I highlight the parameter estimate for cause1 because I will use that in some subsequent specifications.

Specification using lavaan's "short-hand" syntax (i.e., the "<~" operator) – one option.




```
# Specification using lavaan "short-hand" syntax

mod.2a <- 'Comp <~ 1*cause1 + cause2
           response ~ Comp'

# In this case I use "1" as the weight for cause1
```

USGS

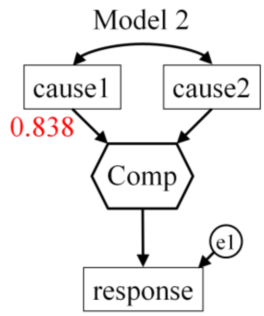
5

Lavaan has a special operator for composites. As with latent variables, we have to give the program some information when we add a composite variable to our model. For each variable in a model there are two basic quantities, its mean and its variance. Here we explicitly indicate that the composite has the same mean (i.e., its on the same scale) as the first indicator by pre-multiplying "cause1" by 1. Thus, we have set the parameter linking Comp to cause1 to 1.0.

Note: This common convention is not necessarily the best way to specify the composite in practice. Option 2, which follows, will show a generally superior approach.

Note: When you automatically set the weight for the composite to 1, you run the risk that that value is very far from the true ideal value. This can cause the model to fail to converge. Option 2 described in the following slides tries to avoid this problem.

Specification using lavaan “short-hand” syntax – second option.




```
# Here I use observed regression weight from Model 1
# as premultiplier for cause1

mod.2b <- 'Comp <~ 0.838*cause1 + cause2
           response ~ Comp'
```

6

Option 2 refers to setting the weight from cause1 to Comp to the value obtained from Model 1 (no composite). If you refer back to slide 4, you will see that the raw regression weight for the effect of cause1 on response = 0.838. For option 2, the preferred specification option, we use that regression weight to construct the composite instead of a default value of 1.

Select results for mod.2a.alt (option 2 specification)

```
> summary(mod.2b.fit, rsq=T, standardized=T)

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

                   Estimate  Std.err  Z-value  P(>|z|)  std.all
Composites:
  Comp <~
    cause1            0.838                                0.818
    cause2            0.590    0.305    1.937    0.053    0.270

Regressions:
  response ~
    Comp              1.000    0.140    7.163    0.000    0.836

R-Square:
    response          0.699
```

7

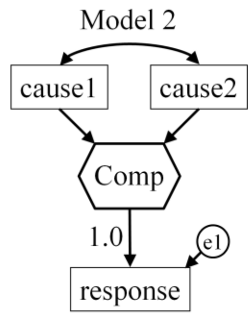
USGS

Note that with a composite specified, we have some new output. Because we used the regression weight from the uncomposited model, our Estimates for cause1 and cause2 are the same as for model1.

We again achieve an R-square of 0.699, just like for the non-composited model.

We now also get a line of estimates for the regression of response on Comp. The raw estimate will be 1.0 which is appropriate because this relationship is essentially the relationship between observed values "response" and their predicted scores "Comp" based on combined effects of cause1 and cause2. What is important for composite effects is the standardized coefficient. This represents how well the composite predicts the response. That idea is validated in this case by the fact that the square of the standardized coefficient, 0.836, is the R-square for the response (0.699) as expected.

Specification NOT using lavaan "short-hand" syntax.




```
# Without special operator, specification more
complex

mod.1c <- 'Comp =~ response
           Comp ~~ 0*Comp
           Comp ~ cause1 + cause2
           response ~~ response'
```

8

USGS

To create a composite in lavaan without using the special operator (only lavaan has a special command for creating a composite), several different specifications work. Here I show the simplest.

(1) Declare a latent variable using the "=~" operator. Be aware that lavaan automatically sets a coefficient of 1.0, so the implied equation is "Comp =~ 1*response".

(2) Set error variance of Comp to zero with "Comp ~~ 0*Comp"

(3) Declare an error variance for response with "response ~~ response"

(4) Regress Comp on its causes.

Results are equivalent for each specification, though the parameters that are fixed or estimated differs.

```
                      Estimate  Std.err  Z-value  P(>|z|)   std.all
Latent variables:
  Comp =~
    response             1.000                                0.836

Regressions:
  Comp ~
    cause1               0.838    0.117    7.163    0.000     0.818
    cause2               0.590    0.249    2.368    0.018     0.270

Variances:
    Comp                 0.000                                0.000
    response             3.097    0.619                       0.301

R-Square:
    Comp                 1.000
    response             0.699
```

9

USGS

Here are the results for mod.1g.