Linear dynamic panel-data estimation using maximum likelihood and structural equation modeling

Richard Williams
University of Notre Dame
Department of Sociology
Notre Dame, IN
rwilliam@nd.edu

Paul D. Allison
University of Pennsylvania
Department of Sociology
Philadelphia, PA
allison@statisticalhorizons.com

Enrique Moral-Benito
Banco de España
Madrid, Spain
enrique.moral@gmail.com

Abstract. Panel data make it possible both to control for unobserved confounders and to include lagged, endogenous regressors. However, trying to do both simultaneously leads to serious estimation difficulties. In the econometric literature, these problems have been addressed by using lagged instrumental variables together with the generalized method of moments, while in sociology the same problems have been dealt with using maximum likelihood estimation and structural equation modeling. While both approaches have merit, we show that the maximum likelihood–structural equation models method is substantially more efficient than the generalized method of moments method when the normality assumption is met and that the former also suffers less from finite sample biases. We introduce the command `xtdpdml`, which has syntax similar to other Stata commands for linear dynamic panel-data estimation. `xtdpdml` greatly simplifies the structural equation model specification process; makes it possible to test and relax many of the constraints that are typically embodied in dynamic panel models; allows one to include time-invariant variables in the model, unlike most related methods; and takes advantage of Stata's ability to use full-information maximum likelihood for dealing with missing data. The strengths and advantages of `xtdpdml` are illustrated via examples from both economics and sociology.

Keywords: st0523, xtdpdml, linear dynamic panel-data, structural equation modeling, maximum likelihood

1   Introduction

Panel data make it possible both to control for unobserved confounders and to include lagged, endogenous regressors.[1] However, trying to do both simultaneously leads to serious estimation difficulties. In the econometric literature, these problems have been addressed by using lagged instrumental variables together with the generalized method of moments (GMM). In Stata, commands such as `xtabond`, `xtdpdsys`, and `xtdpd` have been used for these models.

Perhaps reflecting historical disciplinary differences, sociologists (Allison 2009; Bollen and Brand 2010) have often taken a different approach. As Allison and his colleagues show (Allison 2009; Allison, Williams, and Moral-Benito 2017), the same problems can be dealt with using maximum likelihood (ML) estimation of structural equation models (SEM). The ML–SEM method is substantially more efficient than the GMM method when the normality assumption is met and suffers less from finite sample biases. In Stata, the `sem` command can be used for this purpose. Unfortunately, the process for specifying these models with `sem` is tedious and error-prone.

In this article, we introduce a new command, `xtdpdml`, that fits dynamic panel-data models using ML. It works as a shell for `sem`, generating the necessary commands. It can also generate code for running these models in Mplus—a popular stand-alone package for structural equation modeling. `xtdpdml` tends to work best when panels are strongly balanced, the number of time points is relatively small (for example, fewer than 10), and there are no missing data. But it can also work well when these conditions are not met. Conversely, `xtdpdml` tends to be slower and have more convergence problems than popular alternatives, but there are ways to minimize these problems. We illustrate the multidisciplinary strengths and advantages of `xtdpdml` via examples from both economics and sociology.

`xtdpdml` greatly simplifies the SEM model specification process; makes it possible to test and relax many of the constraints that are typically embodied in dynamic panel models; allows one to include time-invariant variables in the model, unlike most related fixed-effects methods; and takes advantage of Stata's ability to use full-information maximum likelihood (FIML) for dealing with missing data. `xtdpdml` can also fit models involving lagged reciprocal causation and is sometimes superior to the `xtreg` command when data are missing or when time-invariant variables are used. By default, `xtdpdml` also reports a likelihood-ratio test of all overidentifying restrictions and provides access to other fit measures via the `sem` postestimation command `estat gof, stats(all)`. Many other `sem` postestimation commands can be used as well.

---

1. The terms "exogenous" and "endogenous" are defined in different ways in different research literature. Here we define them how the literature on structural equation modeling typically does, which is also how the Stata `sem` command does when fitting our models: endogenous variables are those that appear as dependent variables in at least one equation (including those that might also be independent variables in another equation), and exogenous variables are those that never appear as dependent variables. We elaborate below on how exogenous variables are further subdivided into different types in our models.

2 The cross-lagged panel model[2]

2.1 The GMM approach

Panel data have two major attractions for making causal inferences: the ability to control for unobserved, time-invariant confounders and the ability to fit models with lagged, endogenous regressors—which can be helpful in making inferences about causal direction.

Controlling for unobservables can be accomplished with well-known fixed-effects methods (such as the linear fixed-effects model that can be optionally fit with `xtreg`). For examining causal direction, the most popular approach has long been the cross-lagged panel model. In cross-lagged panel models, x and y at time t affect both x and y at time $t+1$. Economists typically refer to such models as dynamic panel models because of the lagged effect of the dependent variable on itself.

Unfortunately, attempting to combine fixed-effects models with cross-lagged panel models leads to serious estimation problems. The estimation difficulties include error terms that are correlated with predictors, the so-called incidental parameters problem, and uncertainties about the treatment of initial conditions (Allison, Williams, and Moral-Benito [2017]; also see Wooldridge [2010], Baltagi [2013], or Hsiao [2014] for additional review of the extensive literature on dynamic panel-data models).

The most popular econometric method for fitted dynamic panel models is the GMM, which relies on lagged variables as instruments. This method has been incorporated into several commercial software packages, usually named Arellano–Bond (AB) estimators. For example, Stata has the built-in `xtabond` command and the community-contributed `xtabond2` command (Roodman 2009).

While the AB approach provides consistent estimators of the coefficients, there is substantial evidence that the estimators are not fully efficient (Ahn and Schmidt 1995) and often perform poorly when the autoregressive parameter (the effect of a variable on itself at a later point in time) is near 1.0.

---

2. Parts of this section borrow heavily from Allison, Williams, and Moral-Benito (2017). See that article, freely available at http://journals.sagepub.com/doi/suppl/10.1177/2378023117710578, for an extended discussion. Also, because the `xtdpdml` model is a special case of the `sem` model, the Stata manuals contain additional technical information.

2.2   The ML–SEM alternative

Moral-Benito (2013; Moral-Benito, Allison, and Williams [2018]; also see Bai [2013])
shows that ML estimation can be accomplished in a way that eliminates the incidental
parameters problem without the need for special assumptions about initial conditions.
Moral-Benito uses two equations to specify his model.[3] They are

$$y_{it} = \lambda y_{it-1} + \mathbf{x}'_{it}\boldsymbol{\beta} + \mathbf{w}'_i\boldsymbol{\delta} + \alpha_i + \xi_t + \upsilon_{it} \quad (t = 1, \ldots, T)(i = 1, \ldots, N) \qquad (1)$$

where

   y_{it} is the value of y for individual i at time t,

   y_{i0} is the initial observation of y_{it}, treated as an exogenous variable,

   $\mathbf{x}_{it}$ is a vector of sequentially exogenous and predetermined time-varying variables,

   $\mathbf{w}_i$ is a vector of time-invariant, strictly exogenous variables,

   α_i is the unobservable time-invariant fixed effect,

   ξ_t captures unobserved common factors across units in the panel,

   υ_{it} is the time-varying error term,

and
$$E(\upsilon_{it}|\mathbf{y}_i^{t-1}, \mathbf{x}_i^t, \mathbf{w}_i, \alpha_i) = 0 \quad \forall i, t \qquad (2)$$
where $\mathbf{x}_i^t$ denotes a vector of the observations accumulated up to t and $\mathbf{y}_i^{t-1}$ denotes
a vector of y values accumulated up to $t - 1$. This implies, for example, that the
disturbance for y_5 is uncorrelated with predetermined variable x at times 1–5 but could
be correlated with x at later times, for example, x_6, x_7, etc. Put another way, the
predetermined variable x could be affected by earlier values of the dependent variable.
The meaning of each type of variable will become clearer as we proceed. Further details
on the econometric specification and the resulting likelihood function are provided by
Moral-Benito, Allison, and Williams (2018).

Condition (2) is the only assumption required for consistency and asymptotic nor-
mality (under fixed T when N tends to infinity). Although Moral-Benito's (2013) model
does not explicitly include strictly exogenous time-varying predictors, such predictors
are just a special case.

Allison, Williams, and Moral-Benito (2017) show that Moral-Benito's method can
be implemented with SEM software. The essential features of the ML–SEM method
for cross-lagged panel models with fixed effects were previously described by Allison
(2000, 2005a, 2005b, 2009), but his approach was largely pragmatic and computational.
Moral-Benito provided a rigorous theoretical foundation for this method.

---

3. Here and elsewhere, we have slightly modified Moral-Benito's notation to make it consistent with
`xtdpdml`'s.

The justification for using SEM software rests on the fact that (1) and (2) are a special case of the linear SEM proposed by Jöreskog (1978) and generalized by Bentler and Weeks (1980). In its most general form, their model may be compactly specified as

$$\mathbf{y} = \boldsymbol{\mu} + \mathbf{B}\mathbf{y} + \boldsymbol{\Gamma}\mathbf{x} \tag{3}$$

where $\mathbf{y}$ is a $p \times 1$ vector of endogenous variables that may be either observed or latent, $\mathbf{x}$ is a $k \times 1$ vector of exogenous variables that also may be either observed or latent (including any disturbance terms in the model), $\boldsymbol{\mu}$ is a vector of intercepts, and $\mathbf{B}$ and $\boldsymbol{\Gamma}$ are matrices of coefficients. The endogenous vector $\mathbf{y}$ and any latent variables in $\mathbf{x}$ are assumed to have a multivariate normal distribution conditional on the observed exogenous variables. The $\mathbf{B}$ matrix has zeros on the main diagonal, and both $\mathbf{B}$ and $\boldsymbol{\Gamma}$ may have many additional restrictions. Most commonly, these restrictions are to set certain parameters equal to 0, but there may also be equality restrictions. The remaining parameter $\boldsymbol{\Theta}$ is the variance matrix for $\mathbf{x}$, which usually has many elements set to 0.

