Theory and Practice of Total-Factor Productivity Estimation: The Control Function Approach using Stata

Published version:

DOI:10.1177/1536867x1801800307

Theory and Practice of TFP Estimation: the Control Function Approach Using Stata

Gabriele Rovigatti

University of Chicago Booth School of Business
Chicago, IL
gabriele.rovigatti@gmail.com

Vincenzo Mollisi

Free University of Bozen-Bolzano
Faculty of Economics and Management
Bolzano, Italy
vincenzo.mollisi@gmail.com

Abstract. Alongside Instrumental Variable (IV) and Fixed Effects (FE), the Control Function (CF) approach is the most widely used in production function estimation. Olley-Pakes (OP henceforth), Levinsohn-Petrin (LP), Ackerberg-Caves-Frazer (ACF) have all contributed to the field proposing two-steps estimation procedures, while Wooldridge showed how to perform a consistent estimation within a single step GMM framework. In this paper we propose a new estimator, based on Wooldridge's, using dynamic panel instruments à la Blundell-Bond and we evaluate its performance by Monte Carlo simulations. We also present a new Stata module - `prodest` - for production function estimation, show its main features and key strengths in a comparative analysis with other user-written Stata commands. Lastly, we provide evidence of the numerical challenges faced when using OP/LP estimators with ACF correction in empirical applications and document how the GMM estimates vary depending on the optimizer/starting points employed.

Keywords: , production functions, productivity, prodest, MrEst, dynamic panel GMM

I Introduction

The correct estimation of the total factor productivity is a fundamental issue in applied economics and is the main topic of several seminal papers. When subject to productivity shocks, firms respond by expanding their level of output and by demanding more input; negative shocks, on the other hand, lead to a decline in output and demand for input. The positive correlation between the observable input levels and the unobservable productivity shocks is a source of bias in OLS when estimating the total factor productivity. Various methods have been proposed to tackle such simultaneity issue and, according to their approaches, is possible to group them in three families: Fixed Effects (FE), Instrumental Variables (IV) and Control Function (CF). In the latter group, Olley and Pakes (1996) are the first to propose a two-step procedure aimed at overcoming the endogeneity: they use the investment level to proxy for productivity. Their approach has been refined by Levinsohn and Petrin (2003) and Ackerberg et al. (2015). Wooldridge (2009) proposes a novel estimation setting, showing how to obtain LP estimator within a system GMM econometric framework, which can be estimated in a single step, and shows the appropriate moment conditions.

All the mentioned models rely on a crucial assumption that underlies the dynamic profit maximization problem faced by the firm at each period t: the idiosyncratic shock to productivity at time t (i.e. ξ_t) does not affect the choice of the level of state variables, which is taken at $t-b$[1], but only that of free variables. Therefore, ξ_t is uncorrelated to the contemporaneous value of the state and to all the lagged values of the free and state variables and all these are valid instruments for parameter identification. However, adding lags to the system reduces sample dimension and decreases available information. In this paper we propose a modification to the Wooldridge estimator based on a matrix of dynamic panel instruments. Such an approach makes it possible to increase the moment restrictions without losing information, which is a highly desirable feature when dealing with "large N, small T" panel datasets that are so frequent in the related literature. We then show that *MrEst* performs better than Wooldridge's on simulated data with a small number of periods, increases the sample size in overidentified models and produces more stable results.

[1]where $b > 0$ can take different values depending on state variable dynamics.

We then introduce a new Stata module - `prodest` - that implements all the above listed methodologies. We present the command syntax, describe all options and briefly perform a comparison with existing user-written Stata modules. Eventually, we focus on ACF methodology and, using their data generating process (DGP), we show how such nonlinear problems' solutions are extremely dependent on the choice of the optimization starting points.

The remainder of the paper is structured as follows: in Section II we review all control function approaches, list their weaknesses and provide a general overview of the state of the art; in Section we introduce the new MrEst with a comprehensive presentation of its new characteristics; Section III introduces `prodest`, its main features and practical examples of usage; in Section IV we comparatively describe the module and present evidence on ACF dependence on starting points; Section V concludes.

