The Stata Journal (2018)
**18**, Number 4, pp. 844–862

# xtpdyn: A community-contributed command for fitting dynamic random-effects probit models with unobserved heterogeneity

Raffaele Grotti
Department of Political and Social Sciences
European University Institute
San Domenico di Fiesole, Italy
raffaele.grotti@eui.eu

Giorgio Cutuli
Department of Sociology and Social Research
University of Trento
Trento, Italy
g.cutuli@unitn.it

**Abstract.** Dynamic random-effects probit models are increasingly applied in many disciplines to study dynamics of persistence in dichotomous outcomes. Despite the increasing popularity of these models, an estimation command for them does not exist yet. In this article, we present the `xtpdyn` command, which implements the model as proposed by Rabe-Hesketh and Skrondal (2013, *Economics Letters* 120: 346–349). We also present `probat`, a postestimation command that provides estimates of transition rates and a set of associated statistics.

**Keywords:** st0543, xtpdyn, probat, dynamic panel models, dynamic random-effects probit, state dependence, unobserved heterogeneity

## 1 Introduction

Dynamic random-effects model specifications are increasingly used in the literature dealing with the persistence of dichotomous outcomes. The aim of the dynamic specifications is to account for state dependence processes, modeled as a $t-1$ lag in the outcome variable ($y_{it-1}$) and representing the amount of inertia in previous statuses of a binary response variable. However, identifying state dependence rests on the assumption of no correlation between unobserved heterogeneity (UH) and the outcome variable $y_{it}$ (Heckman 1981a,c). It follows that by including only $y_{it-1}$, one cannot assess the presence and evaluate the magnitude of the so-called true (or genuine) state dependence.

A second problem may occur in the case of correlation between the initial observation $y_{i0}$ and the relevant unobserved factors, an issue often referred to as the initial condition problem,[1] meaning that the initial period $y_{i0}$ that the researcher observes might not (and

1. See Skrondal and Rabe-Hesketh (2014) for a review of the different strategies used in the literature to solve the initial condition problem.

 

realistically does not) correspond to the beginning of the stochastic process leading to the experience of the outcome. More precisely, while the researcher observes the values in the response variable for the periods $s = 0, \ldots, T$, the stochastic process starts at period $s < 0$.

In the literature, the initial condition problem has been tackled mainly by either modeling the initial response jointly with the subsequent response as proposed by Heckman (1981b) or conditioning on the response at the initial period $y_{i0}$ as proposed by Wooldridge (2005).

The approach that we follow in this article builds on Wooldridge's (2005) "simple solution to the initial condition problem". By conditioning on $y_{i0}$, Wooldridge explicitly models UH by including in the model the values of the time-varying explanatory variables at each period (excluding the initial period). Other authors instead conditioned on $y_{i0}$ but modeled unobserved effects through the inclusion of within-unit averages computed on the time-varying independent variables (Stewart 2007; Biewen 2009).

Even if the use of the within-unit averages has the advantage of parsimony and does not require a balanced panel, such a model specification tends to provide biased estimates because the conditional distribution of the unobserved effects depends more on the value of the initial period than on the values of the other periods of the explanatory variables (Rabe-Hesketh and Skrondal [2013]; see also Skrondal and Rabe-Hesketh [2014]). However, Rabe-Hesketh and Skrondal (2013) have shown that this issue can be solved, as done with this command, by augmenting the model specification with the initial period of the explanatory variables, $\mathbf{Z}_{i0}$. Indeed, they have shown that this specification provides unbiased estimates because it performs as well as Wooldridge's original one while having the advantage of being more flexible. Additionally, this specification is more parsimonious from a parametrical standpoint, and it can be safely implemented also in case of unbalanced panels.

## 2 The model

Following on what has been said above, the model specification that `xtpdyn` estimates has the following form:[2]

$$y^*_{it} = \boldsymbol{\gamma}\mathbf{Z}_{it} + \rho y_{it-1} + c_i + u_{it} \tag{1}$$

The latent outcome variable $y^*_{it}$ in (1) expresses the chances of experiencing a particular status (household-level poverty in the provided example) for unit $i$ ($i = 1, \ldots, N$) at time $t$ as a function of a set of time-varying explanatory variables, $\mathbf{Z}_{it}$, that are considered strictly exogenous, conditional on the unit-specific unobserved effect $c_i$. $y_{it-1}$ captures (genuine) state dependence, while $u_{it}$ is an idiosyncratic error term.

2. The model that we fit corresponds to the model "**P**" proposed by Rabe-Hesketh and Skrondal (2013).

As described, following Rabe-Hesketh and Skrondal (2013), the unit-specific unobserved effect $c_i$ can be written as follows:

$$c_i = \alpha_0 + \alpha_1 y_{i0} + \overline{\mathbf{Z}}_i \boldsymbol{\alpha}_2 + \mathbf{Z}_{i0} \boldsymbol{\alpha}_3 + a_i \tag{2}$$

Here $y_{i0}$ and $\mathbf{Z}_{i0}$ represent the initial value of the response variable and of the time-varying explanatory variables, respectively.[3] $\overline{\mathbf{Z}}_i = 1/T \sum_{i=0}^{T} \mathbf{Z}_{it}$ stands for the within-unit averages of the explanatory variables where the averages are based on all periods $t = 0, \ldots, T$. Finally, $a_i$ is a unit-specific time-constant error term, normally distributed with mean 0 and variance $\sigma_a^2$.