Equations (1) and (2) are a special case of (3), in that without loss of generality, we treat x_{it} and w_i as scalars rather than vectors. We then have $\mathbf{y}' = (y_{i1}, \ldots, y_{iT})$, $\mathbf{x}' = (\alpha_i, w_i, y_{i0}, x_{i1}, \ldots, x_{iT}, v_{i1}, \ldots, v_{iT})$, and $\boldsymbol{\mu}' = (\xi_1, \ldots, \xi_T)$. For $\boldsymbol{\Gamma}$, we have

$$\boldsymbol{\Gamma} = \begin{bmatrix} 1 & \delta & \lambda & \beta & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\ 1 & \delta & 0 & 0 & \beta & \cdots & 0 & 0 & 1 & \cdots & 0 \\ 1 & \delta & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 0 & 0 & 0 & \ddots & 0 \\ 1 & \delta & 0 & 0 & 0 & \cdots & \beta & 0 & 0 & \cdots & 1 \end{bmatrix}$$

and for $\mathbf{B}$,

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ \lambda & 0 & 0 & \cdots & 0 & 0 \\ 0 & \lambda & 0 & \cdots & 0 & 0 \\ 0 & 0 & \lambda & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & \lambda & 0 \end{bmatrix}$$

For $\boldsymbol{\Theta}$, the following covariances are set to 0:

- α with w

- α with all v

- w with all v

- all v with each other

- x_{it} with v_{is} whenever $s \geq t$

All other elements of $\mathbf{\Theta}$ are left unrestricted. Note that α is allowed to correlate with x and x is allowed to correlate with all prior realizations of υ as a consequence of (2). The restriction that $\text{cov}(\alpha, w) = 0$, while perhaps undesirable, is essential for identification. That is, we must assume that the fixed effects are uncorrelated with any time-invariant variables.[4]

Figure 1 displays a path diagram of this model for the case in which $T = 3$, with no w variables.[5] That is, we have only the y variables and the predetermined x variables. Notice that all the x variables are allowed to freely correlate with each other and with y_0, which is treated like any other exogenous variable. Similarly, the latent variable α (enclosed in a circle) is allowed to correlate with all the exogenous variables, including y_0. α affects each y variable (with a coefficient of 1, not shown). The coefficients for the effects of the x's on the y's are constrained to be the same at all three time points, but this constraint can be easily relaxed.
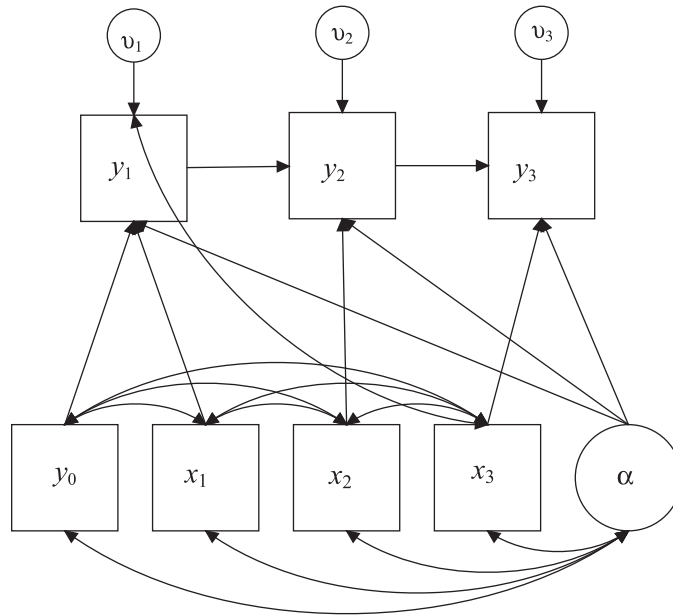
Figure 1. Path diagram for dynamic panel model with $T = 3$

What makes x predetermined in this diagram are the correlations between υ_1 and both x_2 and x_3 and the correlations between υ_2 and x_3. If these correlations were omitted, x would be strictly exogenous rather than predetermined. Again, the rule is that, for any predetermined variable, x at time t is allowed to correlate with the error term for y at any prior time point.

---

4. For an alternative parameterization and a derivation of the likelihood function, see Moral-Benito, Allison, and Williams (2018).
5. The path diagram in figure 1 was produced by Mplus version 7.4.

How do the assumptions of ML–SEM differ from those of AB? ML–SEM makes stronger assumptions in three respects. First, and most importantly, ML–SEM assumes multivariate normality for all observed endogenous and exogenous variables, while AB makes no distributional assumptions.[6] However, ML–SEM produces consistent estimators even when the normality assumption is violated (Moral-Benito 2013). And if there is concern about normality, robust standard errors and other methods (see section 4.4) can be used for constructing confidence intervals and hypothesis tests. Second, to identify the effects of time-invariant variables, we introduced the assumption that $cov(\alpha, w) = 0$. But if you have any reason to doubt that assumption, you can just exclude time-invariant variables from the model. They will still be controlled as part of the α term. Lastly, ML–SEM uses the moment restrictions implied by the assumption that there is no serial correlation in the error terms in (1). Although using these restrictions was recommended by Ahn and Schmidt (1995) to improve efficiency, they have generally not been incorporated into AB estimation, because they imply nonlinear estimating equations.

On the other hand, ML–SEM makes it possible to relax many assumptions that are built into AB. Most notably, the default in `xtdpdml` is to allow for an unrestricted effect of time itself and different error variances at each time point. It is also possible to allow α, the latent variable for the individual effects, to have different coefficients at different time points. Also, as shown, a fixed-effects model is fit. But by constraining the correlations between `Alpha` and the exogenous variables to be zero, the fixed-effects model becomes a random-effects model. Section 4 provides examples of how relaxing and imposing constraints greatly enhances the power and flexibility of the `xtdpdml` approach. Note that options that make it possible to impose or relax constraints do not fundamentally alter the underlying model; rather, they make it possible to fit and test special cases of it.

Allison, Williams, and Moral-Benito (2017) and Moral-Benito (2013) claim that the SEM approach has several advantages over both GMM methods and previous ML methods: there is no "incidental parameters" problem; initial conditions are treated as completely exogenous and do not need to be modeled; no difficulties arise when the autoregressive parameter (the effect of lagged y on y) is at or near 1.0; missing data are easily handled by FIML; coefficients can be estimated for time-invariant predictors (the standard AB method cannot do this, because it uses difference scores, which causes all time-invariant variables to drop out); and many model constraints can be easily relaxed, tested, or both.

---

6. More specifically, the *Stata 15 Structural Equation Modeling Reference Manual* (2017; 46) says that standard linear SEMs generally assume that the observed endogenous variables, the observed exogenous variables, the latent endogenous variables, and the latent exogenous variables have a joint normal distribution. We do not have latent endogenous variables in our models, but the error terms and `Alpha` are latent exogenous variables. Page 46 further clarifies that "although it is typical to assume joint normality of all variables when deriving the standard linear SEM, joint normality is not strictly necessary. The lesser assumption of joint normality conditional on the observed exogenous variables is sufficient. Even the normality assumption can be relaxed and replaced with i.i.d., and even the i.i.d. assumption can be relaxed." Section 4.4 of this article shows how the independent and identically distributed assumption can be relaxed.

Further, it is well known that likelihood-based approaches (for example, ML) are preferred to method-of-moments (for example, GMM) counterparts in terms of finite-sample performance (see Anderson, Kunitomo, and Sawa [1982]) and that ML is more efficient than GMM under normality. Moral-Benito (2013) compares the widely used panel GMM estimator of Arellano and Bond (1991) with its likelihood-based counterpart and confirms these results in the case of dynamic panel models with predetermined regressors.

Allison, Williams, and Moral-Bento (2017) and Moral-Benito, Allison, and Williams (2018) ran several simulation studies to compare AB and ML–SEM under a wide variety of plausible conditions. In their examples, the ML approach generally works at least as well as AB and is often better. They find that ML–SEM produces approximately unbiased estimates under all the studied conditions; confidence interval coverage was excellent; for the autoregressive parameter, the downward bias in the AB estimator was much more substantial than ML–SEM; and AB's relative efficiency was also poorer. Further, the larger the autoregressive parameter, the larger the AB bias. They also found that ML was less biased than AB when the disturbances were not normally distributed.

2.3   The basic xtdpdml command

To show specifically how the SEM approach can be used in Stata, Allison, Williams, and Moral-Benito (2017) reanalyze data described by Cornwell and Rupert (1988) for 595 household heads who reported a nonzero wage in each of 7 years from 1976 to 1982. The variables are wks = number of weeks employed in each year; union = 1 if wage set by union contract, else 0, in each year; lwage = ln(wage) in each year; and ed = years of education in 1976. The model to be fit is

$$\texttt{wks}_{it} = \lambda \texttt{wks}_{it-1} + \beta_2 \texttt{lwage}_{it-1} + \beta_3 \texttt{union}_{it-1} + \delta \texttt{ed}_i + \alpha_i + \xi_t + v_{it}$$

with union treated as predetermined and lwage and ed treated as strictly exogenous. Here is the Stata sem code (adapted from Allison, Williams, and Moral-Benito [2017]):

```
use https://www3.nd.edu/~rwilliam/statafiles/wages, clear
keep wks lwage union ed id t
xtset id t
reshape wide wks lwage union, i(id) j(t)
sem (wks2 <- wks1@b1 lwage1@b2 union1@b3 ed@b4 Alpha@1 E2@1)        ///
        (wks3 <- wks2@b1 lwage2@b2 union2@b3 ed@b4 Alpha@1 E3@1)    ///
        (wks4 <- wks3@b1 lwage3@b2 union3@b3 ed@b4 Alpha@1 E4@1)    ///
        (wks5 <- wks4@b1 lwage4@b2 union4@b3 ed@b4 Alpha@1 E5@1)    ///
        (wks6 <- wks5@b1 lwage5@b2 union5@b3 ed@b4 Alpha@1 E6@1)    ///
        (wks7 <- wks6@b1 lwage6@b2 union6@b3 ed@b4 Alpha@1),        ///
        var(e.wks2@0 e.wks3@0 e.wks4@0 e.wks5@0 e.wks6@0) var(Alpha) ///
        cov(Alpha*(ed)@0) cov(Alpha*(E2 E3 E4 E5 E6)@0)            ///
        cov(_OEx*(E2 E3 E4 E5 E6)@0) cov(E2*(E3 E4 E5 E6)@0)       ///
        cov(E3*(E4 E5 E6)@0) cov(E4*(E5 E6)@0) cov(E5*(E6)@0)      ///
        cov(union3*(E2)) cov(union4*(E2 E3)) cov(union5*(E2 E3 E4))  ///
        cov(union6*(E2 E3 E4 E5))                                  ///
        iterate(250) technique(nr 25 bhhh 25) noxconditional
```

We will explain the different components of the model in a moment, but even just glancing at the code underscores the difficulty of the task. For the SEM approach, data need to be in wide format; note that most dynamic panel datasets will be in long format. Coding is lengthy and error prone; there is a separate equation for each time period, there are many constraints across equations, and getting the covariance structure right is especially difficult. Output (not shown) is voluminous and highly repetitive because of the many equality constraints across time. Stata's limitations make the coding less straightforward than we might like. Stata will not allow covariances between predetermined x's (to be defined shortly) and the y residuals. The `xtdpdml` command therefore fixes the variances of most of the y residuals at 0 (in effect, dropping them from the model) and creates new latent exogenous variables (`E2`, `E3`, etc.) that replace the residuals and can be correlated with the predetermined x's.