II Control function approach

In this section we provide a brief but complete overview of the most common techniques for production function estimation using control function approach. For the remainder of the paper, consider a Cobb-Douglas technology for firm i at time t :

$$y_{it} = \alpha + \mathbf{w}_{it}\beta + \mathbf{x}_{it}\gamma + \omega_{it} + \varepsilon_{it} \tag{1}$$

where y_{it} is the log gross or the value added output, $\mathbf{w}_{it}$ is a $1 \times J$ vector of log free variables and $\mathbf{x}_{it}$ is a $1 \times K$ vector of log state variables. The random component ω_{it} is the unobservable productivity or technical efficiency and ε_{it} is an idiosyncratic output shock distributed as white noise. We assume with OP and LP that productivity evolves according to a first-order Markov process:

$$\omega_{it} = E(\omega_{it} \,|\Omega_{it-1}) + \xi_{it} = E(\omega_{it} \,|\omega_{it-1}) + \xi_{it} = g(\omega_{it-1}) + \xi_{it} \tag{2}$$

where Ω_{it-1} is the information set at $t-1$ and ξ_{it} is the productivity shock, assumed to be uncorrelated with productivity ω_t and with state variables $\mathbf{x}_{it}$.

II.1 Olley-Pakes method

OP were the first to propose a consistent two-step estimation procedure for (1). Their key idea is to exploit firm investment levels as a proxy variable for ω_{it}. They prove their estimates of productivity to be consistent under several assumptions on top of those mentioned above:

A.1 $i_{it} = f(\mathbf{x}_{it}, \omega_{it})$ is the investment policy function, invertible in ω_{it}. Moreover, i_{it} is monotonically increasing in ω_{it};

A.2 The state variables - typically capital - evolve according to the investment policy function i_{it} which is decided at time $t-1$;

A.3 The free variables $\mathbf{w}_{it}$ - typically labor inputs and/or intermediate materials - are non-dynamic, in the sense that their choice at t does not impact future profits, and are chosen at time t after the firm productivity shock realizes.

Hence, given A.1 and A.2, the investment i_{it} is orthogonal to the state variable in t such that $E[i_{it}|\mathbf{x}_{it}] = 0$ and can be inverted, yielding the following proxy for productivity:

$$\omega_{it} = f^{-1}(i_{it}, \mathbf{x}_{it}) = h(i_{it}, \mathbf{x}_{it}) \tag{3}$$

which is an unknown function of observable variables. Plugging (3) in (1), we obtain:

$$\begin{aligned} y_{it} &= \alpha + \mathbf{w}_{it}\beta + \mathbf{x}_{it}\gamma + h(i_{it}, \mathbf{x}_{it}) + \varepsilon_{it} = \\ &= \mathbf{w}_{it}\beta + \Phi_{it}(i_{it}, \mathbf{x}_{it}) + \varepsilon_{it} \end{aligned} \tag{4}$$

where we define $\Phi_{it}(i_{it}, \mathbf{x}_{it}) = \mathbf{x}_{it}\gamma + h(i_{it}, \mathbf{x}_{it}) = \mathbf{x}_{it}\gamma + \omega_{it}$. Equation (4) is a partially linear model identified only in the free variable vector, $\mathbf{w}_{it}$ and can be non parametrically estimated approximating $\Phi_{it}(i_{it}, \mathbf{x}_{it})$ by a n^{th} order polynomial $\hat{\Phi}$ or by a local linear regression (First Stage). This yields a consistent estimate of the free variables' parameters, $\hat{\beta}$. Using (2), then, it becomes possible to estimate γ by rewriting the model for $y_{it} - \mathbf{w}_{it}\hat{\beta}$ conditional on $\mathbf{x}_{it}$:

$$
\begin{aligned}
y_{it} - \mathbf{w}_{it}\hat{\beta} &= \alpha_0 + \mathbf{x}_{it}\gamma + \omega_{it} + \varepsilon_{it} = \\
&= \alpha_0 + \mathbf{x}_{it}\gamma + E[\omega_{it}|\,\omega_{it-1}] + \xi_{it} + \varepsilon_{it} = \\
&= \alpha_0 + \mathbf{x}_{it}\gamma + g(\omega_{it-1}) + e_{it}
\end{aligned} \tag{5}
$$

where $e_{it} = \xi_{it} + \varepsilon_{it}$. Being $\hat{\omega}_{it} = \hat{\Phi}_{it} - \mathbf{x}_{it}\gamma$ equation (5) becomes:

$$
y_{it} - \mathbf{w}_{it}\hat{\beta} = \alpha_0 + \mathbf{x}_{it}\gamma + g(\hat{\Phi}_{it-1} - \mathbf{x}_{it-1}\gamma) + e_{it} \tag{6}
$$