Holding the assumption that UH is captured by $c_i$, the $t-1$ lagged value of the response variable can be interpreted as genuine state dependence—that is, as the causal effect exerted by the experience of poverty in one wave on the experience of poverty status in the subsequent time point.

The model is fit as a standard random-effects probit, and the estimation that `xtpdyn` performs is based on the `meprobit` Stata command.

# 3 The xtpdyn command

`xtpdyn` fits dynamic random-effects probit models with UH. The syntax follows the standard Stata syntax. The user has to specify a) the dependent variable (*depvar*); b) a set of explanatory variables, both time-varying and time-constant (*indepvars*); c) a set of time-varying explanatory variables that contribute to define UH (*varlist*); and d) a series of options.

## 3.1 Syntax

### Model specification

`xtpdyn` *depvar* *indepvars* [ *if* ] [ *in* ] [ *weight* ] `, uh(`*varlist*`)` [ `re(`*re_options*`)` ]

*depvar* is the outcome variable. *indepvars* is the set of explanatory variables. *varlist* is the set of time-varying explanatory variables that contribute to define UH via their means and their initial condition.

### Utilities

`xtpdyn` [ `, keep`|`drop` ]

3. For the sake of clarity, note that the subscript $i0$ refers to the first observation of the unit $i$; $it$ refers to the current time period at which the unit is evaluated, while $it-1$ refers to the time period before the unit is evaluated.

## 3.2 Description

`xtpdyn` fits dynamic random-effects probit models with UH. It implements Wooldridge's (2005) simple solution to the initial condition problem in the alternative proposed by Rabe-Hesketh and Skrondal (2013). UH is addressed by the inclusion in the model of the initial period value of the dependent variable and the initial period and within-unit averages of time-varying explanatory variables.

`xtpdyn` (utilities) shows the last estimates.

## 3.3 Options

### Model options

`uh(`*varlist*`)` specifies time-varying independent variables that contribute to UH in *varlist*. `uh()` is required.

`re(`*re_options*`)` specifies random-effects options for `meprobit`.

### Utilities

`keep` regenerates and keeps the variables in `uh()`.

`drop` drops the variables in `uh()` generated by `keep`.

## 3.4 Stored results

In addition to the results stored by `meprobit`, `xtpdyn` stores the following in `e()`:

Macros

| | |
|---|---|
| `e(depvar)` | name of dependent variable |
| `e(varlist)` | list of independent variables |
| `e(uh)` | list of variables specified in `uh()` |
| `e(if)` | content of the `if` qualifier |

## 3.5 Example

We illustrate the use of the `xtpdyn` command studying poverty persistence and dynamics for a subsample of 1,112 anonymized households in the United States between the years 1970 and 1996.

We begin by loading poverty data and declaring them to be panel data:

```
. use poverty
. xtset id year
```

We present a description of the variables relevant to the analysis in table 1.

Table 1. Description of the variables relevant to the analysis

| Variable name | Description | Values |
|---|---|---|
| `id` | Identification number | min = `1` and max = `1112` |
| `year` | Year of survey | min = `1970` and max = `1996` |
| `poor` | Household poverty | `0` = not poor, `1` = poor |
| `age` | Age of household head | min = `25` and max = `60` |
| `edu` | Education of household head | `0` = less than high school,<br>`1` = high school,<br>`2` = more than high school |
| `black` | Race of household head | `0` = not black, `1` = black |
| `emp` | Employment of household head | `0` = not employed,<br>`1` = part-time, `2` = full-time |
| `marstat` | Marital status of household head | `0` = married, `1` = single,<br>`2` = divorced/separated/widowed |

We study the risk of a household to be poor (`poor`) as a function of the household head's characteristics. These characteristics are both time constant, including race (`black`) and education (`edu`), and time varying, including age (`age`), employment status (`emp`), and marital status (`marstat`). The model is specified as follows:

```
. xtpdyn poor black c.age i.edu i.emp i.marstat,
> uh(i.emp i.marstat age)
```

Note that in `uh()`, we have specified UH to be captured by employment status, marital status, and age. As discussed earlier, UH is captured by the initial value and the within-unit average of time varying explanatory variables and by the initial value of the dependent variable (initial condition). `xtpdyn` automatically produces these variables. In addition, `xtpdyn` automatically includes in the model the variable capturing genuine state dependence as a lag in the dependent variable.

After the estimation, `xtpdyn` automatically drops all the variables created for the analysis. However, we can request that the variables are re-created and kept using

```
. xtpdyn, keep
```

Note that once the variables have been re-created, all the `meprobit` postestimation commands, such as `margins`, are allowed.