`xtdpdml` avoids most of these problems. Below is equivalent coding using `xtdpdml` and the resulting output:

```
. use https://www3.nd.edu/~rwilliam/statafiles/wages, clear
. xtset id t
        panel variable:  id (strongly balanced)
         time variable:  t, 1 to 7
                 delta:  1 unit
. xtdpdml wks L.lwage, inv(ed) predetermined(L.union)
Highlights: Dynamic Panel Data Model using ML for outcome variable wks
```

wks	Coef.	OIM Std. Err.	z	P>|z|	[95% Conf. Interval]	
wks						
wks L1.	.1871266	.0201939	9.27	0.000	.1475473	.2267059
lwage L1.	.6417879	.4842304	1.33	0.185	-.3072862	1.590862
union L1.	-1.191361	.5168944	-2.30	0.021	-2.204455	-.1782663
ed	-.1122268	.0559478	-2.01	0.045	-.2218824	-.0025712

```
# of units = 595. # of periods = 7. First dependent variable is from period 2.
Constants are free to vary across time periods
LR test of model vs. saturated: chi2(71)  =     110.23, Prob > chi2 =  0.0020
IC Measures: BIC =   25470.43  AIC =   24772.64
Wald test of all coeff = 0: chi2(4) =       90.09, Prob > chi2 =  0.0000
```

One short command generates the equivalent of the 13 lines of `sem` code shown earlier. `xtdpdml` also temporarily reshaped the data to wide format.

Unless the user requests otherwise, only the most critical output is shown. By default, all variable coefficients (but not the constants or the error variances) are constrained to be equal across time. Therefore, only the first equation (in this case for time 2) must be presented. The likelihood-ratio statistic provides an overall goodness-of-fit test. This tests all the constraints on the variances and covariances that are

implied by the model. The Bayesian information criterion (BIC) and Akaike informa-
tion criterion (AIC) statistics (which could also be obtained via the `estat ic` command)
are included in the output. Note that these statistics could not be computed correctly
if you were using a highlights-only file, which we describe shortly. The Wald statistic
tests the null hypothesis that all the variables in the model have coefficients of zero. In
this case, where coefficients are constrained to be the same across all time periods, it
produces the same results as the `sem` postestimation command `estat eqtest`. When
some coefficients are free to differ across time periods, `estat eqtest` tests for each time
period separately, whereas `xtdpdml` tests all coefficients for all times simultaneously.

   `xtdpdml` obviously provides a much simpler syntax. The reason it is not even simpler
(and why the `sem` coding is so difficult) is that there are several possible types of
independent variables in the model:

   The lag 1 value of y (for example, `L1.wks`) is included by default. This can be
changed with the `ylags()` option, for example, `ylags(1 2)`, `ylags(2 4)`. Specifying
`ylags(0)` excludes all lagged values of y.

   Strictly exogenous time-varying predictors are those that (by assumption) are uncor-
related with the error terms at all points in time. These variables are listed immediately
after the dependent variable, before the comma. Time-series notation can be used; for
example, `xtdpdml y L1.lwage L2.lwage` would include the first and second lagged
values of wages as independent variables.

   Predetermined variables, also known as sequentially or weakly exogenous, are vari-
ables that might be affected by prior values of the dependent variables or are correlated
with them for some other reason.[7] In the current example, we allow for the possibility
that weeks worked in one year can affect union status in later years. Predetermined
variables are specified with the `predetermined()` option. Mechanically, the y residuals
are allowed to correlate with the later-in-time values of the predetermined variables.

   Time-invariant variables are variables whose values are constant across time, such
as year born. In the current example, years of education do not vary across time. These
variables are specified with the `inv()` option. The ability to use time-invariant variables
in the model is one of the advantages of the SEM approach over methods based on first
differences like AB.

---

7. We consider predetermined variables to be exogenous because they are not dependent variables in
   any equation. Allowing variables to correlate does not necessarily mean that one is a cause of the
   other; for example, they might be correlated because of some omitted common cause. Further,
   predetermined variables are labeled as exogenous in Stata's `sem` output. But other terminologies
   might label predetermined variables as endogenous if it is definitely believed that earlier values of
   the dependent variables are affecting later values of the independent variables.

Also automatically included in each model is the latent exogenous variable α (which the `sem` code calls `Alpha`). `Alpha` represents the "fixed effects" that are common to all equations across time. `Alpha` can freely covary with all the time-varying observed exogenous variables (both strictly exogenous and predetermined) but not with the time-invariant observed exogenous variables. As Allison, Williams, and Moral-Benito (2017) say, "This is exactly what we want to achieve in order for `Alpha` to truly behave as a set of fixed effects". To further clarify, Allison (2009, 2–3) explains that

> [i]n a random-effects model, the unobserved variables are assumed to be uncorrelated with (or, more strongly, statistically independent of) all the observed variables. In a fixed-effects model, the unobserved variables are allowed to have any association whatever with the observed variables (which turns out to be equivalent to treating the unobserved variables as fixed parameters). Unless you allow for such associations, you haven't really controlled for the effects of the unobserved variables. This is what makes the fixed effects approach so attractive.

3   The xtdpdml command and syntax

The general syntax is

xtdpdml y $\left[\,varlist\,\right]$ $\left[\,\right.$, inv(*varlist*) <u>predet</u>ermined(*varlist*) <u>ylags</u>(*numlist*) wide
  <u>staywide</u> tfix std std2(*varlist*) evars alphafree xfree xfree2(*varlist*)
  yfree yfree2(*numlist*) constinv nocsd errorinv re <u>title</u>(*string*) <u>details</u>
  <u>show</u>cmd gof tsoff *display_options* <u>coefl</u>egend <u>dec</u>imals(*integer*)
  <u>mplus</u>(*filenamestub*$\left[\,\right.$, *mplus_options*$\left.\,\right]$) <u>lavaan</u>(*filenamestub*$\left[\,\right.$, replace$\left.\,\right]$)
  <u>semfile</u>(*filename*$\left[\,\right.$, replace$\left.\,\right]$) <u>store</u>(*stubname*) <u>dryrun</u> <u>iterate</u>(#)
  <u>technique</u>(*methods*) *maximize_options* semopts(*options*) fiml v12
  <u>skipcfatransform</u> <u>skipconditional</u> <u>altstart</u> <u>method</u>(*method*) vce(*vcetype*) $\left.\,\right]$

  varlist is time-varying strictly exogenous variables.

3.1   Options

The following is a description of the command's options:

Independent variables (other than strictly exogenous)

inv(*varlist*) specifies time-invariant exogenous variables, for example, year of birth.

predetermined(*varlist*) specifies predetermined variables, also known as sequentially exogenous variables. Predetermined variables can be affected by prior values of the dependent variable. Time-series notation can be used.

ylags(*numlist*) specifies lagged values of y to be included in the model. The default is ylags(1). Different or multiple lags can be specified; for example, ylags(1 2) would include lags 1 and 2 of y. ylags(0) will cause no lagged value of y to be included in the model.

Dataset options

wide should be used if data are already in wide format. By default, data are assumed to be xtset long with both time and panel ID variables specified. The dataset is temporarily converted to wide format for use with sem. However, note that the file must have been created by a reshape wide command, using a file that is in long format and that was xtset, or else it will not have information that xtdpdml needs. Using this option is generally discouraged.

staywide keeps the data in wide format after running xtdpdml. This may be necessary if you want to use postestimation commands like predict. If you use staywide, be careful you do not accidentally save the wide .dta file and overwrite a file you want to keep!

tfix recodes time to equal $1, 2, \ldots, T$ and sets delta = 1. Time should be coded $t = 1, 2, \ldots, T$, where T = number of time points. By default, units such as years (for example, 1990, 1991, 1992) will cause errors or incorrect results. There will also be errors or incorrect results if delta does not equal 1; for example, $t = 1, 3, 5$. However, you can still have problems if delta was not specified correctly in the source dataset or if interval width is not consistent. It is safest if you correctly code time yourself, but tfix should work in most cases.

std standardizes all the variables in the model to have mean 0 and variance 1. It does this while the dataset is in long format. Hence, the standardization does not differ by time period; for example, at all time periods, you might subtract 10 from a variable and divide by 7. By standardizing this way, you keep the coefficients comparable across time. You probably will not want to use this option in most cases, but it can sometimes help when the model is having trouble converging.

std2(*varlist*) standardizes only the selected variables to have mean 0 and variance 1. It does not work if the wide option has been specified. Do not use time-series notation; just list the names of the variables you want standardized.