where the function $g(.)$ can be left unspecified and estimated non parametrically. Alternatively, if we assume $g(\cdot)$ to follow a random walk we can restate equation (6) as:

$$
y_{it} - \mathbf{w}_{it}\hat{\beta} = \alpha_0 + (\mathbf{x}_{it} - \mathbf{x}_{it-1})\,\gamma + \hat{\Phi}_{it-1} + e_{it} \tag{7}
$$

and

$$
e_{it} = y_{it} - \mathbf{w}_{it}\hat{\beta} - \alpha_0 - \mathbf{x}_{it}\gamma^{*} - g(\hat{\Phi}_{it-1} - \mathbf{x}_{it-1}\gamma^{*}) \tag{8}
$$

at the true γ^{*} value.

Equation (7) suggests an immediate approach to the estimation. In fact, residuals e_{it} can be used to build a GMM estimator exploiting the moment conditions $E[e_{it}x_{it}^{k}]$=0, $\forall k$ (Second Stage)[2], where x^{k} are the single elements of vector $mathbfx$. The $\gamma*$ vector is the vector of

[2]Alternatively, estimation of second stage can be carried out on Eq. (7) using non lineal least squares since e_{it} is a combination of pure errors.

parameters which minimizes the criterion function:

$$\gamma* = \text{argmax}\left\{-\sum_k\left(\sum_i\sum_t e_{it}x_{it}^k\right)^2\right\} \tag{9}$$

In their seminal paper OP discuss potential selection bias due to the non-randomness in plants dropping out the sample. More specifically, less productive firms could be forced out of the market exactly due to their low level of productivity, thus leaving only the most productive firms in the sample. They assume that a firm continues to operate provided that its productivity level exceeds the lower bound, i.e. $\chi_{it} = 1 \iff \omega_{it} \geq \underline{\omega}_{it}$, where χ_{it} is a survival binary variable and the $\underline{\omega}_{it}$ is an industry-specific exit-triggering threshold (see Hopenayn (1992) and Melitz (2003)). Hence, they propose a third step in estimation in order to account for that: model (6) is expressed conditionally not only on the state variable, but also on χ_{it} - i.e. productivity is a function of its past values and of the survival indicator variable:

$$y_{it} - \mathbf{w}_{it}\hat{\beta} = \alpha_0 + \mathbf{x}_{it}\gamma + E[\omega_{it}|\,\omega_{it-1},\,\chi_{it}] + e_{it} \tag{10}$$

The bias correction proposed by OP consists in adding to (7) an estimate of the conditional probability of remaining active in the market, i.e. $\hat{Pr}_{it+1} \equiv Pr\left\{\chi_{it+1} = 1|\mathbf{x}_{it}\right\}$. Thus:

$$y_{it} - \mathbf{w}_{it}\hat{\beta} = \alpha_0 + \mathbf{x}_{it}\gamma + g(\hat{\Phi}_{it-1} - \mathbf{x}_{it-1}\gamma,\,\hat{Pr}_{it}) + e_{it} \tag{11}$$

where $\hat{Pr}_{it}$ is the fitted surviving probability - typically estimated through a discrete choice model on a polynomial of the state variable vector $\mathbf{x}_{it}$ and the investment.

II.2 Levinsohn-Petrin method

OP approach has a major drawback in empirical applications which limits its range of applications: real firm- or plant-level data have many zeros in investment preventing, in practice, the estimation. This is due to common industrial practices which violate the monotonicity assumption A.1: investments are not decided at each point in time, but postponed for few years before being made all at once. LP propose to overcome this issue by exploiting intermediate input levels as a proxy variable for ω_{it}. As in the OP case, LP methodology is based on assumptions:

B.1 Firms observe their productivity shock and adjust their optimal level of intermediate inputs - materials - according to the demand function $m(\omega_{it}, \mathbf{x}_{it})$;

B.2 $m_{it} = f(\mathbf{x}_{it}, \omega_{it})$ is the intermediate input function, invertible in ω_{it}. Moreover, m_{it} is monotonically increasing in ω_{it};

B.3 The state variables - typically capital - evolve according to the investment policy function $i()$ which is decided at time $t-1$;

B.4 The free variables $\mathbf{w}_{it}$ - typically labor inputs and/or intermediate materials - are non-dynamic, in the sense that their choice at t does not impact future profits, and are chosen in t after the firm productivity shock realizes.