Below, we show how the data setup for the analysis looks for the first two households in the sample.

```
. list id year poor age L.poor poor__0 age__0 m__age if id<=2, sepby(id) nolabel
```

| | id | year | poor | age | L. poor | poor__0 | age__0 | m__age |
|---|---|---|---|---|---|---|---|---|
| 1. | 1 | 1992 | 1 | 28 | . | 1 | 28 | 30 |
| 2. | 1 | 1993 | 1 | 29 | 1 | 1 | 28 | 30 |
| 3. | 1 | 1994 | 0 | 30 | 1 | 1 | 28 | 30 |
| 4. | 1 | 1995 | 0 | 31 | 0 | 1 | 28 | 30 |
| 5. | 1 | 1996 | 0 | 32 | 0 | 1 | 28 | 30 |
| 6. | 2 | 1987 | 0 | 45 | . | 0 | 45 | 47.5 |
| 7. | 2 | 1988 | 0 | 46 | 0 | 0 | 45 | 47.5 |
| 8. | 2 | 1989 | 0 | 47 | 0 | 0 | 45 | 47.5 |
| 9. | 2 | 1990 | 1 | 48 | 0 | 0 | 45 | 47.5 |
| 10. | 2 | 1991 | 1 | 49 | 1 | 0 | 45 | 47.5 |
| 11. | 2 | 1992 | 1 | 50 | 1 | 0 | 45 | 47.5 |

The first four columns report the original variables ID, year of the survey, whether the household is poor, and the age of the household head (here we show only age, but the same procedure is applied to all the variables specified in `uh()`). The following columns report the variable capturing genuine state dependence—operationalized via the time-series operator `L.`—and the variables that `xtpdyn` creates:

`L.poor` represents genuine state dependence, measured as a lag in the dependent variable. Therefore, `L.poor` assumes the values of poor at $t-1$. The first observation of each household is by definition missing.

`poor__0` is the initial condition and takes the value of the dependent variable in the first period in which the household is observed; that is, `poor` = `1` for the first household in 1992, and `poor` = `0` for the second household in 1987.

`age__0` represents the value that age assumes in the first period in which the household is observed.

`m__age` represents the within-unit average of age. For example, for the second household that we follow from 45 to 50 years old, the average age is `m__age` = `47.5`.

Before fitting the model and discussing the results, we have to drop the variables created by typing

```
. xtpdyn, drop
 poor__0 emp__0 marstat__0 age__0 m1__emp m2__emp m1__marstat m2__marstat m__age
> have been dropped
```

We now proceed with the results of the analysis and specify the model as above:

```
. xtpdyn poor black c.age i.edu i.emp i.marstat, uh(i.emp i.marstat age)
```

Before the estimates table, `xtpdyn` reports a legend with the names of the variables that it creates:

```
               GSD (Yt-1): L.poor
   Initial condition (Yt0): poor__0
      Initial period of Xs: emp__0 marstat__0 age__0
Within-unit averages of Xs: m1__emp m2__emp m1__marstat m2__marstat m__age
```

(*output omitted*)

```
Mixed-effects probit regression                 Number of obs     =      6,173
Group variable:              id                 Number of groups  =      1,112

                                                Obs per group:
                                                              min =          3
                                                              avg =        5.6
                                                              max =          9

Integration method: mvaghermite                 Integration pts.  =         12

                                                Wald chi2(20)     =    1130.59
Log likelihood =  -1963.425                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
        poor |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
             |
      L.poor |
        Poor |   .5744007   .0739081     7.77   0.000     .4295436    .7192579
             |
       black |   .5225842   .0689301     7.58   0.000     .3874837    .6576847
         age |  -.0649581   .0132202    -4.91   0.000    -.0908692   -.0390471
             |
         edu |
 High School |  -.2264437   .0763974    -2.96   0.003    -.3761798   -.0767076
More than HS |  -.5724683   .0959158    -5.97   0.000    -.7604598   -.3844768
             |
         emp |
   Part-time |  -.4357227    .110222    -3.95   0.000    -.6517539   -.2196914
   Full-time |   -1.23497   .1115673   -11.07   0.000    -1.453637   -1.016302
             |
     marstat |
      Single |  -.6937395   .2086061    -3.33   0.001      -1.1026    -.284879
 Div/sep/wid |  -.5112024   .1601757    -3.19   0.001    -.8251411   -.1972638
             |
   1.poor__0 |    .923642    .092197    10.02   0.000     .7429393    1.104345
             |
      emp__0 |
           1 |   -.148413   .1379748    -1.08   0.282    -.4188387    .1220127
           2 |  -.1200204   .1481667    -0.81   0.418    -.4104218     .170381
             |
  marstat__0 |
           1 |  -.3251233   .2235551    -1.45   0.146    -.7632832    .1130366
           2 |  -.2616084   .2009362    -1.30   0.193     -.655436    .1322192
             |
      age__0 |  -.0311116   .0319239    -0.97   0.330    -.0936812    .0314581
     m1__emp |   .2515404   .2445594     1.03   0.304    -.2277873     .730868
     m2__emp |  -.1801458   .2407334    -0.75   0.454    -.6519747    .2916831
 m1__marstat |     1.0849   .3462806     3.13   0.002     .4062023    1.763597
 m2__marstat |   1.051389   .2765035     3.80   0.000     .5094525    1.593326
      m__age |    .081758   .0342647     2.39   0.017     .0146005    .1489155
       _cons |  -.1005763   .2414719    -0.42   0.677    -.5738524    .3726999
-------------+----------------------------------------------------------------
id           |
   var(_cons)|   .3355826   .0620979                      .2335011    .4822919
------------------------------------------------------------------------------
LR test vs. probit model: chibar2(01) = 66.80         Prob >= chibar2 = 0.0000
```

The output table that xtpdyn produces comes from the underlying meprobit model and can be interpreted accordingly. By default, the number of integration (quadrature) points of the underlying meprobit is set to 12, which resultantly provides estimates that are analogous to xtprobit.