Model specification and constraints options

evars is an alternative and usually less efficient but sometimes helpful way of specifying the error terms. As noted earlier, Stata will not allow covariances between predetermined x's and the y residuals. Therefore, when there are predetermined variables,

xtdpdml drops most of the y residuals and replaces them with latent exogenous variables (E2, E3, etc.) that can be correlated with predetermined variables. When there are no predetermined variables, xtdpdml skips this step because it is no longer necessary. evars causes this step to be done anyway, which sometimes helps with convergence.

alphafree lets the coefficients of Alpha (fixed effects) differ across time. By default, they are all constrained to equal 1. Note that if this option is used, Alpha will be normalized by fixing its variance at 1; otherwise, the model sometimes has convergence problems.

xfree lets the coefficients of all the independent variables (except lagged y) freely differ across time.

xfree2(*varlist*) lets the coefficients of the specified independent variables freely differ across time.

yfree lets all lagged y coefficients freely differ across time.

yfree2(*numlist*) allows the specified lagged y coefficients to freely differ across time.

nocsd (alias is constinv) specifies that cross-sectional dependence is not allowed; that is, constants are constrained to be equal across time periods. This is equivalent to no effect of time. This option sometimes causes convergence problems.

errorinv constrains error variances to be equal across waves. The default is to let them freely differ. This option may cause convergence problems.

re specifies the random-effects model (where Alpha is uncorrelated with all observed Xs).

Reporting options

title(*string*) gives a title to the analysis. This title will appear in both the highlights results and (if requested) the Mplus code (described later). For example, title(Baseline Model).

details shows all the output generated by the sem command. Otherwise, only a highlights version is presented. This can be useful if you want to ensure the model specification is correct or if you want information not contained in the highlights. You can also replay all the results just by typing sem after running xtdpdml.

showcmd shows the sem command generated by xtdpdml. This can be useful to make sure the fitted model is what you wanted.

gof reports several goodness-of-fit measures after model estimation. It has the same effect as running the sem postestimation command estat gof, stats(all) after xtdpdml.

tsoff will turn off the use of time-series notation in the highlights printout and show the names of the variables actually used in the reshaped wide data. This may be useful

if you are going to hand-modify the code generated by xtdpdml. By default, when
possible, the highlights output produced by xtdpdml will use time-series notation
similar to what you see with commands like xtabond; for example, L3.xvar will
represent the lag 3 value of xvar. Because the data are reshaped wide, this is not
the same as the name of the variable that was actually used; for example, it might
be that L3.xvar corresponds to xvar2.

display_options: noci, nopvalues, noomitted, vsquish, noemptycells,
   cformat(%*fmt*), pformat(%*fmt*), sformat(%*fmt*), and nolstretch; see [R] **estima-
   tion options**.

coeflegend displays the names of the coefficients instead of the inferential statistics.
   This can be useful if, say, you are trying to use postestimation test commands to
   test hypotheses about effects.

decimals(*integer*) specifies the number of decimal places to display for the coeffi-
   cients, standard errors, and confidence limits. It is a shorthand way of specifying
   cformat(); for example, decimals(3) is the same as specifying cformat(%9.3f).
   You will get an error if you specify both decimals() and cformat(). The value
   specified must range between 0 and 8; 3 is often a good choice for making the output
   easier to read.

Other options

mplus(*filenamestub*[, *mplus_options*]) create .inp and data files that can be used
   by Mplus (the command has been tested only with Mplus 7.4). This command is
   adapted (with permission) from UCLA and Michael Mitchell's stata2mplus com-
   mand (Statistical Consulting Group 2015) but does not require that it be installed.
   The *filenamestub* must be specified; it will be used to name the Mplus .inp and
   .dat files. Everything else is optional. *mplus_options* may be replace, missing(#),
   analysis(*string*), and output(*string*). replace will cause existing .inp and .dat
   files to be overwritten. missing() specifies the missing value for all variables. The
   default is missing(-9999). analysis() and output() specify options to be passed
   to the Mplus analysis() and output() options. As is the case in Mplus, multi-
   ple analysis() and output() options should be separated by semicolons. xtdpdml
   cannot check your Mplus syntax, so be careful. As with Stata, the generated Mplus
   code will specify listwise deletion unless you have also used the fiml option. The
   mplus() option, of course, requires that you have Mplus and know how to use it.
   Because that will not be true of some Stata users, those interested in the option
   should consult the help file and examples provided on the support page for xtdpdml
   for additional details.

lavaan(*filenamestub*[, replace]) creates R commands and Stata .dta files that can
   be used by R's lavaan package. The *filenamestub* must be specified; it will be used
   to name the lavaan .R and .dta files. replace will cause existing .R and .dta
   files by those names to be overwritten. Of course, you must have R installed and
   know how to use it. You may want to edit the generated code if you want to change

or add options. So, for example, if the user-specified `lavaan(myfile, replace)`, `lav_myfile.R` and `lav_myfile.dta` would be created (replacing any existing files by those names). R is free and sometimes executes more quickly than Stata (but not as quickly as Mplus), which may make this a useful option for those who like to use R.

`semfile(`*filename*`[ , replace ])` specifies that the generated `sem` commands will be output to a file named *filename*`.do`. The `replace` option can be specified to replace an existing do-file by that name. This is useful if you want to try to modify the `sem` commands in ways that are not easily done with `xtdpdml`. You may wish to also specify the `staywide` option so that data remain correctly formatted for use with the generated do-file.

`store(`*stubname*`)` generates two sets of results: the full results, generated by `sem`, and a highlights-only set of results that can be used with programs like `esttab`. The stored results have the names *stubname*`_f` and *stubname*`_h`; for example, if you specify `store(model1)`, the results will be stored as `model1_f` and `model1_h`. The default *stubname* is `xtdpdml`, so after running `xtdpdml` without the `store()` option, you should have stored results `xtdpdml_f` and `xtdpdml_h`. You should not try to do most postestimation commands with the highlights version (for example, `predict`, `margins`) because necessary information may not be stored in the file; use the full version instead.

`dryrun` keeps `sem` from actually being executed. This will catch some errors immediately and can be useful if you want to see the `sem` command that is generated or wish to specify `staywide` to reformat the data from long to wide or just want to generate Mplus or `lavaan` code and data files.. This will often be combined with the `showcmd`, `mplus()`, `semfile()`, `lavaan()`, or `staywide` option.

`iterate(`#`)` specifies the maximum number of iterations allowed. The current default (subject to change) is `iterate(250)`. You can increase this number or change the maximization technique if the model is having trouble converging.

`technique(`*methods*`)` specifies the maximization techniques used. The default is `technique(nr 25 bhhh 25)` unless `method(adf)` is specified. You can change this if the model is having trouble converging. If you use `method(adf)` (asymptotic distribution free), the default is `technique(nr 25 bfgs 10)` because `adf` and the `bhhh` technique do not seem to work together.

maximize_options; see [R] **maximize** for details and information on options that can be used, for example, `difficult`.

`semopts(`*options*`)` specifies that other options allowed by `sem` will be included in the generated `sem` command.

fiml causes FIML to be used for missing data. This is the equivalent of specifying
method(mlmv) on the sem command. fiml sometimes dramatically slows down ex-
ecution, so be patient if you use it.[8]

v12 allows xtdpdml to run under Stata 12.1. It has not been extensively tested, so use
it at your own risk. The xtdpdml command was written and tested using Stata 13,
14, and 15.

skipcfatransform and skipconditional causes Stata to compute starting values the
same way as it did before Stata 14.2 when used together. Stata 14.2 changed the
way starting values are computed by sem. Usually the new procedures work better,
especially when fiml is used, but sometimes the old start values speed up execution,
are better for getting models to converge, or both. These options are ignored in
Stata 14.1 or earlier.

altstart is a convenient way to specify both skipcfatransform and skipconditional.

method(*method*) specifies estimation methods supported by sem, for example, ml, mlmv,
and adf. You likely will not use this option unless you want to specify method(adf).
Remember that method(ml) (maximum likelihood) is the default and that fiml is
a shorthand way of specifying method(mlmv) (ML with missing values, also known
as full-information maximum likelihood). If you use method(adf) (asymptotic dis-
tribution free), the default technique is set to technique(nr 25 bfgs 10) because
adf and the bhhh technique do not seem to work together.

vce(*vcetype*) specifies *vcetype*s supported by sem, for example, oim, robust. Not all
*vcetype*s have been tested with xtdpdml, so we recommend caution if you use this
option.

4 Examples

We have already provided one example that illustrates the key features of xtdpdml.
For many purposes, that one example may be enough. Here we illustrate additional
capabilities of xtdpdml that will often be useful. With many of the examples, we will
contrast the abilities of the ML or xtdpdml approach with those of the popular AB or
xtabond method. Specifically, our examples will illustrate xtdpdml's capabilities to
1) use FIML to better fit models with missing data; 2) use goodness-of-fit measures to
improve model specification; 3) compare and contrast fixed versus random effects, using
likelihood-ratio tests that avoid many of the problems that can occur with Hausman
tests; and 4) fit models with nonnormally distributed data. We will also show other

---

8. As the *Stata 15 Structural Equation Modeling Reference Manual* (2017, 574) explains when
method(mlmv) is specified, sem groups the data according to missing-value patterns. Each missing-
value pattern will have its own summary data. The log likelihood for a missing-value pattern
is computed using this summary data. The overall log likelihood is computed by summing the
log-likelihood values from each missing-value pattern. This process can be time consuming and
can have problems producing solutions when there are many missing-data patterns or when some
missing-data patterns have few cases. For an extended discussion of FIML, see Newman (2003),
Finkbeiner (1979), and Enders and Bandalos (2001).

important features of `xtdpdml`, such as its ability to estimate the effects of time-invariant variables in a fixed-effects model.

All the examples are adapted from Bollen and Brand (2010). They examine data from the National Longitudinal Survey of Youth. Respondents were 14 to 22 years old when first interviewed in 1979, and they were interviewed annually or biannually for several years thereafter. Bollen and Brand (2010) originally analyzed data from the years 1983–1993 at two-year intervals. The dependent variable (`lnwg`) is log hourly wages in current job. The main independent variable (`hchild`) is total number of children the respondent had at the time of the interview. Other variables in the model include whether or not the respondent was married (`mar`) or divorced (`div`); the respondent's level of educational attainment (`eduatt`); whether the respondent was currently in school (`cursc`); several measures of the respondent's part-time and full-time work experience (`snrpt`, `snrft`, `exppt`, and `expft`); and breaks in the respondent's employment history (`break`). See Bollen and Brand (2010) for more details on the variables and sample selection.