Under the set of assumptions B.1-B.4, intermediate input demand is orthogonal to the set of state variables in t such that $E[m_{it}|\mathbf{x}_{it}] = 0$ and m_{it} can be inverted, yielding the following technical efficiency proxy:

$$\omega_{it} = h(m_{it}, \mathbf{x}_{it}) \tag{12}$$

which is an unknown function of observable variables. Plugging (12) in (1) and distinguishing the intermediate input variable from the free variables we obtain:

$$\begin{aligned} y_{it} = \alpha + \mathbf{w}_{it}\beta + \mathbf{x}_{it}\gamma + \delta m_{it} + h(m_{it}, \mathbf{x}_{it}) + e_{it} = \\ = \mathbf{w}_{it}\beta + \Phi_{it}(m_{it}, \mathbf{x}_{it}) + e_{it} \end{aligned} \tag{13}$$

where $e_{it} = \xi_{it} + \varepsilon_{it}$.

Equation (13) is a partially linear model identified only in the free variable vector but not in the proxy variable, m_{it}. Similar to OP, equation (12) can be non-parametrically estimated approximating $\Phi_{it}(m_{it}, \mathbf{x}_{it})$ by a n^{th} order polynomial or by local linear regression (First Stage). At the true values [$\gamma*$, δ*] we can define the residual function e_{it} like:

$$e_{it} = y_{it} - \mathbf{w}_{it}\hat{\beta} - \mathbf{x}_{it}\gamma^* - \delta^* m_{it} - g\left(\hat{\Phi}_{it-1} - \mathbf{x}_{it-1}\gamma^* - \delta^* m_{it}\right) \tag{14}$$

However, e_{it} is no longer a combination of pure errors. The intermediate input variable is correlated with the error term given firms' response to the technology efficiency shock ξ_{it}. Thus, non-linear least squares would provide inconsistent estimates and relying on a GMM estimator is mandatory. The GMM estimator might be constructed by exploiting the residuals e_{it} and the set of moment conditions $E[e_{it}z_{it}^k]$=0, $\forall k$, where k is the index of the instrument vector $\mathbf{z} = [\mathbf{x}_{it},\ m_{it-1}]$

$$\begin{bmatrix} \gamma^* \\ \delta^* \end{bmatrix} = \text{argmax}\left\{ -\sum_k \left(\sum_i \sum_t e_{it} z_{it}^k \right)^2 \right\} \tag{15}$$

consistently estimates the set of paramenters $[\gamma,\ \delta]^\intercal$.

II.3 Ackerberg, Caves and Frazer correction

Both OP and LP assume that firms are able to instantly adjust some inputs at no cost when subject to productivity shocks. However, ACF and Bond and Soderbom (2005) remark that the labor coefficient can be consistently estimated in the first stage only if the free variables

show variability independently of the proxy variable. If this is not the case, their coefficients would be perfectly collinear in the first-stage estimation and hence not identifiable.
In particular, in the LP setting labor and intermediate inputs are assumed to be allocated simultaneously at time t. This implies that labor and materials are both chosen as a function of productivity and state variables $\mathbf{x}_{it}$:

$$\begin{aligned} m_{it} &= m(\omega_{it}, \mathbf{x}_{it}) \\ l_{it} &= l\,(\omega_{it}, \mathbf{x}_{it}) \end{aligned} \tag{16}$$

Using the monotonicity condition (B.2) ACF provide the following results:

$$l_{it} = l\,[h(m_{it}, \mathbf{x}_{it}), \mathbf{x}_{it}] \tag{17}$$

Hence, a collinearity issue arises in estimating the first stage, where the labor appears both as a free variable and in the non-parametric polynomial approximation $\hat{\Phi}_{it}$. In the same fashion the collinearity issue affects the OP estimator. ACF propose an alternative approach based on the following assumptions:

C.1 $p_{it} = p_{it}(\mathbf{x}_{it}, \,, l_{it}, \omega_{it})$ is the proxy variable policy function, invertible in ω_{it}. Moreover, p_{it} is monotonically increasing in ω_{it};

C.2 The state variables are decided at time $t - b$;

C.3 The labor input, l_{it}, is chosen at time $t - \zeta$, where $0 < \zeta < b$. The free variables, $\mathbf{w}_{it}$, are chosen at time t when the firm productivity shock is realized.

C.4 The production function is value added in the sense that the intermediate input m_{it} does not enter the production function to be estimated.

Assumption C.4 is needed because Bond and Soderbom (2005) have shown that under the scalar unobservable assumptions of ACF, a gross output production function is not identified without imposing further restrictions of the model, see paragraph 4.1 of (Ackerberg et al.

