Regression Analysis with Auxiliary Variables

William Murrah

November 15, 2017

This is the third tutorial in a series that demonstrates how to us full information maximum likelihood (FIML) estimation using the R package lavaan. In this post, I demonstrate two methods of using auxiliary variable in a regression model with FIML. I am using data and examples from Craig Ender's website Applied Missing Data. The purpose of these posts is to make the examples on Craig's website, which uses Mplus, available to those who prefer to use lavaan.

Mplus allows you to use auxiliary variable when using FIML to include variables that help estimate missing values with variables that are not part of the analytic model. There may be variables that are correlated with variables with missing values or variables that are predictive of missing. However, these auxiliary variable are not part of the model you wish to estimate. See Craig's book Applied Missing Data Analysis for more information about auxiliary variables.

I attended a workshop where Craig showed us how to use the auxiliary command in Mplus to make use of auxiliary variables. However, lavaan does not have this option. He also showed us what he called a 'brute force' method to include auxiliary variables in Mplus. Here is how to do it in lavaan.

Brute Force Method

This model is the same as used in my last post, where job performance (jobperf) is regressed on wellbeing (wbeing) and job satisfaction (jobsat). In this example these three variables are the only ones which we want to model. However, tenure and IQ are related to missingness in these variables. So, we want to use them to help us better estimate our model of interest. If we included them as predictors in the regression model, it would allow us to use all the available information in these five variables, but it would change the model substantially. We can use auxiliary variables to better estimate the original model.

Import Data

First we import data, name the variables, and recode the -99's to NA.

```
# employeeAuxiliary.R ------------------------------------------------

# R packages used
library(lavaan)
# Import text file into R as a data frame.

employee <- read.table("path/to/file/employee.dat")

# Assign names to variables.

names(employee) <- c("id", "age", "tenure", "female", "wbeing", "jobsat",
                     "jobperf", "turnover", "iq")

# Replace all missing values (-99) with R missing value character 'NA'.
employee[employee==-99] <- NA
```

Create Regression Model Object (Brute Force)

Basically, the brute force method entails correlating the auxiliary variables with other auxiliary variable, the predictors, and the residuals for the outcome variable.

```
# The b1* and b2* are labels used in the Wald test below
model <- '
jobperf ~ b1*wbeing + b2*jobsat
wbeing ~~ jobsat
wbeing ~~ turnover + iq
jobsat ~~ turnover + iq
jobperf ~~ turnover + iq
turnover ~~ iq
'
```

Fit and Summarize the Model

```
fit <- sem(model, employee, missing='fiml', fixed.x=FALSE,
           meanstructure=TRUE)
summary(fit, fit.measures=TRUE, rsquare=T, standardize=T)
```

Wald test

Just as we did in the previous post.

```
lavTestWald(fit,
            'b1 == 0
             b2 == 0')
```

Using auxiliary Command in semTools

First, load the semTools package

```
library(semTools)
```

Create Regression Model Object

Next, create a model object with just the model of interest

```
model2 <- '
jobperf ~ wbeing + jobsat
'
```

Then, create a vector of the names of the auxiliary variables

```
aux.vars <- c('turnover', 'iq')
```

Fit the Model

Then, fit the model to the new model object.

```
fit2 <- sem(model2, employee, missing='fiml', meanstructure=TRUE, fixed.x=FALSE)
```

Using this model object, fit another model that incorporates the auxiliary variables using the sem.auxiliary function from the semTools package.

```
auxfit <- sem.auxiliary(model=fit2, aux=aux.vars, data=employee)
```

Finally, summarize the model object that includes the auxiliary variables.

```
summary(auxfit, fit.measures=TRUE, rsquare=TRUE, standardize=TRUE)
```

There you have it! Two way to use auxiliary variables in a regression model using lavaan.