4.1 Missing data

We have previously noted that several simulations demonstrate the superiority of ML over AB in many situations. Of course, real data often offer complications that are not present in simulations. The Bollen–Brand dataset is strongly balanced, but many cases have missing data on one or more variables. Also, the model is a fixed-effects model but includes time-invariant variables.

Here we compare the results from `xtdpdml` and `xtabond`. First, we give the code and then excerpts from the output.

```
*** Section 4.1 -- Comparisons with AB, real data, using fiml and listwise
use https://www3.nd.edu/~rwilliam/statafiles/bollenbrand, clear
set matsize 7500
xtabond lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break  ///
        black hisp
estimates store gmm
* FIML
xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break, ///
        constinv errorinv fiml tfix store(fiml)                          ///
        inv(black hisp) title(Adapted from Bollen & Brand Social Forces 2010)
* Listwise
xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break, ///
        constinv errorinv tfix store(normal)                             ///
        inv(black hisp) gof
```

Both `xtabond` and `xtdpdml` require that the data first be `xtset`. But this dataset was previously saved after invoking the command `xtset id year`, so there is no need to repeat the command for any of the examples shown here.

For `xtdpdml`, the options `constinv` and `errorinv` were used to ensure comparability with `xtabond`, which presumes constant intercepts and constant variance by default. However, in most applications, these constraints would be both unnecessary and unde-

sirable. `tfix` is necessary because the `year` variable starts at 83 and increments by 2 for each period. The output includes the following:

```
. xtabond lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break black
> hisp
note: black dropped from div() because of collinearity
note: hisp dropped from div() because of collinearity
Arellano-Bond dynamic panel-data estimation     Number of obs     =     8,915
Group variable: id                               Number of groups  =     3,488
Time variable: year
                                                 Obs per group:
                                                            min =          1
                                                            avg =   2.555906
                                                            max =          4
Number of instruments =       21                 Wald chi2(11)     =   3315.32
                                                 Prob > chi2       =    0.0000
One-step results
```

lnwg	Coef.	Std. Err.	z	P>|z|	[95% Conf. Interval]	
lnwg						
L1.	-.0072789	.0402023	-0.18	0.856	-.086074	.0715163
hchild	-.0091342	.0090602	-1.01	0.313	-.0268919	.0086236
marr	.0468352	.0168387	2.78	0.005	.0138321	.0798384
div	.0747365	.0225606	3.31	0.001	.0305184	.1189545
eduatt	.0575892	.0102432	5.62	0.000	.0375128	.0776656
cursc	-.081103	.0153101	-5.30	0.000	-.1111103	-.0510956
snrpt	.0132922	.0054544	2.44	0.015	.0026018	.0239826
snrft	.0140817	.0027054	5.21	0.000	.0087792	.0193842
exppt	.056597	.0055597	10.18	0.000	.0457002	.0674937
expft	.0608082	.004636	13.12	0.000	.0517219	.0698946
break	.0200741	.0069791	2.88	0.004	.0063953	.0337528
black	0	(omitted)				
hisp	0	(omitted)				
_cons	.6280031	.1360759	4.62	0.000	.3612993	.894707

```
Instruments for differenced equation
        GMM-type: L(2/.).lnwg
        Standard: D.hchild D.marr D.div D.eduatt D.cursc D.snrpt D.snrft
                  D.exppt D.expft D.break
Instruments for level equation
        Standard: _cons
```

```
. xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break,
> constinv errorinv fiml tfix store(fiml) inv(black hisp)
> title(Adapted from Bollen & Brand Social Forces 2010)
Highlights: Adapted from Bollen & Brand Social Forces 2010
```

lnwg	Coef.	OIM Std. Err.	z	P>|z|	[95% Conf. Interval]	
lnwg						
lnwg						
L1.	.3378925	.0124879	27.06	0.000	.3134166	.3623684
hchild	−.0209521	.0063606	−3.29	0.001	−.0334186	−.0084856
marr	.0359839	.012841	2.80	0.005	.0108161	.0611517
div	.0617287	.017081	3.61	0.000	.0282505	.095207
eduatt	.0583252	.0072068	8.09	0.000	.0442001	.0724503
cursc	−.1075845	.0132218	−8.14	0.000	−.1334988	−.0816701
snrpt	.0088462	.0043731	2.02	0.043	.0002751	.0174173
snrft	.0174143	.0021041	8.28	0.000	.0132904	.0215383
exppt	.0308717	.0037348	8.27	0.000	.0235517	.0381917
expft	.0307015	.0022474	13.66	0.000	.0262966	.0351064
break	.0370938	.0043345	8.56	0.000	.0285983	.0455893
black	−.0074612	.0103375	−0.72	0.470	−.0277223	.0127999
hisp	.0730661	.012675	5.76	0.000	.0482236	.0979086

```
# of units = 5285. # of periods = 6. First dependent variable is from period 2.
Constants are invariant across time periods
LR test of model vs. saturated: chi2(218)  =      831.04, Prob > chi2 =  0.0000
IC Measures: BIC =  345115.17  AIC =  334921.02
Wald test of all coeff = 0: chi2(13) =     6701.90, Prob > chi2 =  0.0000
```

The results are strikingly different. Almost 21,000 records have data on at least one variable in the model, and all of these observations are used by xtdpdml (with the fiml option). However, only 8,915 records are used by xtabond because it deletes any record with missing data. Perhaps for this reason, xtabond produces a highly implausible estimate of almost zero effect of lagged wages on current wages and also says that the effect of the main independent variable (number of children) is statistically insignificant. In the xtdpdml results, both effects are highly significant, and the signs of the effects are in the expected direction. Many other variables have larger z statistics in xtdpdml than they do in xtabond. xtabond cannot estimate effects for the time-invariant variables black and hisp. xtdpdml can and shows that the effect of hisp is highly significant.

Even if we leave out the `fiml` option, thereby deleting all persons who have missing data at any time point, the results from `xtdpdml` seem somewhat more plausible.

```
. * Listwise deletion used instead of fiml
. xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break,
> constinv errorinv tfix store(normal) inv(black hisp) gof
Highlights: Dynamic Panel Data Model using ML for outcome variable lnwg
```

lnwg	Coef.	OIM Std. Err.	z	P>|z|	[95% Conf. Interval]	
lnwg						
lnwg						
L1.	.2776779	.0171863	16.16	0.000	.2439933	.3113624
hchild	−.0144888	.0099824	−1.45	0.147	−.034054	.0050765
marr	.0590625	.0186809	3.16	0.002	.0224486	.0956763
div	.0577631	.0253645	2.28	0.023	.0080495	.1074767
eduatt	.0763835	.0108579	7.03	0.000	.0551024	.0976647
cursc	−.0923283	.0191093	−4.83	0.000	−.1297819	−.0548747
snrpt	.0150824	.0059555	2.53	0.011	.0034099	.0267549
snrft	.0101602	.0027619	3.68	0.000	.004747	.0155734
exppt	.04097	.0054855	7.47	0.000	.0302186	.0517215
expft	.0379746	.0030576	12.42	0.000	.0319817	.0439675
break	.0195759	.0075289	2.60	0.009	.0048195	.0343322
black	−.0299847	.017994	−1.67	0.096	−.0652523	.0052829
hisp	.0737694	.0220227	3.35	0.001	.0306056	.1169331

```
# of units = 1229. # of periods = 6. First dependent variable is from period 2.
Constants are invariant across time periods
LR test of model vs. saturated: chi2(218) =      612.60, Prob > chi2 =  0.0000
IC Measures: BIC =  113801.75  AIC =  105870.00
Wald test of all coeff = 0: chi2(13) =     3171.73, Prob > chi2 =  0.0000
```
   (*output omitted*)

As we would expect, with listwise deletion the smaller sample size causes effects to be less statistically significant, but the effect of lagged wages continues to be positive and highly significant.

In fairness, `xtdpdml` took far longer to run than `xtabond`, especially when FIML was used. Further, there will be other situations where the two methods will yield more similar results, and when panels are far from being strongly balanced, `xtabond` may work better (or `xtdpdml` may not work at all). But at least in this particular case, where many cases have missing data and time-invariant variables are in the model, `xtdpdml` seems to be the better alternative.

4.2 Panel model with fixed effects; goodness-of-fit measures

Bollen and Brand (2010) present a series of panel models with random and fixed effects. They used Mplus for the analysis; but now, many of their models can be more easily fit with `xtdpdml` (hand-tweaking the `sem` code may be required in a few cases). In this relatively simple example, there are no lagged independent variables. With a strongly balanced panel, no missing data, and no effect of lagged y, `xtdpdml` produces results

that are almost identical to `xtreg`. Of course, there are missing data with this dataset, making the use of `xtdpdml` with FIML desirable.

