Supplementary materials for: A closer look at fixed effects regression in structural equation modeling using `lavaan`

Henrik Kenneth Andersen[a]

[a]Chemnitz University of Technology, Institute of Sociology, Chair for Empirical Social Research, Thüringer Weg 9, 09126 Chemnitz, Germany

ARTICLE HISTORY
Compiled July 27, 2020

The following goes into some more detail on the comparison of traditional FE and FE-SEM models and discusses several opportunities to extend the basic FE model outlined in the main article, and provides concrete guidance on the implementation in `lavaan`. First, I verify that the FE-SEM model does, in fact, return essentially identical results compared to the more traditional methods. Then, I go over a number of possibilities to relax assumptions associated with the traditional FE model. Then, I discuss the issue of measurement error and show how we can use latent variables to deal with it and properly estimate the coefficients of interest. Then, I show a type of hybrid FE/RE model that allows us to control for time-invariant unobserved heterogeneity while including time-invariant predictors in the model.

1. A comparison with non-SEM methods

Just to be sure that the FE-SEM results do, in fact, line up with the more traditional methods outlined in Section 2 of the main article, we can use the long-format data (see the supplementary materials at) to run the typical FE model using the `plm` package (Croissant and Millo 2008). By default, the `plm` function assumes the dataframe is

CONTACT Henrik Kenneth Andersen. Email: `henrik.andersen@soziologie.tu-chemnitz.de`

structured so that the first two columns correspond to the individual and time indices, see the documentation or Croissant and Millo (2008).

```
library(plm)

# Run the FE model in plm
fe1 <- plm(y ~ x,
           effect = "individual", model = "within",
           data = df)
summary(fe1)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = y ~ x, data = df, effect = "individual", model = "within")
##
## Balanced Panel: n = 1000, T = 5, N = 5000
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -3.720902 -0.601550 -0.021365  0.600833  3.238716
##
## Coefficients:
##   Estimate Std. Error t-value  Pr(>|t|)
## x 0.293907   0.015635  18.798 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    4452.6
## Residual Sum of Squares: 4091.1
## R-Squared:      0.081192
## Adj. R-Squared: -0.14857
## F-statistic: 353.377 on 1 and 3999 DF, p-value: < 2.22e-16
```

From this, we see that the results are, indeed, essentially identical, with $\hat{\beta}_{FE-SEM} = 0.294$ (0.016) and $\hat{\beta}_{FE} = 0.294$ (0.016).

Other methods of estimating FE models work in the random or mixed effects model framework. For example, we can include the cluster means per individual of the time-varying independent variables, here x, in the equation to achieve within estimates (Mundlak 1978; Chamberlain 1980; Wooldridge 2002).

```
# Generate the cluster means for x per id
clusterMeanx <- aggregate(df$x, by = list(df$id), FUN = mean)
# Rename the columns
names(clusterMeanx) <- c("id", "xbar")
```

```
# Add the cluster means back into df
df <- merge(df, clusterMeanx, by = "id")
```

Here using the `plm` function in the `random` setup:

```
fe2 <- plm(y ~ x + xbar,
           effect = "individual", model = "random",
           data = df)
summary(fe2)
```

```
## Oneway (individual) effect Random Effect Model
##    (Swamy-Arora's transformation)
##
## Call:
## plm(formula = y ~ x + xbar, data = df, effect = "individual",
##     model = "random")
##
## Balanced Panel: n = 1000, T = 5, N = 5000
##
## Effects:
##                   var std.dev share
## idiosyncratic 1.0230  1.0115 0.862
## individual    0.1637  0.4046 0.138
## theta: 0.2546
##
## Residuals:
##       Min.    1st Qu.     Median    3rd Qu.       Max.
## -4.0697324 -0.6786344 -0.0094158  0.6678915  3.5329988
##
## Coefficients:
##              Estimate Std. Error z-value Pr(>|z|)
## (Intercept) -0.022559   0.019192 -1.1754   0.2398
## x            0.293907   0.015635 18.7983   <2e-16 ***
## xbar         0.757738   0.024015 31.5532   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    8878.9
## Residual Sum of Squares: 5112.1
## R-Squared:      0.42424
## Adj. R-Squared: 0.42401
## Chisq: 3682.01 on 2 DF, p-value: < 2.22e-16
```

