



DPB: Dynamic Panel Binary Data Models in gretl

Riccardo Lucchetti

Università Politecnica delle Marche

Claudia Pigni

Università Politecnica delle Marche

Abstract

This paper presents the **gretl** function package **DPB** for estimating dynamic binary models with panel data. The package contains routines for the estimation of the random-effects dynamic probit model proposed by Heckman (1981b) and its generalisation by Hyslop (1999) and Keane and Sauer (2009) to accommodate AR(1) disturbances. The fixed-effects estimator by Bartolucci and Nigro (2010) is also implemented. **DPB** is available on the **gretl** function packages archive.

Keywords: **gretl** function package, random-effects dynamic probit model, quadratic exponential model, gauss-hermite quadrature, simulated maximum likelihood, conditional maximum likelihood.

1. Introduction

Non-linear dynamic models for binary dependent variables with longitudinal data are nowadays quite common in microeconomic applications, especially given the increasing availability of panel datasets. One of the most attractive features of models of the type

$$\begin{aligned}y_{it}^* &= \gamma y_{it-1} + \mathbf{x}_{it}^\top \boldsymbol{\beta} + \alpha_i + \varepsilon_{it}, \\y_{it} &= 1\{y_{it}^* \geq 0\} \quad \text{for } i = 1, \dots, n, \quad t = 2, \dots, T,\end{aligned}$$

where $1\{\cdot\}$ is an indicator function, is that they lend themselves very naturally to an interpretation in terms of *true state dependence*, that is a situation in which the *realization* of an event affects the *probability* of the same event occurring in the future, as opposed to simple time persistence in y_{it}^* , which may be due to covariates \mathbf{x}_{it} and/or to the time-invariant unobserved heterogeneity (Heckman 1981a).

This feature is particularly attractive when the problem at hand displays some form of path dependence. Therefore, these models have been found to be extremely useful in the analysis of several microeconomic topics: labor force participation, more specifically female labor supply

(Heckman and Borjas 1980; Hyslop 1999; Carrasco 2001; Arulampalam 2002; Stewart 2007; Keane and Sauer 2009), self-assessed health condition (Contoyannis, Jones, and Rice 2004; Heiss 2011; Halliday 2008; Carro and Traferri 2014), poverty transitions (Cappellari and Jenkins 2004; Biewen 2009), unionization of workers (Stewart 2006), welfare participation (Wunder and Riphahn 2014), remittance decisions by migrants (Bettin and Lucchetti 2016), access to credit and household finance (Alessie, Hochguertel, and van Soest 2004; Brown, Ghosh, and Taylor 2014; Giarda 2013; Pignini, Presbitero, and Zazzaro 2014).

While static models are relatively mainstream and are supported by most statistical and econometric software, dynamic models are more complex to implement and, therefore, estimation routines are not always readily available to the practitioner. Compared to their static counterparts, dealing with unobserved heterogeneity in these models raises several complex issues, both statistical and computational in nature. The main statistical problem lies in the so-called “initial conditions problem”: Clearly, the recursive nature of the model calls for some kind of conditioning when writing the log-likelihood. This, however, is made problematic by the existence of the time-invariant unobserved individual effects α_i , which is correlated with the initial observation.

Random-effects (RE henceforth) approaches tackle the initial conditions problem by modeling the joint distribution of the outcomes conditional on y_1 . Historically, the first proposal is due to Heckman (1981b) who, building on the static RE estimator, proposed a model for the joint distribution for the response variable $\mathbf{y}_i = [y_{i1}, \dots, y_{iT}]$, in which a separate reduced-form model for the initial observation y_{i1} is approximated via a linearized index function. Under suitable distributional assumptions, the joint log-likelihood for \mathbf{y}_i may be evaluated by means of Gauss-Hermite quadrature (Butler and Moffitt 1982) and estimation may be carried out by maximum likelihood (ML).

Generalizations of Heckman’s estimator were later proposed by Hyslop (1999), who introduced autoregressive error terms, and Keane and Sauer (2009), who further generalized it to a model with a more flexible treatment of the initial condition equation. In these cases, however, it is necessary to evaluate multivariate normal integrals with an arbitrary correlation structure among error terms and unobserved heterogeneity. This calls for simulation techniques, such as GHK (Geweke 1989; Hajivassiliou and McFadden 1998; Keane 1994), which make estimation considerably more costly in terms of CPU requirements.

Alternatively, Wooldridge (2005) proposed modeling the distribution of individual unobserved effects α_i conditional on y_1 and on the history of covariates instead of dealing with the joint distribution of all outcomes.¹ Wooldridge’s estimator employs techniques for dealing with the initial observation problem in such a way that estimation can be carried out through ordinary RE probit routines with the addition of some ad-hoc explanatory variables. This alternative has become extremely popular, but unfortunately does not lend itself to a natural generalization to the case with autocorrelated disturbances, which is often called for in practice.

The initial conditions problem can also be circumvented via a fixed-effects (FE) approach, which makes it possible to estimate the regression parameters consistently without having to make distributional assumptions on the unobserved heterogeneity. The key idea is to condition the joint distribution of \mathbf{y}_i on a suitably defined sufficient statistic for α_i . For static

¹Similar approaches have been proposed by Orme (1997, 2001) and Arulampalam and Stewart (2009) which we refer to for a more detailed discussion.

logit models, it is possible to define a conditional maximum likelihood (CML) estimator. Its dynamic extension, however, has not gained widespread adoption in empirical work since it cannot be easily generalized to every time-configuration of the panel and requires strong restrictions to the model specification. Moreover, these models generally require that at least a transition between the states 0 and 1 is observed for the individual to contribute to the likelihood. As a result, the number of usable observations often reduces drastically compared to the sample size, especially in the cases when strong persistence in the dependent variable is exactly the reason why a dynamic model is needed. Nevertheless, estimators based on a FE approach always represent a useful complement to the practitioner’s toolkit, as they provide reliable inference in cases where the unobserved heterogeneity may be correlated with the individual covariates.

The first proposal of a FE logit model can be found in Chamberlain (1985): Estimation relies on conditional inference and, therefore, is rather simple to perform. Exogenous covariates, however, cannot be included and the proposed sufficient statistic for incidental parameters needs to be determined on a case-wise basis according to the time-series length. Honoré and Kyriazidou (2000) extended Chamberlain’s formulation in order to include explanatory variables; this approach, however, implies the usage of semi-parametric techniques that require a substantial computational effort. Moreover, time-dummies (a customary addition to practically all empirical models) cannot be handled either. Recently, Bartolucci and Nigro (2010) defined a dynamic model, which belongs to the QE family. Their proposed model closely resembles a dynamic logit model and overcomes most of the difficulties encountered by Honoré and Kyriazidou’s estimator and can be implemented by suitably adapting ordinary static FE logit software.²

In this work, we present the **gretl** (gretl Team 2016) implementation of the available set of tools to estimate dynamic models for binary dependent variables in panel datasets by both FE and RE approaches, collected in the **DPB** (*dynamic panel binary*) function package. The RE models contained in **DPB** are the dynamic probit with linearized index initial condition proposed in Heckman (1981b) and the generalizations by Hyslop (1999) and Keane and Sauer (2009). Compared to the other available estimators based on a RE approach, Heckman’s estimator has been shown to suffer from remarkably little small-sample bias (Miranda 2007; Akay 2012) and is widely used in microeconomic applications. On the contrary, the estimator proposed by Wooldridge (2005) is not included in the package *per se*, despite its common use by practitioners, since **gretl** already provides suitable functions natively: Therefore, we just provide an example showing how to implement it via a simple script. Finally, **DPB** also contains the software for estimating the QE model in Bartolucci and Nigro (2010).

The **DPB** function package represents a more comprehensive toolbox to estimate dynamic binary RE models compared with other available software. Stewart (2006) implemented the Stata (StataCorp. 2015) commands `redprobit` and `redpace` for the dynamic probit models proposed by Heckman (1981b) and Hyslop (1999), respectively. Compared with Stewart’s module, apart from including the latest proposal by Keane and Sauer (2009), **DPB** handles unbalanced panel datasets, which is a fundamental feature for practitioners working with real-life datasets. As for the available software to estimate QE models, routines are available in the

²Estimators based on the FE approach for long panels ($T \rightarrow \infty$) have also been proposed, among which Hahn and Newey (2004), Carro (2007), Fernández-Val (2009), Hahn and Kuersteiner (2011), Bartolucci, Bellio, Sartori, and Salvan (2016). Here we focus only on short panels ($n \rightarrow \infty$), which are the datasets usually available to applied microeconomists.

R (R Core Team 2017) package and Stata module **cquad** (Bartolucci and Pignini 2017). These packages include the QE model of Bartolucci and Nigro (2010), as well as some extensions³ that will be included in future versions of **DPB**.

The rest of the paper is organized as follows: Section 2 lays the methodological background for the estimators provided in the package, while in Section 3 we discuss the main computational issues of their implementation; Section 4 describes in detail the features of **DPB**; Section 5 provides an empirical application based on a dataset of unionized workers extracted from the US National Longitudinal Survey of Youth; Section 6 concludes.

2. Methodological background

A general model for the conditional probability of a binary response variable y_{it} can be written as

$$p(y_{it}|\mathcal{F}_t, \alpha_i; \boldsymbol{\psi}_0) \quad \text{for } i = 1, \dots, n, \quad t = 1, \dots, T,$$

where \mathcal{F}_t is the information set at time t available for individual i , which, in general, may consist of a set of individual covariates $\mathbf{X}_i = [\mathbf{x}_{i1}, \dots, \mathbf{x}_{iT}]$ as well as the lag of the response variable $\mathbf{y}_i^{t-1} = [y_{i1}, \dots, y_{it-1}]$; α_i is the individual time-invariant unobserved effect, which is assumed to be a continuous r.v.; $\boldsymbol{\psi}_0$ is the vector of model parameters. In the simple case where the relevant conditioning information set is \mathbf{X}_i , i.e., y_{it} does not depend on \mathbf{y}_i^{t-1} , the joint probability of $\mathbf{y}_i = [y_{i1}, \dots, y_{iT}]$ for individual i is

$$p(\mathbf{y}_i|\mathbf{X}_i; \boldsymbol{\psi}_0) = \int_{\mathbb{R}} \prod_{t=1}^T p(y_{it}|\mathbf{X}_i, \alpha_i; \boldsymbol{\psi}_0) dF(\alpha_i).$$

In a dynamic model y_{it} is allowed to depend on its past history. When unobserved effects are present, the process for the response variable needs to be initialized in order to account for how \mathbf{y}_i relates to the process before the observations started being available. The dependence on the past of y_{it} is usually likely to be limited to its first lag, so that the relevant subset of \mathcal{F}_t is $[y_{it-1}, \mathbf{X}_i]$. Therefore, the joint probability of \mathbf{y}_i , conditional on \mathbf{X}_i , becomes

$$p(\mathbf{y}_i|\mathbf{X}_i; \boldsymbol{\psi}_0) = \int_{\mathbb{R}} p(y_{i1}|\mathbf{X}_i, \alpha_i; \boldsymbol{\psi}_0) \prod_{t=2}^T p(y_{it}|y_{it-1}, \mathbf{X}_i, \alpha_i; \boldsymbol{\psi}_0) dF(\alpha_i).$$

The rest of the section describes the RE approach proposed by Heckman (1981b) and its generalizations and the FE approach put forward by Bartolucci and Nigro (2010). In the first case, a reduced form equation for $p(y_{i1}|\mathbf{X}_i, \alpha_i; \boldsymbol{\psi}_0)$ is specified. In the second case, the unobserved effect α_i is eliminated by conditioning $p(\mathbf{y}_i|\mathbf{X}_i, \alpha_i; \boldsymbol{\psi}_0)$ on a suitable sufficient statistic and, as a result, the initial condition does not need to be dealt with, so that the joint probability can be written as $p(\mathbf{y}_i|\mathbf{X}_i, y_{i1}, \alpha_i; \boldsymbol{\psi}_0)$.