2015) for the details. Under the set of assumptions C.1-C.3 the first stage estimation is meant to remove the shock ε_{it} from the the output y_{it}. In particular the policy function can be inverted and plugged in equation (1) yielding:

$$y_{it} = \Phi_{it}(p_{it}, \mathbf{x}_{it}, \mathbf{w}_{it}, l_{it}) + \varepsilon_{it} \tag{18}$$

where $\Phi_{it}(p_{it}, \mathbf{x}_{it}, \mathbf{w}_{it}, l_{it}) = \mathbf{x}_{it}\gamma + \mathbf{w}_{it}\beta + \mu l_{it} + h(p_{it}, \mathbf{x}_{it}, \mathbf{w}_{it}, l_{it})$. Once $\hat{\Phi}_{it}$ is recovered, for any candidate vector $(\gamma^*, \beta^*, \mu^*)$, it is possible to obtain the residuals

$$\hat{\omega}_{it} = \hat{\Phi}_{it} - \mathbf{x}_{it}\gamma - \mathbf{w}_{it}\beta - \mu l_{it} \tag{19}$$

and, exploiting the Markov chain assumption $\omega_{it} = E(\omega_{it} | \omega_{it-1}) + \xi_{it} = g(\omega_{it-1}) + \xi_{it}$, obtain the residuals ξ_{it}. These, combined with the set of moment conditions $E[\xi_{it} z_{it}^k]$=0, $\forall k$, where k is the index of the instrument vector $\mathbf{z} = [\mathbf{x}_{it}, m_{it-1}, l_{it-1}]$, lead to the GMM criterion function (Second stage):

$$\begin{bmatrix} \gamma^* \\ \beta^* \\ \delta^* \end{bmatrix} = \text{argmax} \left\{ -\sum_k \left(\sum_i \sum_t \xi_{it} z_{it}^k \right)^2 \right\} \tag{20}$$

II.4 Wooldridge

Wooldridge (2009) proposes to address the OP/LP problems by replacing the two-step estimation procedure with a generalized method of moments (GMM) setup as in Wooldridge (1996). In particular, he shows how to write the relevant moment restrictions in terms of two equations: these have the same dependent variable (y_{it}) but are characterized by a different set of instruments. This approach has useful features with respect to previously proposed estimation routines:

- it overcomes the potential identification issue highlighted by ACF in the first stage;

- robust standard errors are easily obtained, accounting for both serial correlation and/or heteroskedasticity[3].

In the first stage by OP/LP, the estimation of the parameters is addressed under the assumption that

$$E(\varepsilon_{it}|\omega_{it-1}, \mathbf{w}_{it}, \mathbf{x}_{it}, m_{it}, \mathbf{w}_{it-1}, \mathbf{x}_{it-1}, m_{it-1}, ..., \mathbf{w}_{i1}, \mathbf{x}_{i1}, m_{i1}) = 0 \tag{21}$$

without imposing any functional form on the control function $\omega_{it} = h(.,.)$. The second stage assumption exploits the Markovian nature of productivity and the assumed orthogonality between productivity shocks and current values of the state variables, as well as between productivity shocks and past realizations of the free variables and the intermediate inputs. Following LP and rewriting Eq. (2) it states:

$$E(\omega_{it}|\mathbf{x}_{it}, \mathbf{w}_{it-1}, \mathbf{x}_{it-1}, m_{it-1}, ..., \mathbf{w}_{i1}, \mathbf{x}_{i1}, m_{i1}) = E(\omega_{it}\,|\omega_{it-1}) = f[h(\mathbf{x}_{it-1}, m_{it-1})] \tag{22}$$

where, as for $h(.,.)$, no functional form is imposed on $f(.)$. Assumptions (21) and (22) directly lead to the formulation of the two following equations:

$$y_{it} = \alpha + \mathbf{w}_{it}\beta + \mathbf{x}_{it}\gamma + h(\mathbf{x}_{it}, m_{it}) + v_{it} \tag{23}$$

$$y_{it} = \alpha + \mathbf{w}_{it}\beta + \mathbf{x}_{it}\gamma + f[h(\mathbf{x}_{it-1}, m_{it-1})] + \eta_{it} \tag{24}$$

where $\eta_{it} = \xi_{it} + v_{it}$.