Here we present the fixed-effects model 2 from their table 3. We also include the `gof` option, which includes several goodness-of-fit measures in the output.

```
. use https://www3.nd.edu/~rwilliam/statafiles/bollenbrand, clear
(Bollen & Brand 2010 Social Forces V 89(1) NLSY 1983-1993 Odd years Long format)
. xtdpdml lnwg hchild marr div, ylags(0) fiml tfix errorinv gof store(baseline)
Highlights: Dynamic Panel Data Model using ML for outcome variable lnwg
```

lnwg	Coef.	OIM Std. Err.	z	P>|z|	[95% Conf. Interval]
lnwg						
hchild	-.0704949	.0055935	-12.60	0.000	-.081458	-.0595319
marr	.0826099	.0104827	7.88	0.000	.0620642	.1031556
div	.0572981	.014612	3.92	0.000	.0286591	.0859372

```
# of units = 5231. # of periods = 6. First dependent variable is from period 1.
Constants are free to vary across time periods
LR test of model vs. saturated: chi2(106)  =    1940.93, Prob > chi2 =  0.0000
IC Measures: BIC =    73683.21  AIC =   72252.61
Wald test of all coeff = 0: chi2(3) =      204.34, Prob > chi2 =  0.0000
```

Fit statistic	Value	Description
Likelihood ratio		
chi2_ms(106)	1940.932	model vs. saturated
p > chi2	0.000	
chi2_bs(123)	8307.362	baseline vs. saturated
p > chi2	0.000	
Population error		
RMSEA	0.058	Root mean squared error of approximation
90% CI, lower bound	0.055	
upper bound	0.060	
pclose	0.000	Probability RMSEA <= 0.05
Information criteria		
AIC	72252.615	Akaike´s information criterion
BIC	73683.209	Bayesian information criterion
Baseline comparison		
CFI	0.776	Comparative fit index
TLI	0.740	Tucker-Lewis index
Size of residuals		
CD	0.870	Coefficient of determination

```
Note: SRMR is not reported because of missing values.
```

See Acock (2013) for a discussion of goodness-of-fit measures in SEM. The `model vs. saturated` chi-squared is a test of all 106 overidentifying restrictions implied by the model. Here the chi-squared (1940.93) is almost 20 times its degrees of freedom, suggesting a poor fit to the data. However, as is well known, it is difficult to find any reasonably parsimonious model that will pass this test with a sample size of more

than 5,000. Other goodness-of-fit tests reported here are less sensitive to sample size. An RMSEA of less than 0.05 is considered to be a good fit, and we are almost there at 0.058. On the other hand, both the CFI and TLI are well below 0.90, which is the usual standard for a minimally acceptable model.

To improve the model fit, we could consider relaxing some of the constraints of the model; for example, we could let the effects of some variables vary across time by using the xfree or yfree option. Modification indices (obtained with the sem postestimation command estat mindices) could provide additional guidance on how to modify the model. Because there are so many equality constraints imposed by the model, the estat scoretests command may be especially useful because it displays score tests (Lagrangian multiplier tests) for each of the linear constraints that are imposed on the model. In this case,

```
. estat scoretests
Score tests for linear constraints
 ( 1)  [lnwg1]hchild1 - [lnwg6]hchild6 = 0
 ( 2)  [lnwg1]marr1 - [lnwg6]marr6 = 0
 ( 3)  [lnwg1]div1 - [lnwg6]div6 = 0
 ( 4)  [lnwg1]Alpha = 1
 ( 5)  [lnwg2]hchild2 - [lnwg6]hchild6 = 0
 ( 6)  [lnwg2]marr2 - [lnwg6]marr6 = 0
 ( 7)  [lnwg2]div2 - [lnwg6]div6 = 0
 ( 8)  [lnwg2]Alpha = 1
 (12)  [lnwg3]Alpha = 1
 (13)  [lnwg4]hchild4 - [lnwg6]hchild6 = 0
 (15)  [lnwg4]div4 - [lnwg6]div6 = 0
 (16)  [lnwg4]Alpha = 1
 (17)  [lnwg5]hchild5 - [lnwg6]hchild6 = 0
 (20)  [lnwg5]Alpha = 1
 (21)  [lnwg6]Alpha = 1
 (22)  [var(e.lnwg1)]_cons - [var(e.lnwg6)]_cons = 0
 (23)  [var(e.lnwg2)]_cons - [var(e.lnwg6)]_cons = 0
 (24)  [var(e.lnwg3)]_cons - [var(e.lnwg6)]_cons = 0
```

	chi2	df	P>chi2
(1)	54.652	1	0.00
(2)	17.970	1	0.00
(3)	4.194	1	0.04
(4)	543.286	1	0.00
(5)	15.011	1	0.00
(6)	5.726	1	0.02
(7)	8.885	1	0.00
(8)	91.101	1	0.00
(12)	5.594	1	0.02
(13)	5.866	1	0.02
(15)	4.213	1	0.04
(16)	98.223	1	0.00
(17)	4.062	1	0.04
(20)	100.611	1	0.00
(21)	134.406	1	0.00
(22)	12.887	1	0.00
(23)	20.007	1	0.00
(24)	20.581	1	0.00

The results strongly suggest that the `Alpha` coefficients (that is, the coefficients for the unmeasured fixed effects–see tests 4, 8, 12, 16, 20, and 21) are not the same across time (by default, all are set equal to 1). The constraint that the error variances are the same across time (which was imposed by the `errorinv` option but is inherent in the AB approach) also seems dubious (see tests 22, 23, and 24). Luckily, unlike AB, both problems are easily addressed with ML–SEM. As Allison, Williams, and Moral-Benito (2017) note, "It's even possible to allow the coefficient of α, the fixed effect, to vary with time instead of being constrained to 1 for every time point. This option is attractive because it removes one of the principal limitations of the classic fixed-effects estimator: that it does not control for unmeasured time-invariant variables when their effects change over time."

We therefore relax those constraints by dropping the `errorinv` option and adding `alphafree`, resulting in

```
. xtdpdml lnwg hchild marr div, ylags(0) fiml tfix alphafree gof store(modified)
   (output omitted)
```

Fit statistic	Value	Description
Likelihood ratio		
chi2_ms(96)	789.252	model vs. saturated
p > chi2	0.000	
chi2_bs(123)	8307.362	baseline vs. saturated
p > chi2	0.000	
Population error		
RMSEA	0.037	Root mean squared error of approximation
90% CI, lower bound	0.035	
upper bound	0.040	
pclose	1.000	Probability RMSEA <= 0.05
Information criteria		
AIC	71120.934	Akaike´s information criterion
BIC	72617.152	Bayesian information criterion
Baseline comparison		
CFI	0.915	Comparative fit index
TLI	0.891	Tucker-Lewis index
Size of residuals		
CD	0.899	Coefficient of determination

Note: SRMR is not reported because of missing values.

```
. lrtest baseline_f modified_f, stats
```

Likelihood-ratio test	LR chi2(10) =	1151.68
(Assumption: baseline_f nested in modified_f)	Prob > chi2 =	0.0000

Akaike´s information criterion and Bayesian information criterion

Model	Obs	ll(null)	ll(model)	df	AIC	BIC
baseline_f	5,231	.	-35908.31	218	72252.61	73683.21
modified_f	5,231	.	-35332.47	228	71120.93	72617.15

Note: N=Obs used in calculating BIC; see [R] BIC note.

Relaxing the constraints on `Alpha` and the error variances requires only 10 degrees of freedom (because equality constraints are relaxed on 5 `Alpha` coefficients and 5 error variances) and produces a dramatic improvement in model fit. The BIC and AIC statistics also strongly favor the less constrained model. There are also clear improvements in other goodness-of-fit measures. `RMSEA` = 0.037, which is much better than the 0.05 value that is considered a good fit. Similarly, `CFI` = 0.915 and `TLI` = 0.891, close to or better than the usual standard of 0.90 for a minimally acceptable model.

Of course, the researcher may also want to reconsider whether other model assumptions (such as not including any lagged independent variables, especially lagged y) are justified.

4.3 Fixed-effects versus random-effects models

An alternative to the Hausman test is the likelihood-ratio test. Allison (2009) notes that a Hausman test is often used to contrast fixed-effects and random-effects models. However, he notes that the Hausman test can sometimes be problematic; for example, it can produce negative values for some data configurations. He argues that a likelihood-ratio test can have superior statistical properties. To illustrate this, we again fit a fixed-effects model with the Bollen and Brand (2010) data. Fitting a random-effects model instead requires only that we add the `re` option to `xtdpdml`. By default, the latent variable representing fixed effects, `Alpha`, is allowed to correlate with all the time-varying exogenous variables, including y_0. In the random-effects model, these correlations are constrained to zero. After fitting both models, we can use the `lrtest` command to contrast the results. The code is

```
*** 4.3 Fixed-effects versus random-effects models; alternative to the Hausman test
use https://www3.nd.edu/~rwilliam/statafiles/bollenbrand, clear
set matsize 7500

* Random effects
xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break, ///
        constinv errorinv fiml tfix re store(re)                         ///
        inv(black hisp) title(Adapted from Bollen & Brand Social Forces 2010)

* Fixed effects
xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break, ///
        constinv errorinv fiml tfix store(fe)                            ///
        inv(black hisp) title(Adapted from Bollen & Brand Social Forces 2010)

esttab re_h fe_h, mtitles(Random Fixed) scalar(chi2_ms df_ms p_ms BIC AIC) z
lrtest re_f fe_f, stats
```

As noted before in section 3, when the `store()` option is used, two versions of the results are stored. In this case, the highlights-only results are stored in `re_h` and `fe_h`. These can be used with the community-contributed command `esttab` (Jann 2007) to display key results in tables. The full results are stored in `re_f` and `fe_f`. These should be used with `lrtest` to test whether the differences between the two models are significant. We show just the postestimation output here:

```
. esttab re_h fe_h, mtitles(Random Fixed) scalar(chi2_ms df_ms p_ms BIC AIC ) z
```

	(1) Random	(2) Fixed
lnwg		
L.lnwg	0.393***	0.338***
	(31.49)	(27.06)
hchild	-0.0116***	-0.0210***
	(-3.45)	(-3.29)
marr	0.00774	0.0360**
	(0.95)	(2.80)
div	0.0323**	0.0617***
	(3.08)	(3.61)
eduatt	0.0547***	0.0583***
	(28.13)	(8.09)
cursc	-0.0952***	-0.108***
	(-8.27)	(-8.14)
snrpt	0.00721*	0.00885*
	(2.18)	(2.02)
snrft	0.0160***	0.0174***
	(10.24)	(8.28)
exppt	0.0195***	0.0309***
	(10.34)	(8.27)
expft	0.0300***	0.0307***
	(19.54)	(13.66)
break	0.00380*	0.0371***
	(2.11)	(8.56)
black	-0.0227**	-0.00746
	(-2.62)	(-0.72)
hisp	0.0570***	0.0731***
	(5.68)	(5.76)
N	5285	5285
chi2_ms	1375.9	831.0
df_ms	269	218
p_ms	7.79e-148	2.28e-72
BIC	345222.9	345115.2
AIC	335363.9	334921.0

```
z statistics in parentheses
* p<0.05, ** p<0.01, *** p<0.001

. lrtest re_f fe_f, stats
```

Likelihood-ratio test	LR chi2(51) =	544.90
(Assumption: re_f nested in fe_f)	Prob > chi2 =	0.0000

Akaike´s information criterion and Bayesian information criterion

Model	Obs	ll(null)	ll(model)	df	AIC	BIC
re_f	5,285	.	-166182	1500	335363.9	345222.9
fe_f	5,285	.	-165909.5	1551	334921	345115.2

Note: N=Obs used in calculating BIC; see [R] BIC note.