³The pseudo CML estimator in Bartolucci and Nigro (2012) and the test for state dependence based on a modified QE model in Bartolucci, Nigro, and Pignini (2017).

2.1. Random-effects approach

The estimator proposed by Heckman (1981b) is based on a standard formulation of a dynamic RE binary choice model with an additional equation for the initial observation y_{i1} :

$$y_{it} = 1\{\gamma y_{it-1} + \mathbf{x}_{it}^\top \boldsymbol{\beta} + \alpha_i + \varepsilon_{it} \geq 0\} \quad \text{for } i = 1, \dots, n, \quad t = 2, \dots, T, \quad (1)$$

$$y_{i1} = 1\{\mathbf{z}_{i1}^\top \boldsymbol{\pi} + \theta \alpha_i + \varepsilon_{i1} \geq 0\} \quad \text{for } i = 1, \dots, n, \quad (2)$$

where y_{it} is the binary response variable, $1\{\cdot\}$ is an indicator function, \mathbf{x}_{it} is a vector of individual covariates and \mathbf{z}_{i1} contains the values of \mathbf{X}_i in the first period and pre-sample information that can be used as exclusion restrictions. The assumptions on α_i and ε_i are: $\mathbb{E}[\varepsilon_{it} | \mathbf{X}_i, \alpha_i] = 0$; orthogonality of α_i and \mathbf{X}_i , $\mathbb{E}[\alpha_i | \mathbf{X}_i] = 0$; joint normality conditional on \mathbf{X}_i , $[\theta \alpha_i + \varepsilon_{i1}, \alpha_i + \varepsilon_{i2}, \dots, \alpha_i + \varepsilon_{iT}]^\top \sim N(\mathbf{0}; \boldsymbol{\Sigma})$ with

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 + \theta^2 \sigma_\alpha^2 & \theta \sigma_\alpha^2 & \theta \sigma_\alpha^2 & \dots \\ \theta \sigma_\alpha^2 & 1 + \sigma_\alpha^2 & \sigma_\alpha^2 & \dots \\ \theta \sigma_\alpha^2 & \sigma_\alpha^2 & 1 + \sigma_\alpha^2 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \quad (3)$$

where $\sigma_\alpha^2 = V(\alpha_i)$.

In this paper, this model will be referred to as the ‘‘DP’’ model (*dynamic probit*). Note that the distributional assumption implies independence between α_i and ε_i and absence of autocorrelation in ε_i , i.e., $\mathbb{E}[\varepsilon_{it}\varepsilon_{is}] = 0$ for $t, s = 1, \dots, T$ and $t \neq s$. Under these assumptions, the parameter vector $\boldsymbol{\psi} = [\boldsymbol{\beta}^\top, \gamma, \boldsymbol{\pi}^\top, \theta, \sigma_\alpha]$ can be estimated by ML, and the i th contribution to the likelihood is given by the following expression:

$$\mathcal{L}_i(\boldsymbol{\psi}) = \int_{\mathbb{R}} \Phi \left[(\mathbf{z}_{i1}^\top \boldsymbol{\pi} + \theta \alpha_i)(2y_{i1} - 1) \right] \prod_{t=2}^T \Phi \left[(\gamma y_{it-1} + \mathbf{x}_{it}^\top \boldsymbol{\beta} + \alpha_i)(2y_{it} - 1) \right] d\Phi \left(\frac{\alpha_i}{\sigma_\alpha} \right),$$

where $\Phi(\cdot)$ is the standard normal c.d.f. Since the unobserved effects are normally distributed, the integral over α_i can be evaluated by means of Gauss-Hermite quadrature (Butler and Moffitt 1982).

Hyslop (1999) considered an interesting generalization of Heckman’s approach by allowing for autocorrelation in ε_i . In terms of interpretation, this setting makes it possible to further disentangle two different sources of time persistence: In Heckman (1981b), the *true* state dependence captured by γ in Equation 1 is isolated from the persistence induced by time-invariant unobserved effects α_i ; with autocorrelated errors, the persistence in the time-varying unobserved effects is also parametrized. The error terms ε_{it} follow the AR(1) process

$$\varepsilon_{it} = \rho \varepsilon_{it-1} + \eta_{it} \quad \text{for } t = 2, \dots, T,$$

where $|\rho| \leq 1$ and $\eta_{it} \sim N(0, 1 - \rho^2)$. Therefore, the variance-covariance matrix of the error components needs to be modified as follows:

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 + \theta^2 \sigma_\alpha^2 & \rho + \theta \sigma_\alpha^2 & \rho^2 + \theta \sigma_\alpha^2 & \dots \\ \rho + \theta \sigma_\alpha^2 & 1 + \sigma_\alpha^2 & \rho + \sigma_\alpha^2 & \dots \\ \rho^2 + \theta \sigma_\alpha^2 & \rho + \sigma_\alpha^2 & 1 + \sigma_\alpha^2 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \quad (4)$$

Note that Equation 4 reduces to Equation 3 for $\rho = 0$; hence, we will call this the “ADP” model (*AR1 dynamic probit*).

Recently, Keane and Sauer (2009) introduced a more general version of Σ in which an additional parameter is defined. The starting point is to modify Equation 2 as

$$y_{i1} = 1\{\mathbf{z}_{i1}^\top \boldsymbol{\pi} + \theta\alpha_i + u_i \geq 0\},$$

with $\mathbf{E}(u_i \cdot \varepsilon_{i2}) = \tau$. Note that since both u_i and ε_{it} need to be normalized for identification, τ is effectively a correlation coefficient, so $|\tau| \leq 1$ holds by the Cauchy-Schwartz inequality. Therefore, Equation 4 becomes: $[\theta\alpha_i + u_i, \alpha_i + \varepsilon_{i2}, \dots, \alpha_i + \varepsilon_{iT}]^\top \sim N(\mathbf{0}; \Sigma)$ with

$$\Sigma = \begin{bmatrix} 1 + \theta^2\sigma_\alpha^2 & \tau\rho + \theta\sigma_\alpha^2 & \tau\rho^2 + \theta\sigma_\alpha^2 & \dots \\ \tau\rho + \theta\sigma_\alpha^2 & 1 + \sigma_\alpha^2 & \rho + \sigma_\alpha^2 & \dots \\ \tau\rho^2 + \theta\sigma_\alpha^2 & \rho + \sigma_\alpha^2 & 1 + \sigma_\alpha^2 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \quad (5)$$

Notice that Equation 5 is equal to Equation 4 for $\tau = 1$ and equal to Equation 3 for $\rho = 0$. The acronym we will use henceforth for this model is “GADP” (*generalized ADP*).

In theory, its greater generality makes model (5) more appealing for applied work than its restricted counterpart Equation 4 and, *a fortiori*, Equation 3. However, there are two factors that may make it advisable to opt for the more restrictive versions of the model. The first one is computational complexity: The specification of ε_{it} as an AR(1) process makes it impossible to evaluate the joint probability $p(\cdot)$ by integrating out the random effect α_i . Instead, T -variate normal probabilities must be evaluated by simulation by using the GHK algorithm (Geweke 1989; Hajivassiliou and McFadden 1998; Keane 1994) in order to compute the likelihood function:

$$\mathcal{L}_i^*(\boldsymbol{\psi}) = \frac{1}{R} \sum_{r=1}^R \Phi_{Tr}^*(a_i, \mathbf{b}_i, \mathbf{C})$$

with $a_i = (\mathbf{z}_{i1}^\top \boldsymbol{\pi})(2y_{i1} - 1)$ and $\mathbf{b}_i = [b_{i2}, \dots, b_{iT}]$, where $b_{it} = (\gamma y_{it-1} + \mathbf{x}_{it}^\top \boldsymbol{\beta})(2y_{it} - 1)$; \mathbf{C} is the lower-triangular Cholesky factor of Σ , defined in Equation 4 or Equation 5, and r is the number of random draws used in the simulation. The second one is that, in finite samples, the ADP and GADP models (especially the latter) may suffer from serious identification problems for certain regions of the parameter space; in some cases, this could cause serious numerical problems for conducting inference, because of insufficient curvature of the log-likelihood function. These issues will be described in greater detail in Section 3.4.

2.2. Fixed-effects approach

With the exception of linear probability models, FE approaches to the estimation of dynamic binary choice models are usually based on the dynamic logit formulation. In this case, model (1) is subject to strict exogeneity conditional on α_i and it is assumed that the error terms ε_{it} , $t = 1, \dots, T$, are i.i.d. standard logistic random variables. Therefore, the probability of observing y_{it} can be written as

$$p(y_{it}|y_{it-1}, \mathbf{x}_{it}, \alpha_i; \boldsymbol{\psi}) = \frac{\exp\left[y_{it} \left(\gamma y_{it-1} + \mathbf{x}_{it}^\top \boldsymbol{\beta} + \alpha_i\right)\right]}{1 + \exp\left(\gamma y_{it-1} + \mathbf{x}_{it}^\top \boldsymbol{\beta} + \alpha_i\right)}.$$

As is well known (Chamberlain 1985), in the static case ($\gamma = 0$) the *total score* $y_{i+} \equiv \sum_t y_{it}$ is a sufficient statistic for α_i , on which the distribution of \mathbf{y}_i can be conditioned, so as to remove the unobserved individual effect α_i ; unfortunately, in the dynamic logit model ($\gamma \neq 0$) equivalent sufficient statistics may exist, but they must be derived on a case-wise basis. Moreover, the inclusion of explanatory variables requires restrictions on their distribution and a non-negligible computational effort (Honoré and Kyriazidou 2000).

The joint probability of observing the response configuration \mathbf{y}_i conditional on the initial observation y_{i1} is

$$p(\mathbf{y}_i | \mathbf{X}_i, y_{i1}, \alpha_i; \boldsymbol{\psi}) = \frac{\exp\left(\sum_t y_{it} y_{it-1} \gamma + \sum_t y_{it} \mathbf{x}_{it}^\top \boldsymbol{\beta} + y_{i+} \alpha_i\right)}{\prod_t [1 + \exp(\gamma y_{it-1} + \mathbf{x}_{it}^\top \boldsymbol{\beta} + \alpha_i)]}, \quad (6)$$

where sums and products go from $t = 2$ to T .