And here using the `lmer` function of the `lme4` package (Bates et al. 2015) to estimate a mixed model:

```r
library(lme4)

# Run the mixed model in lmer with the cluster means for x
mixed1 <- lmer(y ~ x + xbar + (1 | id), data = df)
summary(mixed1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: y ~ x + xbar + (1 | id)
##    Data: df
##
## REML criterion at convergence: 14906.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.9358 -0.6464 -0.0106  0.6349  3.2002
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 0.1637   0.4046
##  Residual             1.0230   1.0115
## Number of obs: 5000, groups:  id, 1000
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) -0.02256    0.01919  -1.175
## x            0.29391    0.01563  18.798
## xbar         0.75774    0.02401  31.553
##
## Correlation of Fixed Effects:
##      (Intr) x
## x     0.000
## xbar  0.010 -0.651
```

In both cases, the models return the same estimates as the FE and FE-SEM models. Also, in both the `random` setup using the `plm` function, and the mixed model using the `lmer` function, we get estimates of the variance components, $\hat{\sigma}^2_\alpha$ and $\hat{\sigma}^2_\varepsilon$:

```r
# Print the variance components for the plm model
print(ercomp(fe2),
      digits = 3)
```

```
##                  var std.dev share
## idiosyncratic 1.023   1.011  0.86
## individual    0.164   0.405  0.14
## theta: 0.255
```

```r
# Print the variance components for the lmer model
print(VarCorr(mixed1),
      comp = c("Variance", "Std.Dev"),
      digits = 3)
```

```
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 0.164    0.405
##  Residual             1.023    1.011
```

From this we see both models report the same estimated variance components, $\hat{\sigma}^2_\alpha = 0.164$ and $\hat{\sigma}^2_\varepsilon = 1.023$, telling us that about 13.8% of the residual variance is due to the differences between individuals (shown in the `share` column of the `ercomp()` output). This is what is referred to as the intraclass correlation coefficient, or ICC (Hox 2010).

2. Extensions

2.1. *Relaxing assumptions meant to mimic traditional FE models*

There are a number of implicit assumptions attached to the typical FE model that can be relaxed in SEM. Some of these assumptions have been discussed already, and a fairly comprehenisve list of assumptions can be found in Bollen and Brand (2010). Here, I will go over just a few, concentrating on the implementation in `lavaan` and the opportunity to empirically test whether the adjustments are justified or not.

The assumptions we will discuss here pertain to the time-invariance of the effects of both the latent individual effects and the observed covariates, as well as a time-invariant error variance. We can also empirically test the correlation between the individual effects and the covariates to see whether a RE model is preferable to the FE model.

For example, we can rewrite the original FE equation as

$$y_{it} = \beta_t x_{it} + \lambda_t \alpha_i + \varepsilon_{it}$$

where β becomes β_t and the implicit regression weight of one turns to λ_t to highlight the fact that the effect of x as well as α on y may vary over time. We can furthermore easily relax the assumption of time-constant error variance, i.e., $\sigma^2_{\varepsilon_t}$. As noted in the

main article, the assumption regarding $\mathbb{E}[\alpha x_t]$ in $\mathbf{\Psi}$ determines whether we have an FE or RE model. We can set these to zero and test whether the RE model would be preferable to the FE model. In general, if the individual effects are truly uncorrelated with the model covariates, it is advisable to switch to an RE model since because it uses up less degrees of freedom, it will have smaller standard errors (Bollen and Brand 2010).

In the following `lavaan` code, we simply remove the factor loadings of one for the latent individual effect variable which allows them to be estimated freely at each timepoint. For the effect of the covariate, we can either delete the constraints `b` in `yt ~ b*xt` or give each regression a different label, e.g., `b1`, `b2`, `b3`, etc. Similarly, to allow the error variance to vary over time, we turn the constraints `e` into simple labels, i.e., `e1`, `e2`, `e3`, etc., or again just delete them. In fact, regarding the error variances, they will be estimated necessarily, and do not need to be explicitly mentioned in the model syntax at all. Finally, to move from an FE to an RE model, we could simply constrain the correlations between the individual effects and the covariates to zero, i.e., `a ~~ 0*x1 + 0*x2 + 0*x3 + 0*x4 + 0*x5`.

```
fe_sem_fullyrelaxed <- '
# Define individual effects variable
a =~ y1 + y2 + y3 + y4 + y5
# Regressions, constrain coefficient to be equal over time
y1 ~ b1*x1
y2 ~ b2*x2
y3 ~ b3*x3
y4 ~ b4*x4
y5 ~ b5*x5
# Allow unrestricted correlation between eta and covariates
a ~~ x1 + x2 + x3 + x4 + x5
# Alternatively: constrain all to 0 for RE model, or
# just individual correlations
# a ~~ 0*x1 + 0*x2 + 0*x3 + 0*x4 + 0*x5
x1 ~~ x2 + x3 + x4 + x5
x2 ~~ x3 + x4 + x5
x3 ~~ x4 + x5
x4 ~~ x5
# Constrain residual variances to be equal over time
y1 ~~ e1*y1
y2 ~~ e2*y2
y3 ~~ e3*y3
y4 ~~ e4*y4
```

```
y5 ~~ e5*y5
'
fe_sem_fullyrelaxed.fit <- sem( model = fe_sem_fullyrelaxed,
                                data = dfw,
                                estimator = "ML")
```

As outlined in Bollen and Brand (2010), the researcher has the opportunity to test each of the assumptions empirically and decide whether a more parsimonious, i.e., restrictive model is justifiable. For each assumption, a likelihood ratio test can be carried out to determine whether the improvement to model fit resulting from the relaxation of various assumptions is significant or whether the more parsimonious model is preferable after all.