The likelihood-ratio test and the BIC and AIC statistics all favor the fixed-effects model.[9] This suggests that at least some important variables that are correlated with the time-varying predictors have been omitted from the model. Note that, with more conventional fixed-effects methods, the effects of time-invariant variables (of which there are several in this case) cannot be estimated. This is not the case with the `xtdpdml` approach.

4.4   Nonnormality

By default, the `xtdpdml` model assumes that observed endogenous variables have a multivariate normal distribution, conditional on the exogenous variables. If FIML is used, the multivariate normal assumption also applies to any exogenous variables with missing data. When the normality assumption holds, ML is asymptotically efficient, implying that the true standard errors are as small as possible.

What if the normality assumption is violated? Even under nonnormality, ML is consistent and asymptotically normal (Moral-Benito 2013). But it will not be efficient, and the reported standard errors will not be consistent estimates of the true standard errors, leading to incorrect p-values and confidence intervals. The degree of bias depends on the circumstances. Simulations by Moral-Benito, Allison, and Williams (2018) and by Allison, Williams, and Moral-Benito (2017) show that under nonnormal data-generating processes, the ML estimator performs quite well in finite samples, both in comparison with its performance under normal data terms and with GMM for nonnormal data.

When the normality assumption is possibly problematic, the `sem` command (and hence `xtdpdml`) provides various ways of adjusting standard errors, test statistics, and the parameter estimates themselves. This section will explain three of these approaches and some of the advantages and disadvantages of each.

First, with Stata 14 and later, the `vce(sbentler)` option can be specified. As StataCorp (2018) explains on its webpages,

> Stata's linear `sem` provides the Satorra–Bentler scaled chi-squared test for model goodness of fit versus the saturated model. Why do you care? The likelihood-ratio test comparing your fitted model with the saturated model is derived under the assumption that the observed variables in your model are normally distributed. If they are not, that test is not appropriate. The Satorra–Bentler scaled chi-squared test is robust to nonnormality. Because many other goodness-of-fit statistics are derived from the model using the chi-squared test, they too become robust to nonnormality. [What's more] The same adjustment that gives you the Satorra–Bentler scaled chi-squared

---

9. The random-effects model imposes constraints on 51 correlations between `Alpha` and the exogenous variables that are free to vary in the fixed-effects model. There are 10 time-varying variables for each of 5 time periods, so their 50 correlations with `Alpha` are set to 0. The correlation between `Alpha` and `lnwg` at time 1 is also constrained to equal 0.

test makes a host of other things robust to nonnormality: standard errors, *p*-values, and confidence intervals reported by `sem` and standard errors, *p*-values, and confidence intervals for most post hoc comparisons and tests, including joint tests, nonlinear tests, linear and nonlinear expressions of parameters, estimated marginal means and marginal effects, equation-level Wald tests, direct and indirect effects, and tests of standardized parameters.

Note that `vce(sbentler)` relaxes the normality assumption when estimating standard errors but does not affect the coefficient estimates. That is, regardless of whether you specify `vce(sbentler)`, the coefficient estimates will be the same.

Unfortunately, a key limitation of the `vce(sbentler)` option is that it does not work with FIML; that is, it requires listwise deletion. If missing data are a concern, researchers may prefer to use a different option, `vce(robust)`. As StataCorp (2018) also points out on the same page,

Stata's `sem` already had an adjustment that makes everything in "What's more" true. It is often called the Huber or White method, or just called the linearized estimator. Whatever you call it, this estimator and the Satorra–Bentler adjustment are making your inferences robust to similar things. They are derived and computed differently, so they produce different estimates. As samples become very large, however, they converge to the same estimates.

When `vce(robust)` is specified, along with the default ML estimation method, Stata calls the estimation method quasi–maximum likelihood (QML). Like `vce(sbentler)`, QML relaxes the normality assumption when estimating standard errors but does not affect the coefficient estimates; that is, regardless of whether you specify `vce(robust)`, the coefficient estimates will be the same.

A key advantage of `vce(robust)` is that, unlike `vce(sbentler)`, it can be used with FIML; that is, it does not require listwise deletion of missing data. Because the standard errors from `vce(robust)` and `vce(sbentler)` are asymptotically equivalent, `vce(robust)` may be preferred when missing data are a concern. However, unlike `vce(sbentler)`, `vce(robust)` does not provide many goodness-of-fit measures, for example, no overall chi-squared, no `RMSEA`, and no `TLI` or `CFI`.

With both `vce(robust)` and `vce(sbentler)`, the standard errors change but the coefficient estimates remain the same. A third approach is the asymptotic distribution free (ADF) estimation method (also known as weighted least squares in some literature), which is achieved by specifying the option `method(adf)`. As the *Stata 15 Structural Equation Modeling Reference Manual* (2017, 47–48) explains,

ADF makes no assumption of joint normality or even symmetry, whether for observed or latent variables.... ADF produces justifiable point estimates and standard errors under nonnormality.... Be aware, however, that ADF is less

efficient than ML when latent variables can be assumed to be normally distributed. If latent variables (including errors) are not normally distributed, on the other hand, ADF will produce more efficient estimates than ML or QML.

Like vce(sbentler), ADF requires listwise deletion of missing data, which could be a major disadvantage in some cases. Also, ADF typically requires large samples to work effectively. Our own limited tests suggest that models using ADF are harder to fit and more likely to have convergence problems.

For the sake of brevity, we will present only the Stata code (with comments) for each of these approaches and not the output. As just explained, coefficient estimates are the same as for the models presented in section 4.1 (either using listwise deletion or FIML), but the standard errors, fit statistics, or both are different. The one exception is with method ADF, which should produce different coefficient estimates, but the models could not converge to a solution in this case.[10]

```
* Section 4.4 -- Nonnormality
use https://www3.nd.edu/~rwilliam/statafiles/bollenbrand, clear
set matsize 7500
* Use vce(sbentler). Coefficients are the same as when listwise is used.
* Standard errors and goodness-of-fit measures change.
xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break, ///
       constinv errorinv tfix store(sbentler) vce(sbentler)          ///
       inv(black hisp) gof

* vce(sbentler) does not work with fiml.
capture noisily xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft  ///
       exppt expft break, constinv errorinv tfix store(sbentler)       ///
       vce(sbentler) inv(black hisp) gof fiml
* Produces the error message "vce(sbentler) not allowed with method(mlmv)".

* Now use vce(robust).
* Coefficients stay the same as when listwise is used but standard errors
* change. In these particular examples, vce(sbentler) and vce(robust)
* produce very similar estimates of the standard errors,
* but few goodness-of-fit measures are reported with vce(robust).
xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break, ///
       constinv errorinv tfix store(robust)                           ///
       inv(black hisp) vce(robust) gof
* vce(robust) does work with fiml.
* Coefficients are the same as in example 4.1 using FIML, but standard    ///
       errors differ.
xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break, ///
       constinv errorinv tfix store(robustfiml)                       ///
       inv(black hisp) vce(robust) gof fiml

* Now use method(adf). Both coefficients and standard errors change
* but will not converge for this example.
xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft exppt expft break, ///
       constinv errorinv tfix store(adf)                              ///
       inv(black hisp) method(adf) gof
```

10. Mplus was also unable to converge to a solution using ADF (which Mplus calls weighted least squares), which suggests that this is not a problem specific to Stata.

```
    * Also, fiml will not work with adf.
capture noisily xtdpdml lnwg hchild marr div eduatt cursc snrpt snrft    ///
        exppt expft break,                                               ///
        constinv errorinv tfix store(adf2)                               ///
        inv(black hisp) method(adf) gof fiml
    * Produces the error message "You cannot specify both fiml and method(adf)".
```

5   Special topics

The `xtdpdml` approach requires that some problems be approached differently than they are with other models. In addition, for some applications, `xtdpdml` models can be harder to fit and may consume far more computing time than other approaches. In this section, we discuss various ways to deal with these issues.

5.1   Interactions with time

Researchers sometimes want constants and coefficients to differ across time. `xtdpdml` can do this, but because data are reshaped wide, the procedure is different than it is with other programs.

By default, `xtdpdml` allows the constants (intercepts) to differ across time periods. With other `xt` commands, this would be like including `i.time` in the model. The `constinv` or `nocsd` option can be specified if the user wants the constants to be invariant across time. Note that these options will sometimes cause convergence problems.

In other situations, the user might want interactions with time where the coefficient of a variable is free to differ across time periods. With other commands, this might be accomplished by specifying something like `i.time#c.ses`. With `xtdpdml`, you use the free options instead; for example, `xfree2(ses)` will allow the effect of `ses` to differ at each time period. Similarly, `alphafree` might be used to allow the fixed effects to differ across time periods, which is generally not possible with other methods.

5.2   Speed, convergence, and missing data problems

`xtdpdml` sometimes has trouble converging to a solution or else is slow in doing so. This might occur, for example, when time-varying variables do not vary much across time, creating problems of collinearity in estimation. Here are some things you can try when that happens.

Stata 14.2 introduced major enhancements to the `sem` command that dramatically helped with both convergence and speed, especially when FIML is used. Try to use 14.2 or later when using `xtdpdml`.

By default, `sem` deletes cases on a listwise basis. Because data are converted to wide format, a missing time period or even missing data on a single variable at a single time can cause all the data for an individual to be lost. In addition, `xtdpdml` models are computationally intensive. `xtdpdml` therefore works best when panels are strongly

balanced, T is small (for example, less than 10), and there are no missing data. If these conditions do not apply to your data, consider doing the following:

- The `fiml` option will often help when some data are missing or when entire time periods are missing for some individuals. Nonetheless, while `fiml` worked well in the examples presented here, we have found it can have problems with unbalanced panels, especially when some time periods have only a few cases.