The QE model proposed by Bartolucci and Nigro (2010) directly defines the joint probability of \mathbf{y}_i as

$$p(\mathbf{y}_i | \mathbf{X}_i, y_{i1}, \alpha_i; \boldsymbol{\psi}) = \frac{\exp\left(\sum_t y_{it} y_{it-1} \gamma + \sum_t y_{it} \mathbf{x}_{it}^\top \boldsymbol{\beta}_1 + y_{iT} (\mu + \mathbf{x}_{iT}^\top \boldsymbol{\beta}_2) + y_{i+} \alpha_i\right)}{\sum_{\mathbf{b} \in \mathbb{B}} \exp\left(\sum_t b_t b_{t-1} \gamma + \sum_t b_t \mathbf{x}_{it}^\top \boldsymbol{\beta}_1 + b_T (\mu + \mathbf{x}_{iT}^\top \boldsymbol{\beta}_2) + b_+ \alpha_i\right)}, \quad (7)$$

where $b_+ \equiv \sum_t b_t$ and $\mathbb{B} \equiv \{\mathbf{b} : \mathbf{b} \in \{0, 1\}^T\}$, that is the set of all possible T -vectors \mathbf{b} containing zeros and ones.

This probability closely resembles the one in Equation 6 and, under Equations 6 and 7, Bartolucci and Nigro (2010) show that γ has the same interpretation in terms of log-odds ratio between each pair of consecutive y_{it} s. It can be shown that the total score y_{i+} is a sufficient statistic for the unobserved effects α_i in Equation 7. The conditional distribution based on the total score is

$$\begin{aligned} p(\mathbf{y}_i | \mathbf{X}_i, y_{i1}, y_{i+}; \boldsymbol{\psi}) &= \frac{p(\mathbf{y}_i | \mathbf{X}_i, y_{i1}, \alpha_i; \boldsymbol{\psi})}{p(y_{i+} | \mathbf{X}_i, y_{i1}, \alpha_i; \boldsymbol{\psi})} \\ &= \frac{\exp\left[\sum_t y_{it-1} y_{it} \gamma + \sum_t y_{it} \mathbf{x}_{it}^\top \boldsymbol{\beta}_1 + y_{iT} (\mu + \mathbf{x}_{iT}^\top \boldsymbol{\beta}_2)\right]}{\sum_{\mathbf{b}: b_+ = y_{i+}} \exp\left[\sum_t b_t b_{t-1} \gamma + \sum_t b_t \mathbf{x}_{it}^\top \boldsymbol{\beta}_1 + b_T (\mu + \mathbf{x}_{iT}^\top \boldsymbol{\beta}_2)\right]}, \quad (8) \end{aligned}$$

so that the denominator contains only those vectors $\mathbf{b} \in \mathbb{B}$ such that $b_+ = y_{i+}$. By using the above conditional probability, the conditional log-likelihood can be written as

$$\ell(\boldsymbol{\psi}) = \sum_{i=1}^n 1\{0 < y_{i+} < T\} \log p(\mathbf{y}_i | \mathbf{X}_i, y_{i1}, y_{i+}; \boldsymbol{\psi})$$

and maximized with respect to $\boldsymbol{\psi} = [\gamma, \boldsymbol{\beta}_1^\top, \mu, \boldsymbol{\beta}_2^\top]^\top$. We refer the reader to Bartolucci and Nigro (2010) for details on the score and Hessian of $\ell(\boldsymbol{\psi})$.

3. Computation and numerical issues

3.1. Treatment of missing values

The **DPB** package can handle unbalanced panels, as long as there are enough consecutive observations for each longitudinal unit. If we use T_i to indicate the maximum time span of consecutive observations for individual i , the requirements for a cross-sectional unit i to be included in the sample are

- $T_i \geq 2$ for the DP, ADP and GADP models;
- $T_i > y_{i+} > 0$ for the QE models.

For each unit, the longest available consecutive set of observations is used. If more than one sequence is available, we take the most recent one to compute T_i . Of course, the choices above imply the assumption that observability of $y_{i,t}$ and/or $\mathbf{x}_{i,t}$ is totally independent of the random variables included in the models. For further details, see [Albarrán, Carrasco, and Carro \(2015\)](#).

3.2. Computation of the log-likelihood and its derivatives

The two most important algorithms employed for computing the log-likelihood of the four models handled by the package are Gauss-Hermite quadrature for the DP model and the GHK algorithm for the ADP and GADP models.⁴ As for the QE model, the main problem lies in efficient computation of the denominator in Equation 8, which does not include problematic functions, but is made tricky by the multiplicity of cases to consider. Fortunately, a recursive approach is possible, which we will describe in Section 3.3.

For Gauss-Hermite quadrature, the **DPB** function package uses the **gretl** native function `quadtable`, which guarantees good speed and accuracy. The number of quadrature points can be chosen by the user, with a default of 24. The **DPB** package does not include, at present, the option of using adaptive quadrature methods as recommended, for example, in [Rabe-Hesketh, Skrondal, and Pickles \(2002\)](#).

As for simulation-based methods, **DPB** relies on the **gretl** function `ghk`, which natively provides analytic derivatives. This function does not implement optimization techniques such as the pivoting method by [Genz \(1992\)](#), because Genz's method may introduce discontinuities that would make numerical differentiation problematic. On the other hand, it automatically switches to a parallel implementation on a multi-core machine in a shared-memory environment, which gives a noticeable performance boost on modern CPUs.⁵ The default method for feeding the uniform sequence to the GHK algorithm is by using Halton sequences ([Halton 1964](#)), but the user can switch to the uniform generator used by default in **gretl** (the SIMD-oriented implementation of the Mersenne Twister algorithm described in [Saito and Matsumoto 2008](#)) if so desired.

⁴It is perfectly possible to use the GHK technique for estimating the DP model, but it would be rather inefficient. However, it may be interesting to do so for comparison purposes, so the package makes it possible via an option.

⁵**gretl** also provides MPI extensions for distributed-memory architectures, but they have not been used in the **DPB** package.

The choice method for optimization is BFGS with analytical derivatives for the probit models DP, ADP and GADP; this has generally proven quite effective and remarkably more robust and efficient than Newton-Raphson. Alternatively, its limited-memory variant described in [Byrd, Lu, Nocedal, and Zhu \(1995\)](#) can also be used. Two of the parameters in the covariance matrix Σ are in fact maximized via an invertible transformation to help numerical stability: σ_α^2 is expressed as its natural logarithm and ρ through the (inverse) hyperbolic tangent transform, so that the parameters that enter the actual maximization routine are unbounded. Initial values for β , γ , and π are obtained by straightforward linear probability models, while σ_α and θ are both set to 1.

For the QE model, instead, the method of choice is Newton-Raphson, which takes advantage of the fact that the computation of the Hessian matrix is remarkably inexpensive once the analytical score has been obtained. Initial values for γ , β_1 , μ and β_2 are simply 0.⁶

3.3. Computation of the denominator term in the QE model

The peculiar nature of the QE model makes it impossible to compute its denominator in a manner akin to the algorithm used in several packages for the ordinary conditional logit model, which is in turn a variation of the recursive algorithm by [Krailo and Pike \(1984\)](#).

The algorithm we implement has the virtue of relegating the recursive computation of the relevant combinations (\mathbf{b} vectors) to the initialization stage; by memorizing the relevant information into an array of matrices, we avoid recursion at each computation of the likelihood. Our algorithm can be briefly described as follows.

The denominator in Equation 8 can be easily written as a function of a matrix Q , whose size and elements are function of three scalars: T_i , y_{i+} and y_{i*} , where $y_{i*} = \sum y_{it}y_{it-1}$ is the number of consecutive ones in \mathbf{y}_i .

Define an injective index function $j = j(T, y_+, y_*)$ (where we omit the i subscript for conciseness). Obviously, all individuals in the sample with the same values for T , y_+ and y_* can share the same j index. The idea is to pre-generate all the needed Q_j matrices and store them in an array; since $T_i \geq y_{i+} > y_{i*}$ holds, the array can have at most $T_{\max}^2(T_{\max} - 1)$ elements, although in practice the number of distinct elements in the array will be much smaller. In a typical panel data setting, it is unlikely that more than a few hundred distinct Q_j matrices will have to be computed and stored in memory.

The internal structure of Q_j is

$$Q_j = \left[Q^1(T, y_+) | q^2(y_*) \right], \quad (9)$$

where

1. $Q^1(T, y_+)$ is a matrix with $\frac{T!}{y_+!(T-y_+)!}$ rows and T columns whose rows are $\mathbf{b} : b_+ = y_{i+}$ (see Section 2.2);
2. $q^2(y_*)$ is a column vector holding the number of times in which consecutive ones are obtained in the corresponding row of Q^1 .

⁶This choice, to our surprise, was found to outperform logit-based initialization.

For example, consider an individual for whom $\mathbf{y}_i = [0, 1, 1, 0]$; in this case we have $T_i = 4$, $y_+ = 2$ and $y_* = 1$. The corresponding Q matrix would be

$$Q = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix},$$

which lists all the possibilities. We call the “relevant” row k_i the one which contains the sequence \mathbf{y}_i actually observed (the third row of Q in this example, so $k_i = 3$).

The $Q^1(T, y_+)$ matrices can be computed recursively by noting that

$$Q^1(T, y_+) = \left[\begin{array}{c|c} Q^1(T-1, y_+) & \begin{array}{c} 0 \\ \vdots \\ 0 \end{array} \\ \hline Q^1(T-1, y_+ - 1) & \begin{array}{c} 1 \\ \vdots \\ 1 \end{array} \end{array} \right]$$

with the special cases

$$\begin{aligned} Q^1(n, 0) &= [0 \dots 0] \\ Q^1(n, 1) &= I_n \\ Q^1(n, n) &= [1 \dots 1]. \end{aligned}$$

Since none of the necessary Q_j depend on parameters to be estimated, they are pre-computed in a preliminary loop and stored in an array of matrices. The algorithm employed can be therefore given the following schematic description:

1. Initialize an empty array of matrices Q .
2. For each individual i :
 - (a) Compute the $j(i)$ index as a function of T_i , y_{i+} and y_{i*} .
 - (b) If Q_j has already been computed, stop and go to the next individual i ; else
 - i. Compute $Q^1(T_i, y_{i+})$ via the recursive method described above.
 - ii. Store the relevant row k_i for individual i .
 - iii. Compute Q_j via Equation 9.
 - iv. Store Q_j into the array at position j .

Once this is done, the likelihood for individual i in Equation 8 becomes a simple function of k_i and Q_j , which do not need to be recomputed during the Newton-Raphson iterations.

3.4. Identification of ρ and τ for RE probit models with AR(1) errors

When estimating RE probit models with autocorrelated errors, maximizing the log-likelihood may be quite challenging from a numerical point of view because of weak identification in some of the parameters. Consider first the ADP model, in which the covariance matrix of the compound disturbance terms for a unit is given in Equation 4. It is quite obvious that some combinations of the parameters give rise to a very badly conditioned matrix. This happens, in particular, when $\theta \simeq 1$ and $\rho = 1 - \epsilon$ (for “small” ϵ). In those cases, Σ will be very close to being singular, especially in those units for which the time dimension is short (note that all these three possibilities are far from being uncommon in practice). The typical outcome is BFGS taking many iterations to converge or, worse, not converging at all.

The situation is even worse for the GADP model, whose covariance matrix is given in Equation 5: First, since the covariance matrix depends on four parameters, an elementary order condition dictates that only the observations for which $T_i \geq 3$ will provide enough curvature to the log-likelihood to separately identify all four parameters, which is not an overly restrictive requirement. However, note that the parameter τ only appears in products like $\tau \cdot \rho^t$, with $1 \geq t \geq (T_i - 1)$. This implies that, for obtaining enough curvature in the objective function in the direction of τ , the sequence $\rho, \rho^2, \rho^3, \dots$ must be noticeably different (from a numerical point of view) from a constant sequence. In practice, this implies that estimation is likely to fail anytime in case (i) ρ is very close to 0 or to 1 and (ii) T_{\max} is small, unless N is truly enormous. In the limit, if $\rho = 0$ or $\rho = 1$ the model is under-identified even when $N \rightarrow \infty$ and $T \rightarrow \infty$.