If we use the original model `fe_sem.fit` (from the main article) as a starting point, the best strategy for testing these assumptions is to work in a stepwise fashion, relaxing one assumption at a time. We can begin by first constraining the correlation between α and x_t to zero (`re_sem`) for an RE model. If turning from an FE to an RE model does not significantly worsen model fit, we can go forward with the rest of the steps with the RE model. If, however, the fit does worsen significantly, it is likely better to stick with the FE model; moving forward then with it to see if a less restrictive FE model is preferable. We can perform a likelihood ratio test in `R` using the `anova()` function:

```
anova( fe_sem.fit, re_sem.fit)
```

```
## Chi-Squared Difference Test
##
##            Df   AIC   BIC   Chisq Chisq diff Df diff Pr(>Chisq)
## fe_sem.fit 32 30998 31111  30.137
## re_sem.fit 37 31809 31897 850.928     820.79       5  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The table that is generated shows a comparison of the nested models, in decending order according to degrees of freedom. The RE model does not estimate the correlations between the individual effects and the covariates, so it is more parsimonious and thus listed at the bottom. The `Chisq` column shows the χ^2 statistic for both models and the `Chisq diff` column calculates the difference between the two. Obviously, according

to the DGP, the correlation between the individual effects and x_t is not zero, so fixing these to zero leads to a substantial amount of misfit. The last column puts the χ^2 difference in relation to the difference in degrees of freedom and gives a p-value for the probability that the difference is solely due to chance. Here, the change in χ^2 is highly significant, so the FE model should be retained.

After now having established once and for all that FE is our preferred model, we can begin relaxing the rest of the assumptions. I show the following merely as a demonstration of the procedure, we know already from the DGP that the parsimonious model as specified in `fe_sem.fit` is appropriate. We can next allow the error variances (`fe_semb.fit`), the effect of x on y (`fe_semc.fit`) and finally the factor loadings of the individual effects (`fe_semd.fit`) all to vary over time.