The first coefficient in the table, L.poor, is the lagged value of the dependent variable capturing genuine state dependence. Once controlled for initial condition and net of the role of UH, the positive coefficient L.poor indicates the presence of significant dynamics of genuine state dependence. The output table then reports the coefficients for control variables. As expected, black people are more prone to experience poverty spells, and education and full-time employment significantly stratify the poverty risk, with decreasing exposure associated with higher work intensity and educational endowments. Next, we find the set of coefficients of the variables capturing UH: the initial condition and the initial value and the within-unit averages of time-varying explanatory variables. Here we observe a statistically significant and substantial positive effect of the initial condition poor__0 and the within-unit averages of marital status m1__marstat (single) and m2__marstat (divorced, separated, or widowed). This indicates that these characteristics are correlated with unobserved factors positively associated with poverty or that, said differently, these households are characterized by time-constant unobserved factors that increase their poverty risks.

As said, xtpdyn allows for random-effects options of meprobit. For example, the user may want to estimate robust variance. This can be done using the option re() of xtpdyn and the option vce(robust) of meprobit. The following command fits the model as above but with robust variance—re(vce(robust)):

```
. xtpdyn poor black c.age i.edu i.emp i.marstat,
> uh(i.emp i.marstat age) re(vce(robust))
  (output omitted)
```

## 3.6 xtpdyn by hand: Results replication

In this section, we show how xtpdyn results can be replicated using both xtprobit and meprobit (results not reported).

Before fitting the models, we need to construct the initial condition variables and the within-unit averages for the time-varying explanatory variable. Following on our example, we first generate an indicator to flag nonmissing observations for all the variables we use:

```
. generate sample = !missing(id, year, poor, black, age, edu, emp, marstat)
```

Second, we construct the variable capturing the initial condition:

```
. foreach var of varlist poor age emp marstat {
  2. bysort sample id (year): generate `var´__0 = `var´[1] if sample &
> sample[1]==1
  3. }
```

Third, we also compute the over-time within-units average:

```
. bysort sample id: egen m__age = mean(age)
. bysort sample id: egen m1__emp = mean(emp==1)
. bysort sample id: egen m2__emp = mean(emp==2)
. bysort sample id: egen m1__marstat = mean(marstat==1)
. bysort sample id: egen m2__marstat = mean(marstat==2)
```

The last step is the estimation of the models. We thus `xtset` the data and estimate the `xtprobit`, including a lagged dependent variable, the set of explanatory variables, the initial condition of the time-varying variable, and their within-unit average:

```
. xtset id year
. xtprobit poor iL.poor black c.age i.edu i.emp i.marstat
> i.poor__0 age__0 i.emp__0 i.marstat__0
> m__age m1__emp m2__emp m1__marstat m2__marstat
```

Then, we fit the same model using `meprobit` and set the number of integration points to 12 to mirror the `xtprobit` estimates:

```
. meprobit poor iL.poor black c.age i.edu i.emp i.marstat
> i.poor__0 age__0 i.emp__0 i.marstat__0
> m__age m1__emp m2__emp m1__marstat m2__marstat || id: , intpoints(12)
```

# 4 Postestimation statistics

In addition to `xtpdyn`, we provide a postestimation command, `probat`, that allows one to estimate steady-state expected dynamics, including transition rates and associated statistics, at different levels of the time-constant covariates. `probat` also allows one to evaluate the effect of genuine state dependence across different components and at different levels of UH.

## 4.1 Steady-state expected dynamics

Based on the model's estimates, we derive a set of profile-specific statistics for the predicted patterns of the dependent variable over time: unit transitions into and out of the status (poverty), the expected spell duration, and the ("long-term") steady-state probability.

Exit probabilities $\Pr(0|1)$ are indirectly derived from the estimated $\Pr(1|1)$ and computed as[4]

$$1 - \Pr(1|1) \tag{3}$$

The expected (average) duration of the spell is given by

$$1/\Pr(0|1) \tag{4}$$

4. $\Pr(0|1)$ indicates the probability of not being in the status (that is, poverty) at time $t$ conditional of having been in the status at time $t-1$; $\Pr(1|1)$ indicates the probability of being in the status at time $t$ conditional of having been in the status at $t-1$.

while the steady-state probability or the expected proportion of time in which the unit $i$ displays a positive outcome is computed as

$$[\Pr(1|0)/\{\Pr(1|0)+\Pr(0|1)\}] \tag{5}$$

Both (4) and (5) are computed under the assumption of a steady-state $\mathbf{Z}_{it} = \mathbf{Z}_i$ for all $t$ and stable entry $[\Pr(1|0)]$ and exit $[\Pr(0|1)]$ probabilities (compare Boskin and Nold [1975]; Cappellari and Jenkins [2009]).

Differently from proposals in previous contributions in the literature, these statistics are estimated and net of UH—in this example, net of the set of advantages and disadvantages that households have accumulated over the life course and that are associated with the poverty condition. In other words, these estimates provide an indication of possible accumulation over time of short-run effects for any given profile once time-constant individual unobservables have been accounted for (Immervoll, Jenkins, and Königs 2015). Following Immervoll, Jenkins, and Königs (2015), entry probabilities $\Pr(1|0)$ and persistence $\Pr(1|1)$ are predicted based on the fitted model—which ensures that UH is accounted for in the prediction—and then averaged across individuals—which ensures that the aggregate transition rates account for the distribution of characteristics in the sample. Therefore, ancillary statistics can be considered as netted out from the inertia effects because of unobserved individual characteristics.