4. The DPB function package

In **gretl**, function packages are collections of user defined functions made available to other users. The user can download and install the function packages available on the **gretl** server by menu or script. A function package can be downloaded and installed simply by invoking the `install` command:

```
Gretl console: type 'help' for a list of commands
```

```
? install DPB.zip
```

```
Installed DPB.zip
```

Then, in each work session, the function package needs to be loaded by:

```
? include DPB.gfn
```

We refer the reader to the **gretl** user guide for illustration on how to download and install function packages from the menu. The function package **DPB** includes five functions that handle model set-up, option management, estimation and printing of results. A summary of these functions is given in Table 1 and a detailed description is contained in the section “List of Public Functions” of the **DPB** documentation file. It is possible to call the public functions from the list window of function packages with the exception of `DPB_printout`.

Function	<code>DPB_setup(string mod, series depvar, list X, list Z[null])</code>
Return type	<code>bundle</code>
Function arguments	<code>string mod</code> "DP": Dynamic probit model (Heckman 1981b) "ADP": AR(1) dynamic probit model (Hyslop 1999) "GADP": Generalized AR(1) dynamic probit model (Keane and Sauer 2009) "QE": Quadratic exponential model (Bartolucci and Nigro 2010) <code>series depvar</code> : the binary dependent variable <code>list X</code> : list of the x_{it} in Equations 1 and 8 <code>list Z</code> : list of the z_{i1} in Equation 2 (optional)
Description	initialize the model bundle, sub-sample the data as needed, build data matrices, handle default settings
Function	<code>DPB_setoption(bundle *b, string opt, scalar value)</code>
Return type	<code>scalar</code>
Function arguments	<code>bundle *b</code> : pointer to the model bundle <code>string opt</code> : a string indicating which option to set <code>scalar value</code> : a scalar value for the option
Description	sets various options of the model (see Table 2); returns an error code, 0 if no error
Function	<code>DPB_estimate(bundle *b, matrix *par[null])</code>
Return type	<code>void</code>
Function arguments	<code>bundle *b</code> : pointer to the model bundle <code>matrix *par[null]</code> : column vector of <code>b.npar</code> elements holding initial values (optional)
Description	estimates the model
Function	<code>DPB_printout(bundle *b)</code>
Return type	<code>void</code>
Function arguments	<code>bundle *b</code> : pointer to the model bundle
Description	prints the estimation results
Function	<code>DPB_printape(bundle *b)</code>
Return type	<code>void</code>
Function arguments	<code>bundle *b</code> : pointer to the model bundle
Description	prints average partial effects (instead of estimation results) for the DP, ADP and GADP models; prints a warning message for the QE model.

Table 1: Summary of functions in **DPB**.

After estimation, this function is called automatically with a double click on the bundle icon in the icon window, where the model bundle is stored.

In the rest of this section, we first illustrate how to estimate the RE models DP, ADP and GADP described in Section 2.1, then we show how to perform estimation of the QE model proposed by Bartolucci and Nigro (2010) by means of code files written in **gretl**'s scripting language, Hansl (Cottrell and Lucchetti 2017).⁷ All the code used in this section as well as the data required are available in the supplementary material for replication accompanying the paper. The minimum version of **gretl** required to use the package is 2016a.

4.1. Random-effects dynamic probit

We start by describing how **DPB** handles the RE approach proposed by Heckman (1981b), by using an artificial dataset generated following Equations 1 and 3:

$$\begin{aligned} y_{it} &= 1\{1 + 0.6 y_{it-1} + 0.5 x_{it} + \alpha_i + \varepsilon_{it} \geq 0\}, \\ y_{i1} &= 1\{1 + x_{i1} + z_{i1} + 1.2 \alpha_i + \varepsilon_{i1} \geq 0\}, \end{aligned} \quad (10)$$

$$\sigma_\alpha^2 = 1, \quad \alpha_i \sim N(0, \sigma_\alpha^2), \quad x_{it} \sim N(0, 1), \quad z_{it} \sim U(0, 1),$$

for $i = 1, \dots, 4096$ and $t = 1, \dots, 6$. We also assume that $\rho = 0$, so ε_{it} is not autocorrelated. The supplementary material contains the code for generating the data. The code generates dynamic binary data under normality with the error structure in Equation 3, sets the parameter values in Equation 10 and stores the artificial data in `DP_artdata.gdtb`.

Once the **DPB** function package has been downloaded from the **gretl** server, a simple Hansl script to estimate the DP model is:

```
? set echo off
? set messages off
? include DPB.gfn
? open DP_artdata.gdtb
? list X = const x
? list Z = const x z
? b = DPB_setup("DP", y, X, Z)
? DPB_estimate(&b)
```

The first two lines of code just prevent **gretl** from echoing unnecessary output, while the `include` command loads the function package. Notice that here we are not setting any panel structure to the data, which is required for **DPB** to work, after they have been opened; this is because it has already been imposed on the sample datasets upon creation. If the data had not already had such structure, it should have been set by the user with the instruction:

```
? setobs id time --panel-vars
```

The lists of explanatory variables are then created. In the next line, the call to the public function `DPB_setup` initializes the model and returns a bundle: The first argument is a string

⁷“Hansl” is a recursive acronym for “Hansl’s A Neat Scripting Language”.

containing the name of the model to be estimated ("DP" in this case). The remaining arguments y , X , Z are the dependent variable, the list of explanatory variables for the primary equation and those for the initial condition equation.

The function `DPB_estimate` takes as its argument the pointer to the model bundle and fills the bundle with the estimated quantities. The bundle elements can be listed by simply typing `print b`. A detailed description of the bundle elements is given in the **DPB** documentation. Bundle elements can be accessed by the syntax `npar = b["npar"]` or `npar = b.npar`. The returned bundle can also be stored as an XML file by using the *gretl* built-in function `bwrite(b, "mod")` and reloaded, if necessary, by the companion function `b = bread("mod")`. The `DPB_estimate` function also assigns initial values for the parameters; the user can supply a vector of initial values as a second argument to `DPB_estimate` and override the default initialization if convergence problems should arise. For instance

```
? scalar npar = b.npar
? inipar = uniform(npar, 1)
? DPB_estimate(&b, &inipar)
```

To print out the results, the corresponding public function is called

```
DPB_printout(&b)
```

which produces the following output:

```
Dynamic Probit model
Dependent variable: y
Units: 4096 (observations: 24576)
Covariance matrix: Sandwich
Method: Gaussian quadrature with 24 quadrature points
```

	coefficient	std. error	z	p-value	
y(-1)	0.624525	0.0347020	18.00	2.06e-72	***

	coefficient	std. error	z	p-value	
const	0.501971	0.0388083	12.93	2.87e-38	***
x	0.521472	0.0143177	36.42	1.95e-290	***

	coefficient	std. error	z	p-value	
const	0.928349	0.0718382	12.92	3.35e-38	***
x	1.00589	0.0485473	20.72	2.30e-95	***
z	1.11562	0.120753	9.239	2.49e-20	***

theta	1.17864	0.0831465	14.18	1.30e-45	***
sigma	1.00819	0.0341844	29.49	3.59e-191	***
Log-likelihood	-10594.039	AIC	21204.078		
	BIC	21268.954	HQC	21225.093	

Wald test = 1326.52 (1 df); p-value = 1.95214e-290

where coefficients, standard errors and related z -tests are reported for the state dependence parameter γ , the coefficients of the main equation β , the coefficients of the initial conditions equation π , and the covariance matrix parameters θ and σ_α . At the end of the output table, a Wald test for the joint significance of the explanatory variables in the main equation is reported.

One of the reasons why RE models are often preferred to FE models by practitioners is the possibility of computing partial effects (PE), that have a meaningful economic interpretation in terms of probability variations. While the computation of PE is straightforward in models for cross-sectional data, additional sources of complication arise when dealing with panel data, mainly because the individual unobserved effect α_i enters the index function. Therefore, in addition to the estimation results, **DPB** provides the user with the possibility of printing out average PE by calling the public function `DPB_printape`, in the same manner as `DPB_printout`:

```
Dynamic Probit model: average partial effects
Dependent variable: y
Units: 4096 (observations: 24576)
Covariance matrix: Sandwich
Method: Gaussian quadrature with 24 quadrature points
```

	coefficient	std. error	z	p-value	
y(-1)	0.141518	0.00859182	16.47	5.91e-61	***

	coefficient	std. error	z	p-value	
x	0.109759	0.00268264	40.91	0.0000	***

where the average partial effects are computed following (Wooldridge 2010, p. 485) and the associated standard errors are derived via the Delta method.

As shown in the output headings, the model has been estimated using the Gauss-Hermite quadrature with 24 quadrature points, and the variance-covariance matrix of the parameters has been estimated by a sandwich formula. These are the default settings, which can be altered by the user by calling the public function `DPB_setoption` before `DPB_estimate`. For instance,

```
? err = DPB_setoption(&b, "nrep", 32)
? DPB_estimate(&b)
```

string opt	scalar value
"method"	0 = Gauss-Hermite quadrature (GHQ), default choice for the DP model, 1 = GHK algorithm. For the ADP and GADP models, <code>method</code> is forced to 1. For the QE model a warning message is printed.
"nrep"	Number of quadrature points or GHK draws. Default is 24 for the DP model with GHQ, 128 for the DP model with GHK and for the ADP and GADP models. For the QE model a warning message is printed.
"vcv"	Parameters covariance matrix. 0 = Sandwich (default), 1 = Outer product of the gradient (OPG), 2 = Hessian.
"verbose"	Degree of output verbosity. 0 = No output is printed, 1 = The log-likelihood at each iterations is printed (default), 2 = Log-likelihood, parameters and gradient at each iteration are printed.
"draws"	Type sequence for the GHK algorithm. 0 = Halton (default), 1 = Uniform with seed 31415927. For the DP model with GHQ and the QE model a warning message is printed.

Table 2: Description of `DPB_setoption`'s arguments.

`DPB_setoption` returns a scalar (called `err` in the above code lines) containing 0 if the option is valid and successfully set, an error code otherwise. For a detailed description of the available options, see Table 2.

In the example above, the string `"nrep"` and the scalar 32 are used to set the number of quadrature points to be used during estimation.

The DP model can also be estimated by computing the multivariate normal probabilities by GHK instead of using numerical integration by Gauss-Hermite quadrature. After the call to `DPB_setup`, the user can switch the method from GHQ (default) to GHK by calling `DPB_setoption` as follows

```
? err = DPB_setoption(&b, "nrep", 100)
? err = DPB_setoption(&b, "method", 1)
```

When the GHK method is invoked, a Halton sequence is used by default. The user may instead use random draws from a uniform distribution by setting `"draws"` to 1, as described in Table 2. The number of Halton points used by the GHK algorithm has a default value of 128 that can also be modified by the function `DPB_setoption` with the string `"nrep"` and the number of points, as illustrated earlier in this section.