In the estimation the approach is to deal with the unknown functional forms using n^{th} order polynomials in $\mathbf{x}_{it}$ and m_{it}[4], where the limiting case with $\mathbf{x}_{it}$ and m_{it} (i.e. $n = 1$)

[3] LP and OP recommend instead to bootstrap the standard errors of their estimators, as usual in two-step estimation procedures.

[4] LP suggest to use third-degree polynomials. However, the higher the degree the better the result.

entering linearly should always be allowed. In particular, if we assume that

$$h(\mathbf{x}_{it},\, \mathbf{m}_{it}) = \lambda_0 + \mathbf{k}(\mathbf{x}_{it}, m_{it})\lambda \tag{25}$$

where $k(.,.)$ is a $1 \times Q$ collection of functions.

$$f(h) = \delta_0 + \delta_1 h + \delta_2 h^2 + ... + \delta_G h^G \tag{26}$$

it implies $f(\omega_{it}) = \delta_0 + \delta_1[\mathbf{k}(\mathbf{x_{it-1}}, m_{it-1})\lambda_1] + \delta_2[\mathbf{k}(\mathbf{x}_{it-1}, m_{it-1})\lambda_1]^2 + ... + \delta_G[\mathbf{k}(\mathbf{x}_{it-1}, m_{it-1})\lambda_1]^G$.

For sake of simplicity, consider the case with $G = 1$ and $\delta_1 = 1$[5]: a simple substitution in Eqs. (23)-(24) yields

$$y_{it} = \zeta + \mathbf{w}_{it}\beta + \mathbf{x}_{it}\gamma + \mathbf{k}(\mathbf{x}_{it}, m_{it})\lambda_1 + v_{it} \tag{27}$$

$$y_{it} = \theta + \mathbf{w}_{it}\beta + \mathbf{x}_{it}\gamma + \mathbf{k}(\mathbf{x}_{it-1}, m_{it-1})\lambda_1 + \eta_{it} \tag{28}$$

where ζ and θ are the new constant parameters obtained through aggregation of all the constant terms. Under the assumptions of $G = 1$ and $\delta_1 = 1$, the system GMM has linear moments. The choice of instruments for both Eq. (27) and (28) is straightforward and reflects the orthogonality conditions listed above: in particular, we define $\mathbf{z}_{it1} = (1, \mathbf{x}_{it}, \mathbf{w}_{it}, \mathbf{k}(\mathbf{x}_{it}, m_{it}))$, $\mathbf{z}_{it2} = (1, \mathbf{x}_{it}, \mathbf{w}_{it-1}, \mathbf{k}(\mathbf{x}_{it-1}, m_{it-1}))$ and $\mathbf{Z}_{it} = \begin{pmatrix} \mathbf{z}_{it1} \\ \mathbf{z}_{it2} \end{pmatrix}$.

For each $t > 1$ the usual GMM with IV setup applies and the moment conditions are derived from the residual functions

$$\mathbf{r}_{it}(\theta) = \begin{pmatrix} r_{it1}(\theta) \\ r_{it2}(\theta) \end{pmatrix} = \begin{pmatrix} y_{it} - \zeta - \mathbf{w}_{it}\beta - \mathbf{x}_{it}\gamma - \mathbf{k}(\mathbf{x}_{it}, m_{it})\lambda_1 \\ y_{it} - \theta - \mathbf{w}_{it}\beta - \mathbf{x}_{it}\gamma - \mathbf{k}(\mathbf{x}_{it-1}, m_{it-1})\lambda_1 \end{pmatrix} \tag{29}$$

[5] This is the case whose estimation is implemented in `prodest`.

and $E[\mathbf{Z}'_{it}\mathbf{r}_{it}(\theta)] = 0$.

In this leading case the estimation is particularly straightforward, as the whole system boils down to a linear estimation problem. Following Wooldridge (2009), we can rewrite the system as $\mathbf{y}_{it} = \mathbf{X}_{it}\theta + \mathbf{r}_{it}$, where $\mathbf{y}_{it}$ is a vector containing y_{it} twice (stacked), θ is the vector of parameters of interest, $\mathbf{r}_{it}$ is defined as above and

$$\mathbf{X}_{it} = \begin{pmatrix} 1 & 0 & \mathbf{w}_{it} & \mathbf{x}_{it} & \mathbf{k}(\mathbf{x}_{it}, m_{it}) \\ 0 & 1 & \mathbf{w}_{it} & \mathbf{x}_{it} & \mathbf{k}(\mathbf{x}_{it-1}, m_{it-1}) \end{pmatrix} \tag{30}$$

Using $\mathbf{Z}_{it}$ as above yields consistent estimates.