More formally, entry and persistence rates are estimated as follows: building on (1), let $\mathbf{X}$ include all the time-constant explanatory variables, the time-varying explanatory variables, and all the variables capturing UH, and let $\boldsymbol{\beta}$ represent a vector of associated coefficients. Then, the estimated entry rate is

$$\Pr(1|0) = \Pr(y_{it} = 1|y_{it-1} = 0, \mathbf{X}) = \Phi(\boldsymbol{\beta}\mathbf{X}) \tag{6}$$

while the estimated persistent rate is

$$\Pr(1|1) = \Pr(y_{it} = 1|y_{it-1} = 1, \mathbf{X}) = \Phi(\rho + \boldsymbol{\beta}\mathbf{X}) \tag{7}$$

where $\Phi(\cdot)$ represents the standard normal cumulative distribution function and $\rho$ the coefficient associated with $y_{it-1}$.

Holding the steady-state assumption, these measures allow for extending the comparisons between profiles moving toward a "long-run" perspective, even when working with relatively short observational windows. On the possibility of a safe extrapolation—even if it is unlikely that these transition probabilities can be assumed to be stable over time—following Cappellari and Jenkins (2009), we still consider the steady-state perspective potentially useful for interpretative purposes because it helps to reveal how the outcome's dynamics differ between households with different characteristics.

At the same time, while profile-specific mean durations and proportions of time spent in poverty might not always have a substantive meaning if considered on their own, they are nonetheless useful for comparisons between profiles or contexts, especially because compositional factors and relevant UH are controlled for.

All of these statistics can be computed using `probat, stats(`*atspec*`)`.

## 4.2 Capturing UH components

Finally, we propose an approach to account for the variation in the marginal effect of genuine state dependence at different levels of UH.

As suggested by the literature [see (1)], UH is meant to be captured by $\alpha_1 y_{i0}$ (initial period of the response variable); $\mathbf{Z}_{i0}\boldsymbol{\alpha}_3$ (initial period of the time-varying explanatory variables); and $\overline{\mathbf{Z}}_i\boldsymbol{\alpha}_2$ (within-unit averages of the time-varying explanatory variables).

Following from this, we can express UH as

$$\text{UH} = \text{UH}_y + \text{UH}_z \tag{8}$$

where the first and the second terms in the right-hand side of the equation refer to UH related to the response variable and the explanatory variables, respectively.

Here, instead of considering an overall distribution of UH, we distinguish the two components such that

$$\text{UH}_y = \begin{cases} y_{i0} = 0 \\ y_{i0} = 1 \end{cases} \tag{9}$$

while

$$\text{UH}_z = \overline{\mathbf{Z}}_i\boldsymbol{\alpha}_2 + \mathbf{Z}_{i0}\boldsymbol{\alpha}_3 \tag{10}$$

Based on these considerations, we jointly evaluate the role of genuine state dependence at the different levels of $\text{UH}_y$ and at different quantiles of the $\text{UH}_z$ distribution; in the example that we provide, the distribution is broken in quintiles. This allows one to compare poverty exposure and genuine state dependence magnitude among households at the same relative position in the $\text{UH}_z$ distribution but to report different initial conditions with respect to the dependent variable $\text{UH}_y$.

Here we estimate $\Pr(1|0)$ and $\Pr(1|1)$ following (6) and (7), respectively. In detail, we estimate group-specific probabilities, where groups are defined by the combination of the individual position in the $\text{UH}_z$ distribution and their initial condition $\text{UH}_y$, namely $\forall_z \in \{1, \ldots, Q\}$, where $Q$ represents the number of quintiles, and $\forall_{y0} \in \{0, 1\}$.

This can be estimated using `probat, prdistr`.

# 5 The probat postestimation command: Syntax and use

`probat` is a postestimation command to be used after `xtpdyn`. It provides estimates of transition probabilities and other statistics.[5]

## 5.1 Syntax

`probat,` {`stats`[`(`*atspec*`)`] | <u>`prd`</u>`istr`} [<u>`ma`</u>`rgins(`*string*`)` <u>`n`</u>`q(`*#*`)` <u>`showf`</u>`req plot keep`]

5. Estimates that `probat` provides are based on the Stata command `margins`.

## 5.2 Options

`stats`[ `(`*atspec*`)` ] defines the profile for which ancillary statistics are computed. Ancillary statistics include entry Pr(1|0), exit Pr(0|1) and persistence Pr(1|1) probabilities, the proportion of time spent in Y=1, the mean duration of the event, and the turnover rate. If only `stats` is specified, `probat` provides ancillary statistics for the overall sample. `stats`[ `(`*atspec*`)` ] may not be used with `prdistr`. In the option `stats`[ `(`*atspec*`)` ], *atspec* is defined as *varname1*`=`*#* [ *varname2*`=`*#* [ ... ] ]. Either `stats`[ `(`*atspec*`)` ] or `prdistr` is required.

`prdistr` computes predicted probabilities Pr(1|0) and Pr(1|1) at different levels of UH. UH is distinguished in two components: $\text{UH}_y$, which is attributable to the initial period of $Y$; and $\text{UH}_z$, which is attributable to the initial period and the within-unit averages of time-varying explanatory variables. Therefore, `prdistr` evaluates the probabilities Pr(1|0) and Pr(1|1) at all the levels of the two UH components jointly. The first component, $\text{UH}_y$, is captured by the levels of $y_{i0}(0/1)$. The second component is defined in terms of quantiles of the $\text{UH}_z$ sample distribution computed on the basis of initial period and the within-unit averages of time-varying explanatory variables. `prdistr` may not be used with `stats`[ `(`*atspec*`)` ]. Either `stats`[ `(`*atspec*`)` ] or `prdistr` is required.