```
anova( fe_sem.fit, fe_semb.fit, fe_semc.fit, fe_semd.fit)
```

```
## Chi-Squared Difference Test
##
##             Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## fe_semd.fit 20 31017 31189 25.140
## fe_semc.fit 24 31010 31162 25.764     0.6249       4     0.9603
## fe_semb.fit 28 31003 31135 26.686     0.9215       4     0.9215
## fe_sem.fit  32 30998 31111 30.137     3.4516       4     0.4853
```

Keep in mind that a less parsimonious model (fewer degrees of freedom) can never fit worse than a more parsimonious one (more degrees of freedom). I.e., chance variations due to sampling error mean that adding constraints to a model will tend to always worsen fit, at least minimally. The question here is whether the improvement to fit by loosening constraints is meaningful or not. In the table above, we should not expect any meaningful improvements moving from `fe_sem.fit` to `fe_semd.fit`. Here, using simulated data, we have the luxury of knowing that any significant differences in χ^2 are due to chance. With real data, it is up to the researcher to apply their best judgment and decide whether the results are plausible or not.

2.2. *Measurement error*

What if the observed variables are not measured perfectly? Then what we observe, call them $\tilde{x}_t$ and $\tilde{y}_t$ are composites of the true score we are after, i.e., x_t and y_t, plus

an additive measurement error portion:

$$\begin{aligned}\tilde{x}_t &= x_t + \upsilon_t,\\ \tilde{y}_t &= y_t + \nu_t.\end{aligned}$$

How does this affect our model? Well, first notice that measurement error in the dependent variable is typically less of a serious problem than measurement error in the independent variables. Let us assume again mean-centered variables so that we can ignore the intercept, and consider the following simple bivariate equation:

$$y = \beta x + \varepsilon$$

if y is measured imperfectly and what we observe is $\tilde{y} = y + \nu$, then we can rewrite the equation as:

$$\begin{aligned}(\tilde{y} - \nu) &= \beta x + \varepsilon\\ \tilde{y} &= \beta x + \varepsilon + \nu.\end{aligned}$$

The measurement error in y just gets added to the regression error. As long as ν is uncorrelated with x, then the regression coefficient will be unbiased (Pischke 2007; Wooldridge 2009). However, this will increase the error variance and thus make the estimates less precise.

We will look at the effect of measurement error in the dependent variable using an example shortly. For now though, let us be safe in the knowledge that the coefficient of interest is likely unbiased, and concentrate on the more serious problem of error in the independent variable.

The intuition behind the problem of measurement error in the independent variable(s) can be explained as follows. Take $\tilde{x} = x + \upsilon$ and substitute this into the equation for y:

$$\begin{aligned}y &= \beta x + \varepsilon\\ &= \beta(\tilde{x} - \upsilon) + \varepsilon\\ &= \beta\tilde{x} + (\varepsilon - \beta\upsilon).\end{aligned}$$

Since $\tilde{x}$ is obviously correlated with υ (unless the variance of υ is so small so that the

correlation is essentially negligible), then the composite error in this regression is also correlated with the independent variable and thus the estimated coefficient of β will be biased.

2.2.1. The consequences of measurement error

To demonstrate the effect of measurement error on the FE-SEM model, and then provide a strategy for dealing with measurement error in SEM, the simulated dataset generates multiple *indicators* of the independent and dependent variables that all measure the intended variable imprecisely. Returning to our panel data, we have three indicators of each the independent and dependent variable, per timepoint:

$$\tilde{x}_{kt} = x_t + \upsilon_{kt},$$
$$\tilde{y}_{kt} = y_t + \nu_{kt}$$

where $k = 1, 2, 3$ and $t = 1, ..., T$. This is like repeatedly presenting a respondent with a multi-item scale designed to measure things like stress, depression, xenophobia, etc. over the course of a panel study. To create the observed indicators, a random amount of measurement error (ranging from $\{\sigma^2_{\upsilon_k}, \sigma^2_{\nu_k}\} \in \{1.0, 1.1, 1.2, 1.3, 1.4, 1.5\}$) was added to the true variables, again see the simulation code.

Let us first focus on the issue of imprecise measurements of the independent variable of interest and run the same FE-SEM model above, but this time we will use one of the measurement error sullied indicators, here $\tilde{x}_{1t}$, instead of the true independent variable, x_t. As for the naming conventions in the `R` code, `x11` stands for the first indicator ($k = 1$) at the first point in time ($t = 1$), whereas for example `x35` stands for the third indicator ($k = 3$) at the fifth point in time ($t = 5$).

```
fe_sem2 <- '
# Define individual effects variable
a =~ 1*y1 + 1*y2 + 1*y3 + 1*y4 + 1*y5
# Regressions, constrain coefficient to be equal over time
# Now the imprecisely measured indicator tilde{x}_kt
# instead of the true variable x_t
y1 ~ b*x11
y2 ~ b*x12
```

```
y3 ~ b*x13
y4 ~ b*x14
y5 ~ b*x15
# Allow unrestricted correlation between eta and covariates
a ~~ x11 + x12 + x13 + x14 + x15
x11 ~~ x12 + x13 + x14 + x15
x12 ~~ x13 + x14 + x15
x13 ~~ x14 + x15
x14 ~~ x15
# Constrain residual variances to be equal over time
y1 ~~ e*y1
y2 ~~ e*y2
y3 ~~ e*y3
y4 ~~ e*y4
y5 ~~ e*y5
'
fe_sem2.fit <- sem( model = fe_sem2,
                    data = dfw,
                    estimator = "ML")
```

Now, for the sake of brevity, let us look just at the estimated coefficients for β.