- Consider restricting your data to a smaller range of time periods where most or all cases have complete data. Or consider using only every kth year, for example, 1980, 1985, 1990, ..., 2015. Using fewer variables in the model may also help.

- Consider rescaling variables; for example, measure income in thousands of dollars rather than in dollars. This can help with numerical precision problems. The `std` option makes rescaling and standardizing variables easy, although it may make coefficients a little harder to interpret. If `std` solves a convergence problem, then you may want to rescale the variables yourself in a more interpretable way; for example, if income is measured in dollars, then compute income/1,000 to measure income in thousands of dollars.

- Stata 14.2 changed the way start values are computed. Our experience is that models using `fiml` tend to run far more quickly in 14.2 compared with earlier versions. However, sometimes the new start values actually make the models run more slowly or cause convergence problems. If you are running Stata 14.2 or later, you can add the options `skipcfatransform`, `skipconditional`, or both to make Stata use the old starting values method. `altstart` is an easy way to specify both options.

- Mplus sometimes succeeds when Stata has problems and is often much faster. Try the `mplus()` option if you have access to that software.

There are several other options you can try if you are having problems achieving convergence. Much of this advice applies to many commands, not just `xtdpdml`.

- The `difficult` option will sometimes work miracles. There is no guarantee it will work (sometimes it makes things worse), but it is easy to try.

- The `technique()` option can be specified to use different maximization techniques. See the help for `maximize`.

- `evars` sometimes helps with convergence when there are no predetermined variables in the model. It is an alternative and usually less efficient way of specifying the error terms. But sometimes it helps and may be necessary for replicating results from earlier versions of `xtdpdml`.

- The `iterate()` option can be used to increase or decrease the number of iterations that `xtdpdml` performs before giving up. The `details` option will show the

iteration log. You can increase or decrease the number of iterations depending on whether it appears the program is converging to a solution.

Finally, remember that problems with regressing y on lagged y may not be that severe when N or T is large or the autoregressive coefficient is small (Arellano 2003). Commands like `xtreg` or `xtabond` may meet your needs in such situations. But even then, as our examples showed, features like FIML and time-invariant independent variables may make it worthwhile to pare your dataset down so you can do at least some analyses with `xtdpdml`.

6   Other alternatives to xtdpdml

The community-contributed commands `xtmoralb` (Moral-Benito 2013) and `xtdpdqml` (Kripfganz 2016) both do ML estimation for dynamic panel models. They can do some of the same things as `xtdpdml` and may be useful in some situations. However, they also have some important limitations. `xtmoralb` works well with predetermined variables (indeed, we used it to refine `xtdpdml`). However, it cannot handle time-invariant variables and lagged exogenous variables and is not fully efficient with strictly exogenous variables.

`xtdpdqml` works with strictly exogenous variables and can also sometimes produce results similar to `xtdpdml`. However, it cannot handle time-invariant variables (in a fixed-effects model) and (according to the author) is inappropriate for predetermined variables. Also, `xtdpdqml` implements the ML method of Hsiao, Pesaran, and Tahmiscioglu (2002), which makes strong and questionable assumptions about initial conditions.

7   Support

Additional information on the `xtdpdml` command, as well as suggestions for dealing with possible problems, can be found on its support page at https://www3.nd.edu/~rwilliam/dynamic/index.html.

8   Acknowledgments

reviewer provided numerous valuable suggestions for the manuscript and for the program itself. Particularly helpful were comments that helped us to make our examples more effective and that made our arguments more relevant and understandable to a broader audience.

9 References

Acock, A. C. 2013. *Discovering Structural Equation Modeling Using Stata*. Revised ed. College Station, TX: Stata Press.

Ahn, S. C., and P. Schmidt. 1995. Efficient estimation of models for dynamic panel data. *Journal of Econometrics* 68: 5–27.

Allison, P. D. 2000. "Inferring causal order from panel data". Paper presented at the Ninth International Conference on Panel Data, June 22, Geneva, Switzerland.

———. 2005a. "Causal inference with panel data". Paper presented at the American Sociological Annual Meeting, Aug. 13–16, Philadelphia, PA.

———. 2005b. *Fixed Effects Regression Methods for Longitudinal Data Using SAS*. Cary, NC: SAS Institute.

———. 2009. *Fixed Effects Regression Models*. Thousand Oaks, CA: Sage.

Allison, P. D., R. Williams, and E. Moral-Benito. 2017. Maximum likelihood for cross-lagged panel models with fixed effects. *Socius: Sociological Research for a Dynamic World* 3: 1–17.

Anderson, T. W., N. Kunitomo, and T. Sawa. 1982. Evaluation of the distribution function of the limited information maximum likelihood estimator. *Econometrica* 50: 1009–1027.

Arellano, M. 2003. *Panel Data Econometrics*. Oxford: Oxford University Press.

Arellano, M., and S. Bond. 1991. Some tests of specification for panel data: Monte Carlo evidence and an application to employment equations. *Review of Economic Studies* 58: 277–297.

Bai, J. 2013. Fixed-effects dynamic panel models, a factor analytical method. *Econometrica* 81: 285–314.

Baltagi, B. H. 2013. *Econometric Analysis of Panel Data*. 5th ed. New York: Wiley.

Bentler, P. M., and D. G. Weeks. 1980. Linear structural equations with latent variables. *Psychometrika* 45: 289–308.

Bollen, K. A., and J. E. Brand. 2010. A general panel model with random and fixed effects: A structural equations approach. *Social Forces* 89: 1–34.

Cornwell, C., and P. Rupert. 1988. Efficient estimation with panel data: An empirical comparison of instrumental variables estimators. *Journal of Applied Econometrics* 3: 149–155.

Enders, C. K., and D. L. Bandalos. 2001. The relative performance of full information maximum likelihood estimation for missing data in structural equation models. *Structural Equation Modeling: A Multidisciplinary Journal* 8: 430–457.

Finkbeiner, C. 1979. Estimation for the multiple factor model when data are missing. *Psychometrika* 44: 409–420.

Hsiao, C. 2014. *Analysis of Panel Data*. 3rd ed. Cambridge: Cambridge University Press.

Hsiao, C., M. H. Pesaran, and A. K. Tahmiscioglu. 2002. Maximum likelihood estimation of fixed effects dynamic panel data models covering short time periods. *Journal of Econometrics* 109: 107–150.

Jann, B. 2007. Making regression tables simplified. *Stata Journal* 7: 227–244.

Jöreskog, K. G. 1978. Structural analysis of covariance and correlation matrices. *Psychometrika* 43: 443–477.

Kripfganz, S. 2016. Quasi–maximum likelihood estimation of linear dynamic short-T panel-data models. *Stata Journal* 16: 1013–1038.

Moral-Benito, E. 2013. Likelihood-based estimation of dynamic panels with predetermined regressors. *Journal of Business and Economic Statistics* 31: 451–472.

Moral-Benito, E., P. Allison, and R. Williams. 2018. Dynamic panel data modeling using maximum likelihood: An alternative to Arellano–Bond. https://www3.nd.edu/~rwilliam/dynamic/Benito_Allison_Williams.pdf.

Newman, D. A. 2003. Longitudinal modeling with randomly and systematically missing data: A simulation of ad hoc, maximum likelihood, and multiple imputation techniques. *Organizational Research Methods* 6: 328–362.

Over, M. 1999. linewrap: Split a long string into shorter strings. http://digital.cgdev.org/doc/stata/MO/Misc.

Roodman, D. 2009. How to do xtabond2: An introduction to difference and system GMM in Stata. *Stata Journal* 9: 86–136.

StataCorp. 2017. *Stata 15 Structural Equation Modeling Reference Manual*. College Station, TX: Stata Press.

StataCorp. 2018. Satorra–Bentler adjustments. https://www.stata.com/features/overview/sem-satorra-bentler/.

Statistical Consulting Group. 2015. stata2mplus: Convert Stata files to Mplus files. UCLA: Statistical Consulting Group. https://stats.idre.ucla.edu/stat/stata/ado/analysis.

Wooldridge, J. M. 2010. *Econometric Analysis of Cross Section and Panel Data.* 2nd ed. Cambridge, MA: MIT Press.

About the authors

Richard Williams is associate professor and a former chairman of the Department of Sociology at the University of Notre Dame. His work has appeared in the *American Sociological Review*, *Social Forces*, *Social Problems*, *Demography*, *Cityscape*, and *Sociological Methods and Research*. Recent work by Williams has looked at issues involving the analysis of categorical data. He notes that assumptions of heterogeneity and proportionality are often violated in commonly used logit and ordered logit models. He discusses how heterogeneous choice models and generalized ordered logit models provide potential solutions to these problems. He is the author of the Stata programs `gologit2` and `oglm`, which make the estimation of these models possible. In his work, he also shows how the use of adjusted predictions and marginal effects can be major aids in interpreting the results from categorical models. He won the 2015 *Stata Journal* Editors' Prize for his work in these areas.

Paul D. Allison is professor of sociology at the University of Pennsylvania and president of Statistical Horizons LLC. After completing his doctorate in sociology at the University of Wisconsin, he did postdoctoral study in statistics at the University of Chicago and the University of Pennsylvania. He has published 8 books and more than 70 articles on topics that include linear regression, logistic regression, structural equation models, inequality measures, missing data, survival analysis, and scientific careers. At present, his principal methodological research is focused on the analysis of longitudinal data, especially with determining the causes and consequences of events, and on methods for handling missing data. A former Guggenheim Fellow, Allison received the 2001 Lazarsfeld Award for distinguished contributions to sociological methodology. In 2010, he was named a Fellow of the American Statistical Association.

Enrique Moral-Benito is an economist at Banco de España (DG Economics, Statistics and Research) and adjunct professor of quantitative methods at CEMFI. He received a PhD in economics from UIMP–CEMFI in 2010. His main research interests are aggregate productivity, economic growth, and panel-data econometrics. His work has appeared in journals such as the *Review of Economics and Statistics*, *Journal of Applied Econometrics*, *Journal of Business and Economic Statistics*, and *European Economic Review*.