### stats[(atspec)] suboptions

`margins(`*string*`)` specifies options usually allowed with `margins`. See the help file for `margins`.

### prdistr suboptions

`nq(`*#*`)` specifies the number of quantiles used to split the distribution of UH. `nq(`*#*`)` can assume only the following values: 2, 3, 4, 5, and 10. The default is `nq(5)`.

`showfreq` shows frequencies for the outcome by $y_{it-1}$, $\text{UH}_y$, and $\text{UH}_z$.

`plot` plots results of `prdistr`.

`keep` makes changes to the dataset permanent, that is, keeps variables capturing the UH distribution.

`margins(`*string*`)` specifies options usually allowed with `margins`. See the help file for `margins`.

## 5.3 Stored results

If `stats`[ (*atspec*) ] is specified, then `probat` stores the following in `r()`:

Scalars

| | |
|---|---|
| `r(entry_pr)` | entry probability Pr(1\|0) |
| `r(exit_pr)` | exit probability Pr(0\|1) |
| `r(prop_t)` | proportion of time spent in $y = 1$ |
| `r(meandur)` | mean duration |

If either `stats`[ (*atspec*) ] or `prdistr` is specified, then `probat` stores the following in `r()`:

Matrices

| | |
|---|---|
| `r(probest)` | matrix of predicted probabilities |

## 5.4 Example

### The option stats[()]

Based on the model fit in section 3.5 using `xtpdyn`, we now show how the postestimation command `probat` works. The results are based on model estimates and computed as described in (3), (4), and (5). First, we start presenting `stats`. When specified alone, the option `stats` provides time patterns in poverty exposure for the overall sample. However, these statistics can also be computed for a specific profile. In the proposed example, we define the profile in terms of education.

In the first example, we compute the statistics for low-educated households, that is, `edu=0`:

```
. probat, stats(edu=0)
Probability for the profile chosen
```

| poor | Prob. | Std. Err. | P>\|z\| | Lower CI | Upper CI |
|---|---|---|---|---|---|
| Pr(1\|0) | 0.20953 | 0.01169 | 0.00000 | 0.18661 | 0.23244 |
| Pr(1\|1) | 0.33632 | 0.02044 | 0.00000 | 0.29626 | 0.37639 |

| Additional statistics | |
|---|---|
| Entry probability P(1\|0) | 0.20953 |
| Exit probability P(0\|1) | 0.66368 |
| Proportion of T in y=1/Steady state Pr. | 0.23995 |
| Mean duration | 1.50675 |

We then compute the statistics for high-educated households, that is, edu=2:

```
. probat, stats(edu=2) margins(nose)
Probability for the profile chosen
```

| poor | Prob. |
|---|---|
| Pr(1\|0) | 0.11826 |
| Pr(1\|1) | 0.20990 |

| Additional statistics | |
|---|---|
| Entry probability P(1\|0) | 0.11826 |
| Exit probability P(0\|1) | 0.79010 |
| Proportion of T in y=1/Steady state Pr. | 0.13019 |
| Mean duration | 1.26566 |

As expected, it is apparent that, net of other compositional factors, education stratifies poverty dynamics over time, both reducing entry risks (0.12 versus 0.21) and increasing the exit chances (0.79 versus 0.66). Consistently, state dependence, expected mean duration of poverty spell, and the projected steady-state probability of poverty are significantly higher among those holding lower educational degrees.

Note that in the second case, we added the suboption `margins(nose)`, with which we ask `probat` not to estimate standard errors. Being an option of the `margins` command, it has to be specified within the option `margins()`.

`probat` also reports a matrix with the estimated probabilities and a set of scalars reporting the computed statistics.

```
. return list
scalars:
           r(entry_pr) =  .1182633463562954
            r(exit_pr) =  .7901038342265223
             r(prop_t) =  .1301933280773271
            r(meandur) =  1.265656432333298
matrices:
            r(probest) :  2 x 1
. matrix list r(probest)
r(probest)[2,1]
                  b
L.poor=0  .11826335
L.poor=1  .20989617
```

## The option prdistr

In addition to the option `stats()`, `probat` also provides the option `prdistr` to evaluate entry and persistence probabilities for units settled at different points of the distribution of the modeled UH.