II.5 IV estimation of ACF: the Robinson estimator

The Wooldridge (2009) estimator implemented in prodest collapse into the Robinson (1988) semiparametric estimator for the ACF case. If we agree with the input timing in ACF, the first equation would be unable to identify any of the parameters. However, identification is achieved by estimating semi-parametrically eq. 28 only.

Following Wooldridge (2009), equation (24) requires the orthogonality condition

$$E(\eta_{it}|\mathbf{x}_{it}, \mathbf{w}_{it-1}, \mathbf{x}_{it-1}, m_{it-1}, ..., \mathbf{w}_{i1}, \mathbf{x}_{i1}, m_{i1}) = 0, t = 2, .., T \tag{31}$$

to be consistently estimated. Provided that, then, within the ACF framework is possible to estimate β and γ by a instrumental variable version of Robinson (1988)'s estimator,[6] with $\mathbf{x}_{it}, \mathbf{x}_{it-1}$ and m_{it} used as included instruments and $\mathbf{w}_{it-1}$ instrumenting the endogenous $\mathbf{w}_{it}$.

[6] In the same ACF setting, equation (23) does not identify β even under the orthogonality condition in (21).

II.6 MrEst: Introducing dynamic panel instruments

As Wooldridge suggests previous lags are valid instruments in the above GMM estimation framework, but using them can be costly in terms of sample size as each additional lag implies the loss of n observations during the estimation. Most datasets in the literature have a relatively modest number of observations per panel, hence this may be problematic; in particular, it could be detrimental in combination with the use of investments as proxy variable (OP) which already leads to a reduced sample in the estimation.

In order to tackle this issue we propose to use dynamic panel data instruments à la Blundell and Bond (1998) within the Wooldridge framework outlined above.

As before, for each $t > 1$ define a $2 \times (T-1)$ residual function matrix as:

$$\mathbf{r}_i(\theta) = \begin{pmatrix} y_{i2} - \zeta - \mathbf{w}_{i2}\beta - \mathbf{x}_{i2}\gamma - k(\mathbf{x}_{i2}, \mathbf{m}_{i2})\lambda_1 \\ y_{i2} - \theta - \mathbf{w}_{i2}\beta - \mathbf{x}_{i1}\gamma - k(\mathbf{x}_{i1}, \mathbf{m}_{i1})\lambda_1 \\ \cdots \\ \cdots \\ y_{iT} - \zeta - \mathbf{w}_{iT}\beta - \mathbf{x}_{iT}\gamma - k(\mathbf{x}_{iT}, \mathbf{m}_{iT})\lambda_1 \\ y_{iT} - \theta - \mathbf{w}_{iT}\beta - \mathbf{x}_{iT}\gamma - k(\mathbf{x}_{iT-1}, \mathbf{m}_{iT-1})\lambda_1 \end{pmatrix} \tag{32}$$

For each panel i we define $t - b$ the last available lag (i.e. when $b = 1$ at $t = 2$, $b = 2$ at $t = 3$ and $b = T - 1$ at $t = T$). Then, let $\mathbf{Z}_i$ denote the dynamic panel instrument matrix for each panel (we suppress the subscript i to avoid an abuse of notation):

$$\mathbf{Z} = \begin{bmatrix} \mathbf{z}_2' & \mathbf{z}_3' & \cdots & \mathbf{z}_T' & 0 & 0 & 0 & 0 \\ 0 & 0 & \cdots & 0 & \tilde{\mathbf{z}}_3' & 0 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & \tilde{\mathbf{z}}_4' & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \tilde{\mathbf{z}}_T' \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{33}$$

where component $\tilde{\mathbf{z}}_t$ is a vector of dimension $1 \times b$ consisting of $\mathbf{z}_{t-1}, \ldots, \mathbf{z}_{t-b}$.

As usual, GMM moment conditions are defined as:

$$E\left[\mathbf{Z}_i \mathbf{r}_i(\theta)\right] = 0 \tag{34}$$

Using dynamic panel data instruments in a setting *à la Wooldridge* strengthens the robustness and the efficiency of estimates. Indeed, using (34) allows to maximize the number of restrictions and enhances parameter identification: in section IV we report the results of several Monte Carlo simulations to show how MrEst performs better than Wooldridge in many applications due to its increased precision.