```
summary( fe_sem2.fit)
```

```
...
##                    Estimate  Std.Err  z-value  P(>|z|)
##   y1 ~
##     x11        (b)    0.091    0.009    9.611    0.000
##   y2 ~
##     x12        (b)    0.091    0.009    9.611    0.000
##   y3 ~
##     x13        (b)    0.091    0.009    9.611    0.000
##   y4 ~
##     x14        (b)    0.091    0.009    9.611    0.000
##   y5 ~
##     x15        (b)    0.091    0.009    9.611    0.000
##
...
```

Obviously, the estimated coefficient $\hat{\beta} = 0.091$ is substantially smaller than the true population coefficient of $\beta = 0.3$. And the discrepancy is not just due to sampling error. In fact, we can derive the bias we are observing here.

For a simple bivariate regression model, it is straightforward to quantify the bias due to measurement error. It will be

$$
\begin{aligned}
\operatorname{Cov}(y, \tilde{x}) &= \mathbb{E}[y\tilde{x}] \\
&= \mathbb{E}[(\beta\tilde{x} + \varepsilon)\tilde{x}] \\
&= \mathbb{E}[\beta\tilde{x}^2 + \varepsilon\tilde{x}] \\
&= \beta \operatorname{Var}(\tilde{x}) \\
\hat{\beta} &= \frac{\operatorname{Cov}(y, \tilde{x})}{\operatorname{Var}(\tilde{x})} \\
&= \frac{\mathbb{E}[(\beta x + \varepsilon)(x + \upsilon)]}{\mathbb{E}[(x + \upsilon)^2]} \\
&= \frac{\mathbb{E}[\beta x^2 + \beta x\upsilon + \varepsilon x + \varepsilon\upsilon]}{\mathbb{E}[x^2 + 2x\upsilon + \upsilon^2]} \\
&= \beta \frac{\operatorname{Var}(x)}{\operatorname{Var}(x) + \operatorname{Var}(\upsilon)}.
\end{aligned}
$$

which results if we assume that $\mathbb{E}[x\upsilon] = 0$, $\mathbb{E}[x\varepsilon] = 0$, $\mathbb{E}[\tilde{x}\varepsilon] = 0$ and $\mathbb{E}[\varepsilon\upsilon] = 0$ (Wooldridge 2009). However, the model we are interested is not a bivariate model, so what was the point of showing the this? For one, it points out that the bias will always move the estimated coefficient closer to 0, since $\operatorname{Var}(x) \leq \operatorname{Var}(x) + \operatorname{Var}(\upsilon)$. This means positive effects will be biased downwards and negative effects biased upwards, always towards zero. This is why it is referred to as *attenuation bias*. Second, it will help to familiarize ourselves with this equation to better understand the one for the multivariate case.

Indeed, the magnitude of the bias in a multivariate model is somewhat more complex to derive, but it will be

$$
\hat{\beta} = \beta \frac{\operatorname{Var}(\theta)}{\operatorname{Var}(\theta) + \operatorname{Var}(\upsilon)}
$$

where θ is just the residual of a regression in which the true underlying variable is regressed on all other covariates. In this case, we need to regress x_t on α_1 and α_2 for: $x_t = \tau + \gamma_1\alpha_1 + \gamma_2\alpha_2 + \theta_t$ where τ is the intercept, and γ_1, γ_2 are the regression coefficients and θ_t is the residual (Wooldridge 2009, p. 318–320).

Normally it is not possible to reconstruct the bias since in cases where we have to rely on indicators, we would not know the true underlying variable. Furthermore, in the case of a fixed-effects model, the covariates are the unobserved time-invariant characteristics. However, because we are working with simulated data, we have every-

thing we need. Going back to the results above, we can get the residuals of x_t by either running a regression and saving the residuals, or we could skip a step and get them directly using the 'residual maker' matrix (Rüttenauer and Ludwig 2020) which is $\boldsymbol{M} = \boldsymbol{I} - \boldsymbol{A}(\boldsymbol{A}^\intercal\boldsymbol{A})^{-1}\boldsymbol{A}^\intercal$ and $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{\iota}_n & \boldsymbol{\alpha_1} & \boldsymbol{\alpha_2} \end{pmatrix}$ is the $n \times 3$ matrix of the covariates (plus a constant).

```
# Make the n x n identity matrix
Id <- diag( n)

# n x 2 matrix of covariates a1 and a2
A <- matrix( c( rep( 1, n), dfw$a1, dfw$a2),
             nrow = n, ncol = 3)

# The residual maker matrix M = I - A(A'A)^-1 A'
M <- Id - A %*% solve( t( A) %*% A) %*% t( A)

# Save the residuals, t for 'theta'
t <- M %*% dfw$x1

# Re-run the FE model from above with the 'true'
# independent variable for the correct estimate for beta
fe_sem.fit <- sem( model = fe_sem, data = dfw, estimator = "ML")

# The equation for the biased beta
lavInspect( fe_sem.fit, "list")[ 6, 14]*
  (( var( t))/( var( t) + var( dfw$x11 - dfw$x1)))
```