In the following command, we also specify `plot`, a suboption of `prdistr`. This suboption graphs the results (figure 1).

```
. probat, prdistr plot
```
(*output omitted*)

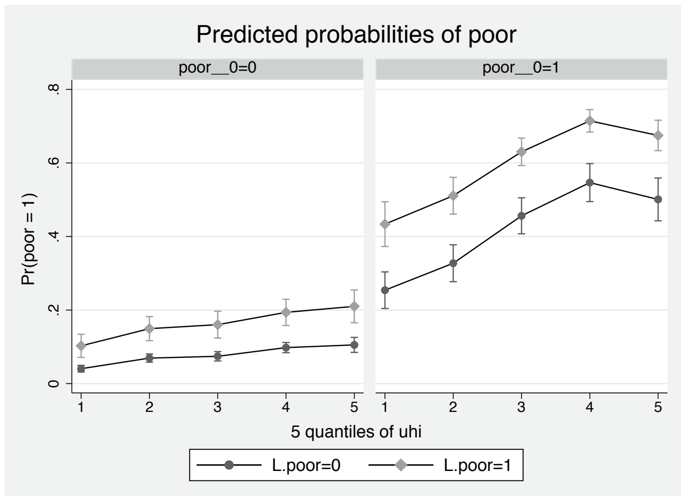


Figure 1. Variation of genuine state dependence over the distribution of UH components

Based on the model fit, figure 1 shows the variation of poverty exposure according to both $\text{UH}_y$ and $\text{UH}_z$ as described in (8), (9), and (10). The figure shows that both the initial condition (`poor__0` $= 0$ and `poor__0` $= 1$ in the left and right subgraphs, respectively) representing $\text{UH}_y$ and the relative individual position in the sample distribution of $\text{UH}_z$ (defined in each subgraph by the five quantiles) substantively affect the model predictions. Comparing left and right subgraphs, we see that units reporting poverty experiences as the initial condition appear more prone to face subsequent poverty spells. This holds regardless of their relative position with respect to $\text{UH}_z$. Moving along the distribution of $\text{UH}_z$, the poverty exposure increases, indicating the independent relevance of $\text{UH}_z$ in shaping poverty risks over time. The same is true concerning the marginal effect of genuine state dependence, represented in figure 1 by the difference between the two lines `L.poor` $= 0$ and `L.poor` $= 1$. In addition, genuine state dependence tends to be higher for those already poor at $t0$ (`poor__0` $= 1$). This result can be conceptualized as a sort of interplay between genuine state dependence and $\text{UH}_y$, because for those with initial poverty experience, each subsequent spell is more likely to reiterate or prolong than for those not reporting poverty as initial condition (`poor__0` $= 0$). The picture does not suggest instead a significant interplay between $\text{UH}_z$ and genuine state dependence, because the marginal effect of the lagged outcome variable (`L.poor`) appears largely unchanged across $\text{UH}_z$ sample distribution.

### Some more suboptions: nq(#), showfreq, and margins()

Finally, in the following command line, we show further suboptions that can be specified with `prdistr`. By specifying `nq(3)`, we split the sample distribution of $\text{UH}_z$ in three (rather than five) quintiles.

```
. probat, prdistr nq(3) showfreq margins(post)
-> poor__0 = 0
```

| poor at time t-1 | 3 quantiles of uhi and Household poverty 1: Not poor | 1: Poor | 2: Not poor | 2: Poor | 3: Not poor | 3: Poor |
|---|---|---|---|---|---|---|
| 0 | 1,595 | 104 | 1,077 | 95 | 1,131 | 94 |
| 1 | 97 | 55 | 81 | 89 | 73 | 87 |

```
-> poor__0 = 1
```

| poor at time t-1 | 3 quantiles of uhi and Household poverty 1: Not poor | 1: Poor | 2: Not poor | 2: Poor | 3: Not poor | 3: Poor |
|---|---|---|---|---|---|---|
| 0 | 71 | 21 | 152 | 61 | 115 | 57 |
| 1 | 53 | 74 | 142 | 351 | 132 | 366 |

The output above is the result of the suboption `showfreq`, which shows the frequency distribution of the outcome (`Not poor` or `Poor`) across the three quintiles of the UH distribution and the outcome at $t-1$, separately for the initial condition status (`poor__0`).

The output below is the table of estimates that `probat` displays (output omitted in the previous example).

| L.poor#poor__0#uh_q | Prob. | Std. Err. | P>\|z\| | Lower CI | Upper CI |
|---|---|---|---|---|---|
| 0 0 1 | 0.0526 | 0.005071 | 0.000 | 0.0426 | 0.0625 |
| 0 0 2 | 0.0815 | 0.006332 | 0.000 | 0.0691 | 0.0940 |
| 0 0 3 | 0.0972 | 0.008110 | 0.000 | 0.0813 | 0.1131 |
| 0 1 1 | 0.2935 | 0.026083 | 0.000 | 0.2424 | 0.3446 |
| 0 1 2 | 0.4713 | 0.025018 | 0.000 | 0.4222 | 0.5203 |
| 0 1 3 | 0.5119 | 0.027147 | 0.000 | 0.4587 | 0.5651 |
| 1 0 1 | 0.1221 | 0.016305 | 0.000 | 0.0902 | 0.1541 |
| 1 0 2 | 0.1707 | 0.018028 | 0.000 | 0.1354 | 0.2061 |
| 1 0 3 | 0.1950 | 0.019936 | 0.000 | 0.1560 | 0.2341 |
| 1 1 1 | 0.4743 | 0.028878 | 0.000 | 0.4177 | 0.5309 |
| 1 1 2 | 0.6440 | 0.018266 | 0.000 | 0.6082 | 0.6798 |
| 1 1 3 | 0.6843 | 0.018162 | 0.000 | 0.6487 | 0.7199 |

`probat, prdistr` automatically stores the above table of estimates in the following matrix:

```
. return list
matrices:
            r(probest) :  12 x 6
```

Similarly to the `stats` option, `prdistr` also supports the suboptions allowed by the Stata command `margins`. In this example, we have specified `margins(post)`, which posts the command estimates.