In some cases, estimation can be computationally quite intensive. In particular, the GHK algorithm is quite sensitive to the time dimension of the panel, as T sets the dimension for the covariance matrix given in Equation 3. Table 3 reports the performance of the **DPB** function

	γ (se)	θ (se)	σ_α (se)	Log-lik	# iter.	time mm:ss
GHQ						
16	0.6246 (0.0350)	1.1787 (0.0807)	1.0083 (0.0339)	-10594.021	28	0:32
24	0.6245 (0.0350)	1.1786 (0.0807)	1.0082 (0.0339)	-10594.039	29	0:38
32	0.6245 (0.0350)	1.1786 (0.0807)	1.0082 (0.0339)	-10594.041	28	0:43
GHK						
128	0.6293 (0.0348)	1.1660 (0.0789)	1.0083 (0.0340)	-10593.130	24	2:29
192	0.6270 (0.0349)	1.1727 (0.0794)	1.0081 (0.0341)	-10593.399	25	3:42
256	0.6268 (0.0349)	1.1737 (0.0799)	1.0075 (0.0341)	-10593.538	23	4:17

Table 3: **DPB** performance with different numbers of quadrature points/GHK replications. Covariance matrix in Equation 3. Results obtained on a system with two Intel Pentium CPU G640 @ 2.80GHz processors.

package for different numbers of quadrature points and GHK replications to give the user an idea of the trade-off between algorithm quality and time. The table was obtained by the following command lines:

```
? b = DPB_setup("DP", y, X, Z)
? err1 = DPB_setoption(&b, "verbose", 2)
? err2 = DPB_setoption(&b, "vcv", 1)
? loop foreach i 16 24 32
> set stopwatch
> err_$i = DPB_setoption(&b, "nrep", $i)
> DPB_estimate(&b)
> DPB_printout(&b)
> t_$i = $stopwatch
> print t_$i
> endloop
```

The `verbose` string and the scalar 2 in `DPB_setoption` result in the printing of the values of parameters and gradients at each iteration in the log-likelihood maximization. In addition, we use the option `vcv` with value 1 to set the covariance matrix estimator to OPG, which is the least computationally demanding. Since the `method` option was left unmodified, the above code estimates the dynamic probit model by GHQ, with 16, 24, and 32 quadrature points. If, instead, one wishes to inspect the performance of the GHK algorithm, the method must be changed via `err = DPB_setoption(&b, "method", 1)` after the set-up function; in Table 3 we display the results obtained by setting the number of GHK draws to 128, 192, and 256. The estimates obtained via GHQ do not exhibit remarkable differences from each other. Conversely, the GHK algorithm obviously takes a longer time to yield the results because of its nature. In particular, the CPU-time increases in a roughly linear way with the number of replications.

As described in Section 2.1, the ADP and GADP models can be formulated by setting the variance-covariance matrix of the error terms as in Equations 4 and 5, respectively. Such a

	ADP	GADP
$y(-1)$	0.6063 (0.0762)	0.7017 (0.0784)
const	0.5030 (0.0552)	0.4364 (0.0596)
x	0.5256 (0.0172)	0.5162 (0.0158)
const	1.0603 (0.0706)	0.9425 (0.0752)
x	0.9879 (0.0500)	0.9798 (0.0555)
z	0.8485 (0.1150)	1.2036 (0.1252)
θ	1.0480 (0.0770)	1.1029 (0.0999)
σ	1.0399 (0.0370)	1.0137 (0.0387)
ρ	0.2877 (0.0526)	0.2360 (0.0538)
τ		0.6858 (0.1585)
Log-lik	-9982.842	-9919.944

Table 4: Estimation results: Simulated data. Standard errors (in parentheses) are computed using the sandwich variance-covariance matrix estimator. Number of GHK Halton draws: 128.

structure does not allow the use of GHQ to evaluate the relevant normal integral and GHK has to be used instead. In order to illustrate how **DPB** handles these extensions, we use the simulated dataset in Equation 10 with $\rho = 0.3, \tau = 1$ (stored in `ADP_artdata.gdtb`) and $\rho = 0.3, \tau = 0.6$ (stored in `GADP_artdata.gdtb`) to build Equations 4 and 5, respectively. The ADP model can be estimated by setting the first argument of `DPB_setup` to "ADP":

```
? open ADP_artdata.gdtb
? list X = const x
? list Z = const x z
? b = DPB_setup("ADP", y, X, Z)
? DPB_estimate(&b)
? DPB_printout(&b)
```

In this case, if the user tries to set GHQ as estimation method, a warning message is printed and the estimation proceeds with GHK using 128 Halton points.

Similarly, the GADP model can be estimated by setting the first value in the `DPB_setup` to "GADP":

```
? open GADP_artdata.gdtb
? list X = const x
? list Z = const x z
? b = DPB_setup("GADP", y, X, Z)
```

Running the two examples above produces the results shown in Table 4.

4.2. Quadratic exponential model

We exemplify the FE estimator proposed in [Bartolucci and Nigro \(2010\)](#), by means of an

artificial data set, generated in a similar way as in Equation 10:

$$\begin{aligned} y_{it} &= 1\{0.6 y_{it-1} + 0.5 x_{1it} + 0.5 x_{2it} + \alpha_i + \varepsilon_{it} \geq 0\}, \\ y_{i1} &= 1\{0.5 x_{1i1} + 0.5 x_{2i1} \alpha_i + \varepsilon_{i1} \geq 0\} \quad \text{for } i = 1, \dots, 1024, \quad t = 1, \dots, 6, \end{aligned}$$

where α_i is generated as in Equation 10, the regressors x_{1it} and x_{2it} are standard normal random variables and the error terms ε_{it} are logistically distributed with zero mean and variance $\pi^2/3$. The supplementary material also contains the `Hansl` code for generating the artificial data which is then stored in `QE_artdata.gdtb`. A simple script to estimate the quadratic exponential model described in Section 2.2 is:

```
? include DPB.gfn
? open QE_artdata.gdtb
? list X = x1 x2 x3
? b = DPB_setup("QE", y, X)
? DPB_setoption(&b, "vcv", 2)
? DPB_estimate(&b)
? DPB_printout(&b)
```

After the required panel structure is set and the list of explanatory variables has been created, the script calls the set-up function with the string "QE" as its first argument. For illustrative purposes, in the above example the covariance matrix estimator is set to the Hessian by setting the `vcv` option to 2. The script returns the following output:

```
Quadratic Exponential model
Dependent variable: y
Units: 837 (observations: 4185)
Total units: 1024 (total observations: 5120)
Covariance matrix: Hessian
```

	coefficient	std. error	z	p-value	
y(-1)	0.402312	0.0891894	4.511	6.46e-06	***

	coefficient	std. error	z	p-value	
x1	0.399268	0.0665217	6.002	1.95e-09	***
x2	0.513881	0.0494483	10.39	2.69e-25	***
x3	0.508360	0.0317102	16.03	7.71e-58	***

	coefficient	std. error	z	p-value	
const	0.232003	0.102851	2.256	0.0241	**
x1	0.258186	0.130109	1.984	0.0472	**
x2	0.580285	0.0964468	6.017	1.78e-09	***
x3	0.498261	0.0610953	8.155	3.48e-16	***

Log-likelihood	-1133.555	AIC	2283.110
BIC	2333.824	HQC	2301.045

Wald test = 491.74 (3 df); p-value = 2.94277e-106

Notice that, since only the units with $0 < y_{i+} < T$ are used, **DPB** reports in the output headings the number of actual contributions to the log-likelihood. The output reports coefficients, standard errors and z statistics for the parameters in Equation 8: the coefficient associated with the lagged dependent variable γ , the parameters β_1 for the explanatory variables and, finally, the parameters associated with the last observation, $[\mu, \beta_2^\top]^\top$.

The handling of time dummies in the model specification deserves a special mention. For instance, let us discuss the case of a balanced dataset: In a panel of T periods, the QE model identifies $T - 3$ time effects, as two dummies are dropped for the initial and rank condition in the main equation; another one gets dropped as the observation at time T is handled separately in the model specification (see Equation 8) and it includes an intercept term.

5. Examples: The union dataset

In this section, we illustrate the **DPB** function package by means of two empirical applications. In the first exercise, we replicate the example proposed by Stewart (2006), used to present the software components to estimate RE dynamic probit models.

In the second, we show how to implement in Hansl the estimator proposed by Wooldridge (2005), how to compute predicted probabilities, and compare it with the DP model available in **DPB**. As a side-product, we show how to implement the popular correlated random effects (CRE) approaches (Mundlak 1978; Chamberlain 1980).

Both examples use data extracted from the US National Longitudinal Survey of Youth on unionized workers, a very popular dataset often used as a benchmark for RE (dynamic) models for binary dependent variables with longitudinal data. The code and the dataset used for these examples are available in the supplementary material for replication.

5.1. Example 1

In the following, we replicate the empirical example provided in Stewart (2006), where the `union` dataset is used to illustrate the `Stata` software component for the estimation of the RE dynamic probit models of Heckman (1981b), DP, and Hyslop (1999), ADP.⁸ Differently from **DPB**, the software provided by Stewart (2006) unsurprisingly does not allow for the implementation of the estimator proposed by Keane and Sauer (2009). The results reported in this section can be replicated using the code and the data file `union_full.gdtb` available in the supplementary material.

Using the variables given in the `union` dataset, a model is specified for the binary dependent

⁸The data for this example were retrieved from the URL <http://www.stata-press.com/data/r9/union.dta>. As far as we are aware, usage of this dataset was unrestricted for academic use and we believe that a comparison between our results and Stewart's was indispensable.

variable at time t and for the initial condition as in Heckman (1981b):

$$\begin{aligned} \text{union}_{it} &= 1\{\gamma \text{union}_{it-1} + \beta_0 + \beta_1 \text{age} + \beta_2 \text{grade}_{it} + \beta_3 \text{south}_{it} + \alpha_i + \varepsilon_{it} \geq 0\}, \\ \text{union}_{i1} &= 1\{\pi_0 + \pi_1 \text{age}_{i1} + \pi_2 \text{grade}_{i1} + \pi_3 \text{south}_{i1} + \pi_4 \text{not_smsa} + \theta\alpha_i + \varepsilon_{i1} \geq 0\}, \end{aligned}$$

for $i = 1, \dots, n$ and $t = 2, \dots, T$, where `union` is the binary dependent variable, `age` is the age at time t , `grade` are the years of schooling and `south` is a dummy variable for living in the South. For the initial condition, `not_smsa`, living outside a standard metropolitan statistical area, is used as an exclusion restriction.

The dataset is an unbalanced panel which starts in 1970 and ends in 1988; a few years do not appear in the dataset, so several gaps are present. The number of units is 4434 and the maximum time length $T = 12$, for a total of 26200 observations. Simple descriptive statistics on the dataset are given by the summary command, followed by the variable names:

```
? summary union age grade south not_smsa --simple
```

	Mean	Median	S.D.	Min	Max
<code>union</code>	0.2218	0.000	0.4155	0.000	1.000
<code>age</code>	30.43	31.00	6.489	16.00	46.00
<code>grade</code>	12.76	12.00	2.412	0.000	18.00
<code>south</code>	0.4130	0.000	0.4924	0.000	1.000
<code>not_smsa</code>	0.2837	0.000	0.4508	0.000	1.000

In order to replicate the example provided in Stewart (2006), the dataset needs to be sub-sampled: Only the years from 1978 are kept and 1983 is dropped; in addition, the panel is balanced by keeping units that are present for six consecutive waves. The following code fragment sub-samples the dataset in order to keep the portion used in Stewart (2006) and stores it, for convenience, in the `gretl` data file `union.gdtb`:

```
? series valid = 0
? smpl year >= 78 --restrict
? smpl year != 83 --restrict
? matrix m = aggregate(const, idcode)
? series nwav = replace(idcode, m[,1], m[,2])
? smpl nwav == 6 --restrict
? series valid = 1
? store union.gdtb
? open union.gdtb --quiet
? smpl full
? smpl valid --dummy
? setobs idcode year --panel-vars
```

The resulting dataset is a balanced panel of 799 units observed for $T = 6$ periods, for a total of 3995 observations.