```
##           [,1]
## [1,] 0.1027325
```

From this we can see that the biased estimate above of $\hat{\beta} = 0.091$ roughly comes from $\beta\frac{\text{Var}(\theta_t)}{\text{Var}(\theta_t)+\text{Var}(\upsilon_t)} = 0.294\frac{1.054}{3.018} = 0.103$; 'roughly' because the equation here is the population equation. Due to sampling error, the estimates will tend vary slightly.

2.2.2. Using latent variables to deal with measurement error

The way we deal with measurement error in SEM is surprisingly similar to the logic of fixed-effects regression. Namely, if we have multiple cross-sectional observations of the underlying construct of interest, then we can define a latent variable that represents the common variance across those multiple variables. Contrast this with the use of longitudinal repeated measures to isolate the common variance across time.

So, if we do in fact have multiple cross-sectional indicators for the underlying variables of interest, then we can partition them into an explained and unexplained portion:

$$x_{kt} = \lambda^x_{kt}\xi_t + \delta_{kt},$$
$$y_{kt} = \lambda^y_{kt}\eta_t + \varepsilon_{kt},$$

where x_{kt} and y_{kt} are the k^{th} indicators, ξ_t and η_t are latent factors representing the common variance across the cross-sectional repeated measures, and δ_{kt} and ε_{kt} are the unexplained portions of x_t and y_t, respectively. The latent factors are linked to the observed indicators through the factor loadings λ_{kt}.

Thus, our FE regression equation changes from $y_t = \beta x_t + \alpha + \varepsilon_t$ to:

$$\eta_t = \beta\xi_t + \alpha + \zeta_t$$

where ζ_t represents the disturbance, in other words the residual of the latent dependent variable η_t. First, however, let us double-check that measurement error in the dependent variable only increases the error variance (thus also increasing standard errors and reducing R^2), but does not systematically bias the coefficients of interest. The next model uses the indicators of x and specifies latent variables (ξ_t, `xi` in the code) to represent the valid cross-sectional variance. The dependent variable in the model is one of the impercisely measured indicators of y.

```
fe_sem3 <- '
# Define individual effects variable
a =~ 1*y11 + 1*y12 + 1*y13 + 1*y14 + 1*y15
# Measurement model for independent variables, xi
xi1 =~ 1*x11 + x21 + x31
xi2 =~ 1*x12 + x22 + x32
xi3 =~ 1*x13 + x23 + x33
xi4 =~ 1*x14 + x24 + x34
xi5 =~ 1*x15 + x25 + x35
# Regressions, constrain coefficient to be equal over time
y11 ~ b*xi1
y12 ~ b*xi2
y13 ~ b*xi3
y14 ~ b*xi4
```

```
y15 ~ b*xi5
# Allow unrestricted correlation between eta and covariates
a ~~ xi1 + xi2 + xi3 + xi4 + xi5
xi1 ~~ xi2 + xi3 + xi4 + xi5
xi2 ~~ xi3 + xi4 + xi5
xi3 ~~ xi4 + xi5
xi4 ~~ xi5
# Constrain residual variances to be equal over time
y11 ~~ e*y11
y12 ~~ e*y12
y13 ~~ e*y13
y14 ~~ e*y14
y15 ~~ e*y15
'
fe_sem3.fit <- sem( model = fe_sem3,
                    data = dfw,
                    estimator = "ML")
```

```
summary( fe_sem3.fit)
```

```
...
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   y11 ~
##     xi1        (b)    0.299    0.029   10.302    0.000
##   y12 ~
##     xi2        (b)    0.299    0.029   10.302    0.000
##   y13 ~
##     xi3        (b)    0.299    0.029   10.302    0.000
##   y14 ~
##     xi4        (b)    0.299    0.029   10.302    0.000
##   y15 ~
##     xi5        (b)    0.299    0.029   10.302    0.000
...
```

The estimated coefficient here in model `fe_sem3.fit` is $\hat{\beta}_{y_{1t},\xi_t} = 0.299$ which is very close to the estimated coefficient in the first, correctly specified model `fe_sem.fit`, where $\hat{\beta}_{y_t,x_t} = 0.294$. Notice, however, that the standard error of the estimate is substantially larger, with 0.029 in `fe_sem3.fit` vs. 0.016 in `fe_sem.fit` in which y was measured without error. The explained variance (R^2) in the dependent variable was also much higher in the first model:

```
lavInspect( fe_sem.fit, "r2")[ 1:5]
```

```
##        y1        y2        y3        y4        y5
```

```
## 0.5893174 0.5928913 0.5894869 0.5853815 0.5845328
```

compared to the current model:

```
lavInspect( fe_sem3.fit, "r2")[ 1:5]
```

```
##       y11       y12       y13       y14       y15
## 0.3901187 0.3914316 0.3828739 0.3850362 0.3703136
```

Finally, to see the benefits of removing measurement error from the dependent variable in terms of standard errors and R^2 statistics, we can specify a model with latent variables representing the valid cross-sectional variance in y (`n` for η in the code).