## 5.5 Postestimation statistics: Results replication

As we have done in section 3 for `xtpdyn`, we now show how the results of the `probat` postestimation command can be replicated using `margins`. As mentioned above, the estimates that `probat` provides are based on the `margins` command.

Let us quietly fit the model and re-create the variables needed for the use of `margins`:

```
. quietly xtpdyn poor black c.age i.edu i.emp i.marstat, uh(i.emp i.marstat age)
. xtpdyn, keep
poor__0 emp__0 marstat__0 age__0 m1__emp m2__emp m1__marstat m2__marstat m__age
> have been created
```

We can now compute entry Pr(0|1) and persistence probabilities Pr(1|1) for low-educated households, as shown in the first example above (`probat, at(edu=0)`):[6]

```
. margins, at(L.poor=(0 1) edu=0) expression(normal(predict(xb))) force
```

Based on these two probabilities, the exit probability, expected average duration, and steady-state probability can be computed following (3), (4), and (5), respectively.

Finally, we can reproduce the results for the entry and persistence probabilities at different levels of UH. Variables capturing UH are automatically computed by `prdistr` and can be kept using the option `keep` as follows:

```
. quietly probat, prdistr keep
```

In particular, the variables `uhi` and `uh_q` that this command creates represent, respectively, the value of the unit-specific UH and the relative position of the unit over the sample distribution of UH, namely, the quantile.

The results presented above can thus be replicated as follows:

```
. margins, at(L.poor=(0 1)) over(poor__0 uh_q)
> expression(normal(predict(xb))) force
```

This command estimates the probability of being poor for households that in the previous year were not poor (`L.poor` $= 0$) and the households that were poor (`L.poor` $= 1$),

6. The option `force` is made necessary to provide an estimate even in the presence of a lagged variable or variables included in the model specification.

according to their poverty experience when they entered the survey (`poor__0`) and their relative position in the sample distribution of UH (`uh_q`).

## 6 Conclusions

In this article, we proposed a command, `xtpdyn`, to fit dynamic random-effects probit models with UH. The specification allows for the estimation of the relevance of genuine state dependence dynamics in accounting for the inertia in dichotomous outcomes over time, net of the role exerted by the possible time constant UH. In addition, we presented a postestimation command, `probat`, that under a steady-state scenario provides different predictions regarding entry and exit rates and average duration of the spell identified by the response variable. Finally, `probat` allows for the variation in the marginal effect of genuine state dependence over the distribution of two distinct components of UH.

## 7 Acknowledgments

We would like to thank the editors and an anonymous reviewer whose comments led to improvements in both `xtpdyn` and the article.

## 8 References

Biewen, M. 2009. Measuring state dependence in individual poverty histories when there is feedback to employment status and household composition. *Journal of Applied Econometrics* 24: 1095–1116.

Boskin, M. J., and F. C. Nold. 1975. A Markov model of turnover in aid to families with dependent children. *Journal of Human Resources* 10: 467–481.

Cappellari, L., and S. P. Jenkins. 2009. The dynamics of social assistance benefit receipt in Britain. Working Paper Series No. 2009-29, Institute for Social & Economic Research. https://www.iser.essex.ac.uk/files/iser_working_papers/2009-29.pdf.

Heckman, J. J. 1981a. Heterogeneity and state dependence. In *Studies in Labor Markets*, ed. S. Rosen, 91–140. Chicago: University of Chicago Press.

———. 1981b. The incidental parameters problem and the problem of initial conditions in estimating a discrete time—Discrete data stochastic process. In *Structural Analysis of Discrete Data with Econometric Applications*, ed. C. F. Manski and D. McFadden, 179–195. Cambridge: MIT Press.

———. 1981c. Statistical models for discrete panel data. In *Structural Analysis of Discrete Data with Economic Applications*, ed. C. F. Manski and D. McFadden, 114–178. Cambridge, MA: MIT Press.

Immervoll, H., S. P. Jenkins, and S. Königs. 2015. Are recipients of social assistance 'benefit dependent'? Concepts, measurement and results for selected countries. IZA Discussion Paper No. 8786, Institute of Labor Economics (IZA). ftp://ftp.iza.org/dps/dp8786.pdf.

Rabe-Hesketh, S., and A. Skrondal. 2013. Avoiding biased versions of Wooldridge's simple solution to the initial conditions problem. *Economics Letters* 120: 346–349.

Skrondal, A., and S. Rabe-Hesketh. 2014. Handling initial conditions and endogenous covariates in dynamic/transition models for binary data with unobserved heterogeneity. *Journal of the Royal Statistical Society, Series C* 63: 211–237.

Stewart, M. B. 2007. The interrelated dynamics of unemployment and low-wage employment. *Journal of Applied Econometrics* 22: 511–531.

Wooldridge, J. M. 2005. Simple solutions to the initial conditions problem in dynamic, nonlinear panel data models with unobserved heterogeneity. *Journal of Applied Econometrics* 20: 39–54.

## About the authors

Raffaele Grotti is a research fellow at the Department of Political and Social Sciences, European University Institute, Italy. His research interests include socioeconomic inequality, life-course, income dynamics, welfare and labor market in comparative perspective, and quantitative methods.

Giorgio Cutuli, PhD, is currently a research fellow at the Department of Sociology and Social Research, University of Trento, Italy. His main research interests include welfare and labor market dynamics, socioeconomic inequality, and quantitative longitudinal analysis.