The models estimated in Stewart (2006) are a pooled probit, a RE probit (with no special treatment for the initial conditions), the DP and the ADP. Table 5 shows the estimation results of the pooled probit and of the RE probit. In the first case, the estimates of both the

	Pooled probit	RE probit (static)
union(-1)	1.8849 (0.0525)	1.1509 (0.1419)
const	-0.6986 (0.2474)	0.1785 (0.4182)
age	-0.0087 (0.0058)	-0.0240 (0.0086)
grade	-0.0145 (0.0103)	-0.0386 (0.0207)
south	-0.1685 (0.0519)	-0.3691 (0.1034)
ln(σ^2)		0.0909 (0.2923)
Log-lik	-1573.642	-1563.184

Table 5: Estimation results: Replication of [Stewart \(2006\)](#), Table 1, p. 265, for pooled and RE probit. Standard errors (in parentheses) are computed using the Hessian. Number of quadrature points: 24.

coefficients and standard errors are identical to those obtained by [Stewart \(2006\)](#), whereas the estimates of the RE probit (with 24 quadrature points) exhibit very small differences (the coefficient associated with union_{t-1} is 1.1507 in [Stewart 2006](#) compared to 1.1509, and so on).

The first column of [Table 6](#) reports the estimation results of the DP model, using 24 quadrature points. In this case, estimated coefficients, standard errors and the value of the log-likelihood at convergence are identical.⁹ The second and third columns of [Table 6](#) show the estimation results of the DP model using GHK instead of GHQ and of the ADP, both using 500 Halton points. While substantially similar, the results do present some discrepancies from those obtained by [Stewart \(2006\)](#). Such differences can probably be ascribed to the use of Halton sequences in **DPB** as opposed to random draws from the uniform distribution in the command written by [Stewart \(2006\)](#). Nevertheless, both sets of estimates show that ignoring the autocorrelation in the unobservables leads to the underestimation of the state dependence effect in these data. Finally, [Table 7](#) reports the estimation results for the GADP and the QE model. In the first case, the results are very similar to those obtained by estimating the ADP model, which is expected since the additional correlation coefficient τ is positive and close to unity. In the second, the values of the estimated coefficients are coherent with those of the RE models with no autocorrelation.

Since **DPB** handles unbalanced datasets, we repeat the exercise keeping all the usable observations. Starting from the full dataset at our disposal (`union_full.gdtb`), the `DPB_setup` function for RE models automatically sub-samples the dataset as described in [Section 3](#): Out of the 4434 units 3790 are kept, while 400 are dropped because they are observed only once and 244 because they are never observed for at least two consecutive periods. [Table 8](#) reports the estimation results based on this dataset and only for the models implemented in **DPB**. Retaining all usable units in the dataset considerably increases the estimate of the state dependence parameter in all models while reducing the estimated persistence due to time invariant unobserved heterogeneity.

⁹The reader should take into account that we directly estimate σ_α while [Stewart \(2006\)](#) estimates $\lambda = \frac{\sigma_\alpha^2}{\sigma_\alpha^2 + 1}$.

	DP	DP with GHK	ADP
union(-1)	0.6344 (0.0983)	0.6343 (0.1012)	1.3181 (0.1554)
const	0.5633 (0.4799)	0.5957 (0.4818)	0.0948 (0.4034)
age	-0.0286 (0.0092)	-0.0294 (0.0092)	-0.0237 (0.0081)
grade	-0.0539 (0.0269)	-0.0539 (0.0269)	-0.0369 (0.0201)
south	-0.4883 (0.1239)	-0.4953 (0.1247)	-0.3742 (0.0998)
const	-0.9597 (0.8414)	-0.9419 (0.8417)	-0.8710 (0.8496)
age	0.0081 (0.0238)	0.0077 (0.0238)	0.0100 (0.0243)
grade	-0.0064 (0.0341)	-0.0064 (0.0341)	-0.0132 (0.0334)
south	-0.7261 (0.1651)	-0.7310 (0.1650)	-0.7559 (0.1670)
not smsa	-0.4152 (0.1644)	-0.4151 (0.1644)	-0.4182 (0.1662)
θ	0.8641 (0.1095)	0.8622 (0.1106)	1.2250 (0.2143)
σ	1.5261 (0.1251)	1.5293 (0.1619)	1.0438 (0.1519)
ρ			-0.3372 (0.0554)
$\ln(\sigma^2)$			
Log-lik	-1860.215	-1860.275	-1854.618

Table 6: Estimation results: Replication of [Stewart \(2006\)](#), Table 1, p. 265, for DP, DP with GHK and ADP. Standard errors (in parentheses) are computed using the Hessian. Number of quadrature points: 24; number of GHK Halton draws: 500.

5.2. Example 2

In the following, we show how to implement in **gretl** the popular RE estimator for dynamic binary models proposed by [Wooldridge \(2005\)](#), CDP (*conditional DP*) henceforth. To this aim, we use the same dataset available in the data archive of the Journal of Applied Econometrics as well as in the **gretl** data archive, automatically included with the installation of **gretl**, and accessible from “File, Open data, Sample file, Gretl, union_wooldridge”.¹⁰ The data are a balanced panel of workers, extracted from the US Longitudinal Survey of Youth, which comprises 545 individuals observed for $T = 7$ periods, from 1981 to 1987.

The method proposed by [Wooldridge \(2005\)](#) relies on specifying a distribution for the individual unobserved heterogeneity conditional on the initial value of the dependent variable and the observed history of strictly exogenous explanatory variables. Following the paper, to which we refer the reader for details, the distribution of the individual unobserved effect is specified as

$$\alpha_i | y_{i1}, \mathbf{w}_i \sim N(\delta_0 + \delta_1 y_{i1} + \boldsymbol{\delta}_2^\top \mathbf{w}_i; s_\alpha^2),$$

where y_{i1} is the initial observation and \mathbf{w}_i contains the whole sequence of the strictly exogenous covariate w_{it} as in the approach of [Chamberlain \(1980\)](#). Notice that we use a different notation of the variance parameter as we will use σ_α^2 to denote the variance of the unconditional distribution.

¹⁰Like all sample datasets natively supplied with **gretl**, it can also be opened by issuing the command `open union_wooldridge`.

	GADP	QE
union(-1)	1.3006 (0.1618)	1.0638 (0.17060)
const	0.1164 (0.4101)	
age	-0.0241 (0.0082)	-0.0889 (0.0231)
grade	-0.0376 (0.0204)	-0.1907 (0.1811)
south	-0.3791 (0.1016)	-0.1187 (0.4895)
const	-0.8824 (0.8421)	0.5677 (0.9465)
age	0.0101 (0.0241)	-0.1140 (0.0516)
grade	-0.0121 (0.0332)	-0.1105 (0.1850)
south	-0.7486 (0.1659)	0.0266 (0.5355)
not smsa	-0.4149 (0.1648)	
θ	1.1813 (0.2256)	
σ	1.0635 (0.1619)	
ρ	-0.3419 (0.0567)	
τ	0.8499 (0.1877)	
$\ln(\sigma^2)$		
Log-lik	-1854.534	-467.495

Table 7: Estimation results: GADP and QE models. Standard errors (in parentheses) are computed using the Hessian. Number of GHK Halton draws: 500. For the last column, the coefficients related to the second block of covariates refers to the parameters for the last observation in Equation 7.

Estimation of the CDP probit

As is well known, the main virtue of the CDP model is that it can be easily estimated by means of standard routines used for the static RE probit with additional regressors. In Section 6 of [Wooldridge \(2005\)](#) the CDP model is estimated with union_{it} as the dependent variable, a dummy variable married_{it} and time dummies as explanatory variables. In addition, the whole history of married and the initial observation union_{i80} are included in the set of regressors to account for the conditional distribution of α_i . A second specification is also used, where the set of regressors contains also educ (years of schooling) and the black dummy variable. The practice of including lags and leads of a time-varying strictly exogenous variable is often adopted by practitioners as a preventive measure against possible correlation between the random effect and the covariates. For this reason, a static/dynamic model with this feature is often referred to as a CRE (correlated random effects) approach.

The Hansl script to replicate the results in Table I, p. 52 of [Wooldridge \(2005\)](#) is

```
? list TIME = dummify(year)
? TIME -= Dyear_2
? list MARR = marr8*
? list X = const union_1 union80 married MARR TIME
? probit union X --random-effects --quadpoints=12
? probit union X educ black --random-effects --quadpoints=12
```

	DP	ADP	GADP	QE
union(-1)	0.9183 (0.0472)	1.5168 (0.0637)	1.5129 (0.0648)	1.6640 (0.0736)
const	-1.7983 (0.1749)	-1.6159 (0.1402)	-1.6185 (0.1408)	
age	0.0029 (0.0033)	0.0005 (0.0028)	0.0005 (0.0028)	0.0143 (0.0060)
grade	0.0316 (0.0110)	0.0200 (0.0087)	0.0201 (0.0087)	-0.0802 (0.0577)
south	-0.4206 (0.0530)	-0.3215 (0.0434)	-0.3225 (0.0437)	-0.6180 (0.1665)
const	-1.8242 (0.2238)	-1.6905 (0.2159)	-1.6857 (0.2156)	-0.4120 (0.6163)
age	-0.0011 (0.0052)	-0.0052 (0.0051)	-0.0051 (0.0051)	0.0372 (0.0146)
grade	0.0659 (0.0136)	0.0682 (0.0130)	0.0678 (0.0131)	-0.0782 (0.0590)
south	-0.4674 (0.0683)	-0.4741 (0.0666)	-0.4722 (0.0666)	-0.6913 (0.1956)
not smsa	0.0031 (0.0691)	-0.0174 (0.0686)	-0.0170 (0.0684)	
θ	0.8405 (0.0575)	1.0856 (0.0908)	1.0714 (0.0996)	
σ	1.1735 (0.0535)	0.8305 (0.0508)	0.8348 (0.0524)	
ρ		-0.3207 (0.0246)	-0.3233 (0.0258)	
τ			0.9461 (0.0767)	
Log-lik	-7480.237	-7445.438	-7445.387	-2348.336

Table 8: Estimation results: DP, ADP, GADP and QE models, 3790 units. Standard errors (in parentheses) are computed using the Hessian. Number of quadrature points: 24; number of GHK Halton draws: 500.

where, after creating the list of explanatory variables, the two `probit` commands with the `-random-effects` flag estimate the CDP models. The default number of quadrature points has to be set to 12 to have an exact replication of Wooldridge’s original results.

For comparison purposes, we use the same specifications to estimate the DP model with the CRE approach. The estimation results are displayed in Table 9. The results in the first two columns of the table perfectly replicate the results in Wooldridge (2005). In addition, the same models estimated as DP models produce very similar results, as to be expected from the simulation study in Akay (2012).