```
fe_sem4 <- '
# Measurement model for dependent variable, n for eta
n1 =~ 1*y11 + y21 + y31
n2 =~ 1*y12 + y22 + y32
n3 =~ 1*y13 + y23 + y33
n4 =~ 1*y14 + y24 + y34
n5 =~ 1*y15 + y25 + y35
# Define individual effects variable
a =~ 1*n1 + 1*n2 + 1*n3 + 1*n4 + 1*n5
# Measurement model for independent variables, xi
xi1 =~ 1*x11 + x21 + x31
xi2 =~ 1*x12 + x22 + x32
xi3 =~ 1*x13 + x23 + x33
xi4 =~ 1*x14 + x24 + x34
xi5 =~ 1*x15 + x25 + x35
# Regressions, constrain coefficient to be equal over time
n1 ~ b*xi1
n2 ~ b*xi2
n3 ~ b*xi3
n4 ~ b*xi4
n5 ~ b*xi5
# Allow unrestricted correlation between eta and covariates
a ~~ xi1 + xi2 + xi3 + xi4 + xi5
xi1 ~~ xi2 + xi3 + xi4 + xi5
xi2 ~~ xi3 + xi4 + xi5
xi3 ~~ xi4 + xi5
xi4 ~~ xi5
# Constrain residual variances to be equal over time
n1 ~~ e*n1
n2 ~~ e*n2
n3 ~~ e*n3
n4 ~~ e*n4
n5 ~~ e*n5
'
```

```
fe_sem4.fit <- sem( model = fe_sem4,
                   data = dfw,
                   estimator = "ML")
```

```
summary( fe_sem4.fit)
```

```
...
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   n1 ~
##     xi1        (b)    0.264    0.023   11.515    0.000
##   n2 ~
##     xi2        (b)    0.264    0.023   11.515    0.000
##   n3 ~
##     xi3        (b)    0.264    0.023   11.515    0.000
##   n4 ~
##     xi4        (b)    0.264    0.023   11.515    0.000
##   n5 ~
##     xi5        (b)    0.264    0.023   11.515    0.000
...
```

Here, the effect $\hat{\beta}_{\eta_t,\xi_t}$ is somewhat further off of the true effect of 0.3 than the preceding models. This will depend on how the latent variables are estimated, which themselves will depend on the underlying correlations between the indicators. Again, if the main goal of the model is to avoid bias, it may be advisable to just leave the manifest dependent variable as it is, and worry about measurement error in the independent variables.

2.3. *Time-invariant predictors*

What if we do not just want to just control for the effects of all time-invariant variables, but investigate some of them in detail? Many time-invariant variables, like sex, birth cohort, nationality, education, etc. can be interesting on their own. And typically, many of these variables are readily available in a given dataset. The traditional OLS-based FE model does not allow for this, as it wipes out the effect of *all* time-invariant variables, whether observed or not.

In SEM, we can easily specify a type of *hybrid* FE/RE model (Bollen and Brand 2010) that allows us to control for time-invariant unobserved heterogeneity while also

investigating the effects of specific observed time-invariant predictors.[1]

In the next example, we continue with the most complex model we have specified so far, `fe_sem4.fit` in which measurement error in both the independent and dependent variables is accounted for using latent variables. Now, we would like as well to specifically investigate the effect of α_2 on the dependent variable. The equation for this model changes to: $\eta_t = \beta\xi_t + \alpha + \gamma\alpha_2 + \zeta_t$.

```
fe_sem5 <- '
# Measurement model for dependent variable, n for eta
n1 =~ 1*y11 + y21 + y31
n2 =~ 1*y12 + y22 + y32
n3 =~ 1*y13 + y23 + y33
n4 =~ 1*y14 + y24 + y34
n5 =~ 1*y15 + y25 + y35
# Define individual effects variable
a =~ 1*n1 + 1*n2 + 1*n3 + 1*n4 + 1*n5
# Measurement model for independent variables, xi
xi1 =~ 1*x11 + x21 + x31
xi2 =~ 1*x12 + x22 + x32
xi3 =~ 1*x13 + x23 + x33
xi4 =~ 1*x14 + x24 + x34
xi5 =~ 1*x15 + x25 + x35
# Regressions, constrain coefficient to be equal over time
n1 ~ b*xi1 + g*a2
n2 ~ b*xi2 + g*a2
n3 ~ b*xi3 + g*a2
n4 ~ b*xi4 + g*a2
n5 ~ b*xi5 + g*a2
# Allow unrestricted correlation between eta and covariates
a ~~ xi1 + xi2 + xi3 + xi4 + xi5 + 0*a2
a2 ~~ xi1 + xi2 + xi3 + xi4 + xi5
xi1 ~~ xi2 + xi3 + xi4 + xi5
xi2 ~~ xi3 + xi4 + xi5
xi3 ~~ xi4 + xi5
xi4 ~~ xi5
# Constrain residual variances to be equal over time
n1 ~~ e*n1
n2 ~~ e*n2
n3 ~~ e*n3
n4 ~~ e*n4
n5 ~~ e*n5
'
fe_sem5.fit <- sem( model = fe_sem5,
                    data = dfw,
```

[1] These types of models have become well known outside of SEM as well, see for example Allison (2011); Schunck (2013); Bell, Fairbrother, and Jones (2018).