The Mundlak version (Mundlak 1978) of the CRE approach is also very popular among practitioners: Instead of lags and leads, the regression is augmented by the within group mean of one or more time-varying strictly exogenous covariates. This estimator can easily be implemented in `gretl` by means of dedicated panel data functions, available to the user after setting the panel data structure. In the replication code file, the within group mean of the variable `married` is readily created in the line `m_marr = pmean(married)`. Using the baseline specification in Wooldridge (2005), we estimate the CDP and DP models with Mundlak’s CRE specification. The results are displayed in Table 10.

Computation of predicted probabilities

In Section 4.1, we illustrated how to have `DPB` report average PE and the associated standard errors by calling `DPB_printape` after estimation of the dynamic probit models. Here we also show how to compute predicted probabilities by `Hansl` scripting, that can be used to derive

	CDP (1)	CDP (2)	DP (1)	DP (2)
union(-1)	0.8747 (0.0944)	0.8857 (0.0942)	0.8866 (0.1090)	0.8885 (0.1128)
const	-1.8276 (0.1522)	-1.7124 (0.4494)	-1.4907 (0.1373)	-1.4921 (0.5034)
married	0.1677 (0.1111)	0.1691 (0.1110)	0.1686 (0.1120)	0.1703 (0.1126)
year82	0.0280 (0.1141)	0.0274 (0.1139)	0.0303 (0.1276)	0.0297 (0.1276)
year83	-0.0880 (0.1179)	-0.0893 (0.1178)	-0.0836 (0.1125)	-0.0869 (0.1126)
year84	-0.0484 (0.1196)	-0.0509 (0.1194)	-0.0399 (0.1196)	-0.0429 (0.1197)
year85	-0.2675 (0.1230)	-0.2681 (0.1228)	-0.2587 (0.1211)	-0.2591 (0.1215)
year86	-0.3190 (0.1250)	-0.3173 (0.1248)	-0.3085 (0.1212)	-0.3109 (0.1216)
year87	0.0738 (0.1194)	0.0727 (0.1193)	0.0789 (0.1293)	0.0772 (0.1296)
educ		-0.0169 (0.0361)		-0.0126 (0.0426)
black		0.5349 (0.1942)		0.7816 (0.2055)
marr81	0.0637 (0.2088)	0.0547 (0.2071)	0.1173 (0.1829)	0.1136 (0.2007)
marr82	-0.0707 (0.2556)	-0.0606 (0.2458)	-0.0752 (0.2467)	0.0216 (0.2908)
marr83	-0.1292 (0.2425)	-0.1363 (0.2423)	-0.0824 (0.2453)	-0.1403 (0.2956)
marr84	0.0251 (0.2651)	0.0698 (0.2678)	-0.0019 (0.2610)	-0.0330 (0.2463)
marr85	0.4070 (0.2459)	0.4282 (0.2445)	0.3621 (0.2235)	0.3367 (0.1801)
marr86	0.1089 (0.2626)	0.0789 (0.2628)	0.1915 (0.2489)	0.2394 (0.2112)
marr87	-0.4266 (0.2106)	-0.3878 (0.2157)	-0.5065 (0.1907)	-0.4085 (0.1657)
union80	1.5144 (0.1646)	1.4771 (0.1706)		
const			-0.9775 (0.1013)	-0.6868 (0.5808)
married			0.2279 (0.1907)	0.2462 (0.1887)
educ				-0.0291 (0.0486)
black				0.5401 (0.2363)
θ			0.7135 (0.0993)	0.6991 (0.1040)
σ_α			1.3181 (0.1083)	1.3097 (0.1307)
$\ln(s_\alpha)$	0.2435 (0.1812)	0.1885 (0.1788)		
Log-lik	-1287.475	-1283.390	-1594.371	-1587.094

Table 9: Estimation results: Replication of [Wooldridge \(2005\)](#), Table I, p. 52 and DP models, Chamberlain's CRE approach.

PE or average PE for arbitrary combinations of values of the model's explanatory variables. As an example, [Wooldridge \(2005\)](#) computes the estimated probability of being in a union in 1987 conditional on being or not being in a union in 1986 and on being married or not. After estimating the RE model and saving the relevant quantities, the predicted probabilities can easily be computed by means of a few lines of code. For instance, the probability of being in a union in 1987, conditional on being in a union in 1986 and being married, $P(\text{union}_{i1987} = 1 \mid \text{union}_{i1986} = 1, \text{married}_{it} = 1, \mathbf{x}_{it}, \mathbf{w}_i)$, can be written as

$$\Phi \left(\frac{\hat{\gamma} + \hat{\delta}_0 + \hat{\delta}_1 \text{union}_{i1980} + \hat{\beta} + \mathbf{w}_i^\top \hat{\delta}_2 + \hat{\beta}_{1987}}{\sqrt{1 + s_\alpha^2}} \right)$$

and is computed as

	CDP	DP
union(-1)	0.8875 (0.0925)	0.8988 (0.1068)
const	-1.8568 (0.1420)	-1.5462 (0.1362)
married	0.1698 (0.1104)	0.1677 (0.1109)
year82	0.0293 (0.1138)	0.0299 (0.1271)
year83	-0.0889 (0.1177)	-0.0865 (0.1117)
year84	-0.0494 (0.1192)	-0.0402 (0.1191)
year85	-0.2673 (0.1227)	-0.2579 (0.1206)
year86	-0.3182 (0.1246)	-0.3095 (0.1208)
year87	0.0717 (0.1192)	0.0762 (0.1289)
m_marr	0.0332 (0.1993)	0.0570 (0.2035)
union80	1.4776 (0.1630)	
const		-0.9569 (0.1005)
married		0.1956 (0.1805)
θ		0.6962 (0.0941)
σ_α		1.3058 (0.1040)
$\ln(s_\alpha)$	0.2295 (0.1694)	
Log-lik	-1291.377	-1598.478

Table 10: Estimation results: CDP and DP models, Mundlak's CRE approach.

```
? list X = const union_1 union80 married MARR TIME
? probit union X --random-effects --quadpoints=12 --quiet
? k = nelem(X)
? bw = $coeff[1 : k]
? s2a = exp($coeff[k + 1])
? series ndx = bw[1] + bw[2] + bw[3] * union80 + bw[4] + \
>   lincomb(MARR, bw[5 : 11]) + bw[17]
? series ndx /= sqrt(1 + s2a)
? wp11_m = cnorm(ndx)
```

After the model estimation command is invoked on the second line, we retrieve the estimated parameters by means of the accessor `$coeff`, which contains the parameter $\ln(s_\alpha)$ at the end. Then, we proceed to the computation of the index function, where `bw[1]` is $\hat{\delta}_0$, `bw[2]` is $\hat{\gamma}$, `bw[3]` is $\hat{\delta}_1$, `bw[4]` is the coefficient that multiplies the dummy `married`, `lincomb(MARR, bw[5 : 11])` is $\mathbf{w}_i^\top \hat{\boldsymbol{\delta}}_2$ and `bw[17]` is the coefficient for the 1987 time dummy. After the index function is normalized by $\sqrt{1 + \hat{s}_\alpha^2}$, the predicted probability is computed by the function `cnorm`, which returns the standard normal cdf. Similarly the estimated probabilities $P(\text{union}_{i1987} = 1 | \text{union}_{i1986} = 0, \text{married}_{it} = 1, \mathbf{x}_{it}, \mathbf{w}_i)$, and for `marriedit = 0`, can be computed by appropriately excluding `bw[2]` and `bw[4]` from the index function calculation.

The top panel of Table 11 reports the average estimated probabilities in the four cases and is the equivalent of Table II on page 52 in Wooldridge (2005). APEs can readily be derived by taking cell differences: For instance, the state dependence for married individuals is 0.182, and so on. The lower panel of Table 11 reports the results for the same exercise performed

Replication of Wooldridge (2005), Table II, p. 52		
	In union, 1986	Not in union, 1986
Married, 1987	0.4082	0.2256
Not married, 1987	0.3696	0.1970
DP		
	In union, 1986	Not in union, 1986
Married, 1987	0.4068	0.2211
Not married, 1987	0.3680	0.1922

Table 11: Probability of being in a union in 1987.

on the estimated coefficients by the Heckman's RE dynamic probit. In this case, the script file looks like

```
? list X = const married MARR TIME
? list Z = const married
? b = DPB_setup(1, union, X, Z)
? DPB_setoption(&b, "nrep", 12)
? DPB_setoption(&b, "verbose", 0)
? DPB_estimate(&b)
? k = b.nk
? z = b.nz
? bh = b.coeff[1 : k + 1]
? sig_a = b.coeff[k + z + 3]
? s2a = sig_a^2
? series ndx = bh[1] + bh[2] + bh[3] + lincomb(MARR, bh[4 : 10]) + bh[16]
? series ndx =/ sqrt(1 + s2a)
? series hp11_m = cnorm(ndx)
```

This time, after estimating the model, estimated coefficients need to be extracted from the bundle `b`, where they are stored into the vector `coeff`. The syntax `bh = b.coeff[1 : k + 1]` and `sig_a = b.coeff[k + z + 3]` extracts the needed estimates according to the order in which they are stored. The index function and the estimated probabilities are computed as in the previous exercise. The script file contains the code to replicate the full Table 11.

6. Conclusions

The aim of the **DPB** function package is to provide the practitioner with an intuitive and simple-to-use tool for the estimation of panel data dynamic binary choice models, whose adoption is often called for in applied microeconometrics.

Compared to existing pre-packaged software, we believe that **DPB** offers several improvements; the most important are, in our opinion, the ability to work with unbalanced or even

“gappy” panel datasets; a relatively broad variety in the available estimation techniques; an extremely efficient implementation of certain numerical techniques, so that estimation of realistically-sized models becomes feasible even on ordinary PCs.

Hopefully, **DPB** will allow practitioners, who may otherwise be discouraged by the complexity that arises from implementing these estimators in-house, to easily employ these estimators in standard research problems.

Acknowledgments

We are immensely grateful to two anonymous referees for their comments and suggestions, to Matteo Picchio for thoroughly testing the function package and for his many comments on the manuscript, and to Allin Cottrell for providing assistance with the **gretl** supporting infrastructure.

References

- Akay A (2012). “Finite-Sample Comparison of Alternative Methods for Estimating Dynamic Panel Data Models.” *Journal of Applied Econometrics*, **27**(7), 1189–1204. doi:10.1002/jae.1254.
- Albarrán P, Carrasco R, Carro J (2015). “Estimation of Dynamic Nonlinear Random Effects Models with Unbalanced Panels.” *Economics working papers*, Universidad Carlos III, Departamento de Economía.
- Alessie R, Hochguertel S, van Soest A (2004). “Ownership of Stocks and Mutual Funds: A Panel Data Analysis.” *The Review of Economics and Statistics*, **86**(3), 783–796. doi:10.1162/0034653041811761.
- Arulampalam W (2002). “State Dependence in Unemployment Incidence: Evidence for British Men Revisited.” *Technical report*, IZA Discussion Paper Series.
- Arulampalam W, Stewart MB (2009). “Simplified Implementation of the Heckman Estimator of the Dynamic Probit Model and a Comparison with Alternative Estimators.” *Oxford Bulletin of Economics and Statistics*, **71**(5), 659–681. doi:10.1111/j.1468-0084.2009.00554.x.
- Bartolucci F, Bellio R, Sartori N, Salvan A (2016). “Modified Profile Likelihood for Fixed-Effects Panel Data Models.” *Econometric Reviews*, **35**(7), 1271–1289. doi:10.1080/07474938.2014.975642.
- Bartolucci F, Nigro V (2010). “A Dynamic Model for Binary Panel Data with Unobserved Heterogeneity Admitting a \sqrt{n} -Consistent Conditional Estimator.” *Econometrica*, **78**(2), 719–733. doi:10.3982/ecta7531.
- Bartolucci F, Nigro V (2012). “Pseudo Conditional Maximum Likelihood Estimation of the Dynamic Logit Model for Binary Panel Data.” *Journal of Econometrics*, **170**(1), 102–116. doi:10.1016/j.jeconom.2012.03.004.

- Bartolucci F, Nigro V, Pignini C (2017). “Testing for State Dependence in Binary Panel Data with Individual Covariates.” *Econometric Reviews*. doi:10.1080/07474938.2015.1060039. Forthcoming.
- Bartolucci F, Pignini C (2017). “**cquad**: An R and Stata Package for Conditional Maximum Likelihood Estimation of Dynamic Binary Panel Data Models.” *Journal of Statistical Software*, **78**(7), 1–26. doi:10.18637/jss.v078.i07.
- Bettin G, Lucchetti R (2016). “Steady Streams and Sudden Bursts: Persistence Patterns in Remittance Decisions.” *Journal of Population Economics*, **29**(1), 263–292. doi:10.1007/s00148-015-0565-9.
- 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**(7), 1095–1116. doi:10.1002/jae.1081.
- Brown S, Ghosh P, Taylor K (2014). “The Existence and Persistence of Household Financial Hardship: A Bayesian Multivariate Dynamic Logit Framework.” *Journal of Banking & Finance*, **46**, 285–298. doi:10.1016/j.jbankfin.2014.05.022.
- Butler JS, Moffitt R (1982). “A Computationally Efficient Quadrature Procedure for the One-Factor Multinomial Probit Model.” *Econometrica*, **50**(3), 761–764. doi:10.2307/1912613.
- Byrd RH, Lu P, Nocedal J, Zhu C (1995). “A Limited Memory Algorithm for Bound Constrained Optimization.” *SIAM Journal on Scientific Computing*, **16**(5), 1190–1208. doi:10.1137/0916069.
- Cappellari L, Jenkins SP (2004). “Modelling Low Income Transitions.” *Journal of Applied Econometrics*, **19**(5), 593–610. doi:10.1002/jae.778.
- Carrasco R (2001). “Binary Choice with Binary Endogenous Regressors in Panel Data.” *Journal of Business & Economic Statistics*, **19**(4), 385–394. doi:10.1198/07350010152596637.
- Carro JM (2007). “Estimating Dynamic Panel Data Discrete Choice Models with Fixed Effects.” *Journal of Econometrics*, **140**(2), 503–528. doi:10.1016/j.jeconom.2006.07.023.
- Carro JM, Traferri A (2014). “State Dependence and Heterogeneity in Health Using a Bias-Corrected Fixed-Effects Estimator.” *Journal of Applied Econometrics*, **29**(2), 181–207. doi:10.1002/jae.2301.
- Chamberlain G (1980). “Analysis of Covariance with Qualitative Data.” *The Review of Economic Studies*, **47**(1), 225–238. doi:10.2307/2297110.
- Chamberlain G (1985). “Heterogeneity, Omitted Variable Bias, and Duration Dependence.” In JJ Heckman, B Singer (eds.), *Longitudinal Analysis of Labor Market Data*. Cambridge University Press, Cambridge.
- Contoyannis P, Jones AM, Rice N (2004). “The Dynamics of Health in the British Household Panel Survey.” *Journal of Applied Econometrics*, **19**(4), 473–503. doi:10.1002/jae.755.

- Cottrell A, Lucchetti R (2017). *A Hansl Primer*. URL <http://ricardo.ecn.wfu.edu/pub/gretl/manual/PDF/hansl-primer-a4.pdf>.
- Fernández-Val I (2009). “Fixed Effects Estimation of Structural Parameters and Marginal Effects in Panel Probit Models.” *Journal of Econometrics*, **150**(1), 71–85. doi:10.1016/j.jeconom.2009.02.007.
- Genz A (1992). “Numerical Computation of Multivariate Normal Probabilities.” *Journal of Computational and Graphical Statistics*, **1**(2), 141–149. doi:10.2307/1390838.
- Geweke J (1989). “Bayesian Inference in Econometric Models Using Monte Carlo Integration.” *Econometrica*, **57**(6), 1317–1339. doi:10.2307/1913710.
- Giarda E (2013). “Persistency of Financial Distress Amongst Italian Households: Evidence From Dynamic Models for Binary Panel Data.” *Journal of Banking & Finance*, **37**(9), 3425–3434. doi:10.1016/j.jbankfin.2013.05.005.
- gretl** Team (2016). *gretl 2016a: GNU Regression, Econometrics and Time-Series Library*. URL <http://gretl.sourceforge.net/>.
- Hahn J, Kuersteiner G (2011). “Bias Reduction for Dynamic Nonlinear Panel Models with Fixed Effects.” *Econometric Theory*, **27**(6), 1152–1191. doi:10.1017/s0266466611000028.
- Hahn J, Newey WK (2004). “Jackknife and Analytical Bias Reduction for Nonlinear Panel Models.” *Econometrica*, **72**(4), 1295–1319. doi:10.1111/j.1468-0262.2004.00533.x.
- Hajivassiliou VA, McFadden DL (1998). “The Method of Simulated Scores for the Estimation of LDV Models.” *Econometrica*, **66**(4), 863–896. doi:10.2307/2999576.
- Halliday TJ (2008). “Heterogeneity, State Dependence and Health.” *The Econometrics Journal*, **11**(3), 499–516. doi:10.1111/j.1368-423x.2008.00256.x.
- Halton JH (1964). “Algorithm 247: Radical-Inverse Quasi-Random Point Sequence.” *Communications of the ACM*, **7**(12), 701–702. doi:10.1145/355588.365104.
- Heckman JJ (1981a). “Heterogeneity and State Dependence.” In CF Manski, D McFadden (eds.), *Structural Analysis of Discrete Data with Econometric Applications*. MIT Press, Cambridge.
- Heckman JJ (1981b). “The Incidental Parameters Problem and the Problem of Initial Conditions in Estimating a Discrete Time-Discreted Data Stochastic Process.” In CF Manski, D McFadden (eds.), *Structural Analysis of Discrete Data with Econometric Applications*. MIT Press, Cambridge MA.
- Heckman JJ, Borjas GJ (1980). “Does Unemployment Cause Future Unemployment? Definitions, Questions and Answers from a Continuous Time Model of Heterogeneity and State Dependence.” *Economica*, **47**(187), 247–283. doi:10.2307/2553150.
- Heiss F (2011). “Dynamics of Self-Rated Health and Selective Mortality.” *Empirical Economics*, **40**(1), 119–140. doi:10.1007/s00181-010-0422-3.
- Honoré BE, Kyriazidou E (2000). “Panel Data Discrete Choice Models with Lagged Dependent Variables.” *Econometrica*, **68**(4), 839–874. doi:10.1111/1468-0262.00139.

- Hyslop DR (1999). “State Dependence, Serial Correlation and Heterogeneity in Intertemporal Labor Force Participation of Married Women.” *Econometrica*, **67**(6), 1255–1294. doi:[10.1111/1468-0262.00080](https://doi.org/10.1111/1468-0262.00080).
- Keane MP (1994). “A Computationally Practical Simulation Estimator for Panel Data.” *Econometrica*, **62**(1), 95–116. doi:[10.2307/2951477](https://doi.org/10.2307/2951477).
- Keane MP, Sauer RM (2009). “Classification Error in Dynamic Discrete Choice Models: Implications for Female Labor Supply Behavior.” *Econometrica*, **77**(3), 975–991. doi:[10.3982/ecta7642](https://doi.org/10.3982/ecta7642).
- Krailo MD, Pike MC (1984). “Algorithm AS 196: Conditional Multivariate Logistic Analysis of Stratified Case-Control Studies.” *Journal of the Royal Statistical Society C*, **33**(1), 95–103. doi:[10.2307/2347671](https://doi.org/10.2307/2347671).
- Miranda A (2007). “Dynamic Probit Models for Panel Data: A Comparison of Three Methods of Estimation.” In *United Kingdom Stata Users’ Group Meetings*, volume 11.
- Mundlak Y (1978). “On the Pooling of Time Series and Cross Section Data.” *Econometrica*, **46**(1), 69–85. doi:[10.2307/1913646](https://doi.org/10.2307/1913646).
- Orme CD (1997). “The Initial Conditions Problem and Two-Step Estimation in Discrete Panel Data Models.” Mimeo, University of Manchester.
- Orme CD (2001). “Two-Step Inference in Dynamic Non-Linear Panel Data Models.” Mimeo, University of Manchester.
- Pigini C, Presbitero AF, Zazzaro A (2014). “State Dependence in Access to Credit.” *Mo.Fi.R. Working Papers 102*, Money and Finance Research Group (Mo.Fi.R.) – Univ. Politecnica Marche – Dept. Economic and Social Sciences.
- Rabe-Hesketh S, Skrondal A, Pickles A (2002). “Reliable Estimation of Generalized Linear Mixed Models Using Adaptive Quadrature.” *The Stata Journal*, **2**(1), 1–21.
- R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
- Saito M, Matsumoto M (2008). “SIMD-Oriented Fast Mersenne Twister: a 128-Bit Pseudo-random Number Generator.” In A Keller, S Heinrich, H Niederreiter (eds.), *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pp. 607–622. Springer-Verlag, Berlin.
- StataCorp (2015). *Stata Data Analysis Statistical Software: Release 14*. StataCorp LP, College Station. URL <http://www.stata.com/>.
- Stewart M (2006). “Maximum Simulated Likelihood Estimation of Random-Effects Dynamic Probit Models with Autocorrelated Errors.” *The Stata Journal*, **6**(2), 256–272.
- Stewart MB (2007). “The Interrelated Dynamics of Unemployment and Low-Wage Employment.” *Journal of Applied Econometrics*, **22**(3), 511–531. doi:[10.1002/jae.922](https://doi.org/10.1002/jae.922).
- Wooldridge JM (2005). “Simple Solutions to the Initial Conditions Problem in Dynamic, Nonlinear Panel Data Models with Unobserved Heterogeneity.” *Journal of Applied Econometrics*, **20**(1), 39–54. doi:[10.1002/jae.770](https://doi.org/10.1002/jae.770).

Wooldridge JM (2010). *Econometric Analysis of Cross Section and Panel Data*. MIT Press.

Wunder C, Riphahn RT (2014). “The Dynamics of Welfare Entry and Exit Amongst Natives and Immigrants.” *Oxford Economic Papers*, **66**(2), 580–604. doi:10.1093/oep/gpt025.

Affiliation:

Riccardo Lucchetti, Claudia Pigni
Department of Economics and Social Sciences
Faculty of Economics
Università Politecnica delle Marche
60121 Ancona, Italy
E-mail: r.lucchetti@univpm.it, c.pigni@univpm.it
URL: <http://www.univpm.it/riccardo.lucchetti>