```
                    estimator = "ML")
```

Keep in mind, based on the DGP, the true parameters are $\beta = 0.3$ and $\gamma = 0.45$.

```
summary( fe_sem5.fit)
```

```
...
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   n1 ~
##     xi1        (b)    0.265    0.023   11.515    0.000
##     a2         (g)    0.490    0.033   14.999    0.000
##   n2 ~
##     xi2        (b)    0.265    0.023   11.515    0.000
##     a2         (g)    0.490    0.033   14.999    0.000
##   n3 ~
##     xi3        (b)    0.265    0.023   11.515    0.000
##     a2         (g)    0.490    0.033   14.999    0.000
##   n4 ~
##     xi4        (b)    0.265    0.023   11.515    0.000
##     a2         (g)    0.490    0.033   14.999    0.000
##   n5 ~
##     xi5        (b)    0.265    0.023   11.515    0.000
##     a2         (g)    0.490    0.033   14.999    0.000
...
```

From this we can see that such a hybrid model is does a good job of estimating the coefficients of interest, with $\hat{\beta} = 0.265$ (0.023) and $\hat{\gamma} = 0.49$ (0.033).

It is important, however, to realize that the unbiasedness of $\hat{\gamma}$ in this model is dependent on the assumption that $\mathbb{E}[\zeta|\boldsymbol{\xi_t}, \alpha_2] = 0$. In other words, the idiosyncratic error is mean independent of $\boldsymbol{\xi_t} = (\xi_1, \xi_2, ..., \xi_T)$ as well as α_2. The first part is easier to accept because we are controlling for all potential time-invariant confounders that could induce a relationship between the independent variable and the error. The unbiasedness of $\hat{\gamma}$, on the other hand rests on the assumption that the time-invariant predictor is independent of the error. If α_2 represented the respondent's intelligence and η_t, the dependent variable, represented the respondent's income, for example, then $\hat{\gamma}$ would be biased if both were dependent on a third time-invariant variable, say level of schooling, if it is not controlled for. For this reason, we need to treat the regression on a time-invariant predictor like any other regular multivariate regression model and look to include all plausible potential confounders as controls in the model, or turn to

other methods, e.g., instrumental variables.

References

Allison, Paul. 2011. *Fixed Effects Regression Models*. Thousand Oaks: Sage Publications.

Bates, Douglas, Martin Mächler, Ben Bolker, and Steve Walker. 2015. "Fitting Linear Mixed-Effects Models Using lme4." *Journal of Statistical Software* 67 (1): 1–48.

Bell, Andrew, Malcolm Fairbrother, and Kelvyn Jones. 2018. "Fixed and random effects models: Making an informed choice." *Quality and Quantity* 53: 1051–1074.

Bollen, Kenneth, and Jennie Brand. 2010. "A General Panel Model with Random and Fixed Effects: A Structural Equations Approach." *Social Forces* 89(1): 1–34.

Chamberlain, Gary. 1980. "Analysis of Covariance with Qualitative Data." *Review of Economic Studies* 47(1): 225–238.

Croissant, Yves, and Giovanni Millo. 2008. "Panel Data Econometrics in R: The plm Package." *Journal of Statistical Software* 27 (2): 1–43.

Hox, Joop J. 2010. *Multilevel Analysis: Techniques and Applications. Second Edition*. New York, Hove: Routledge.

Mundlak, Yair. 1978. "On the Pooling of Time Series and Cross Section Data." *Econometrica* 46(1): 69–85.

Pischke, Jörn-Steffen. 2007. "Lecture Notes on Measurement Error." `http://econ.lse.ac.uk/staff/spischke/ec524/Merr_new.pdf`.

Rüttenauer, Tobias, and Volker Ludwig. 2020. "Fixed Effects Individual Slopes: Accounting and Testing for Heterogeneous Effects in Panel Data or Other Multilevel Models." *Sociological Methods and Research* Forthcoming.

Schunck, Reinhard. 2013. "Within and between estimates in random-effects models: Advantages and drawbacks of random effects and hybrid models." *The Stata Journal* 13(1): 65–76.

Wooldridge, Jeffery. 2002. *Econometric analysis of cross sectional and panel data*. Cambridge, Massachusetts: The MIT Press. ISBN 0-262-23219-7.

Wooldridge, Jeffery. 2009. *Introductory Econometrics: A Modern Approach, 4^{th} Edition*. Mason, Ohio: South-Western Cengage Learning.