III Production Function Estimation Using Stata

III.1 Syntax

Prodest

`prodest` *depvar* `[if` *exp*`] [in` *range*`] , free(`*varlist*`) proxy(`*varlist*`) state(`*varlist*`) method(`*name*`) [valueadded control(`*varlist*`) acf id(`*varname*`) t(`*varname*`) reps(#) level(#) poly(#) seed(#) fsresidual(`*newname*`) attrition endogenous(`*varlist*`) opt_options translog overidentification gmm]`

Predict

`predict` *newvarname* `[if` *exp*`], [residuals exponential parameters omega]`

III.2 Options

Prodest

- `free(`*varlist*`)` free variable(s). Ln(labour) in OP, LP and ACF.
- `state(`*varlist*`)` state variable(s). Ln(capital) in OP, LP and ACF.
- `proxy(`*varlist*`)` proxy variable(s). Ln(investment) in OP, ln(intermediate inputs) in LP and ACF.
- `control(`*varlist*`)` control variable(s) to be included
- `endogenous(`*varlist*`)` endogenous variable(s) to be included
- `acf` applies the Ackerberg et al. (2015) correction
- `valueadded` indicates that *depvar* is output value added. Default is gross output
- `attrition` correct for attrition - i.e. firm exit - in the data
- `method` methodology to be used: *op* (Olley-Pakes), *lp* (Levinsohn-Petrin), *wrdg* (Wooldridge), *rob* (Wooldridge/Robinson) or *mr* (Mollisi-Rovigatti)
- `id(`*varname*`)` specifies the *panelvar* to which the unit belongs. The user can either specify *id()* or `xtset` *panelvar timevar* before launching the command. See `xtset`
- `t(`*varname*`)` specifies the *timevar* of the observation. The user can either specify *t()* or `xtset` *panelvar timevar* before launching the command. See `xtset`
- `reps(#)` number of bootstrap repetitions
- `poly(#)` degree of polynomial approximation for the first stage
- `seed(#)` seed to be set before estimation
- `fsresiduals(`*newvarname*`)` store the first stage residuals (OP and LP only) in *newvarname*
- `translog` use a translog production function for estimation
- `level(#)` specifies the confidence level α

- `optimizer` available optimizers are Nelder Mead (nm), modified Newton-Raphson (nr), Davidon-Fletcher-Powell (dfp), Broyden-Fletcher-Goldfarb-Shanno (bfgs) and Berndt-Hall-Hall-Hausman (bhhh)
- `maxiter(#)` maximum number of iterations, default is 10,000
- `evaluator(`*`name`*`)` evaluator type
- `tolerance` sets the tolerance in optimization algorithm
- `gmm` uses the gmm command to run the estimation instead of ivregress - wrdg only
- `overidentification` includes the lagged polynomial in state and proxy variables among instrumtents - wrdg only

Predict

- *`newvarname`*`, residuals` stores the residuals of the log production function (equation 1) after the estimation - in *newvarname*.
- *`newvarname`*`, exponential` stores the exponential of the residuals of the log production function in *newvarname*.
- `parameters` reports the estimated input elasticities. In case of Cobb-Dougles, they are the estimated parameters. In the case of translog production function, $\bar{\beta}_w^{translog} = \frac{\sum_{i=1}^N \sum_{t=1}^T (\hat{\beta}_w + 2\hat{\beta}_{ww}\mathbf{w}_{it} + \hat{\beta}_{wx}\mathbf{x}_{it})}{N \times T}$ for free variable and, similarly, $\bar{\beta}_x^{translog} = \frac{\sum_{i=1}^N \sum_{t=1}^T (\hat{\beta}_x + 2\hat{\beta}_{xx}\mathbf{x}_{it} + \hat{\beta}_{wx}\mathbf{w}_{it})}{N \times T}$ for state variables.
- *`newvarname`*`, omega` stores the predicted values of omega - i.e., $\hat{\phi}_{it} - f(w_{it}, k_{it}, \hat{\beta})$ in *newvarname*

III.3 Example

In the following examples we show the use of `prodest`; interested readers will find that the syntax is similar to other user-written Stata modules for production function estimation, namely `opreg` (see Yasar et al. (2008)) for OP estimation, `levpet` (see Petrin et al. (2004)) for LP and `acfest` for ACF. Our command is able to estimate all models and adds new methodologies, is faster in many applications (see tables (1) and (2) for a comparison), allows the user to customize the optimization processes - which is a desirable feature mostly with ACF applications - and makes use of GMM optimization instead of non-linear least squares in OP estimation.

We test `prodest` on a dataset of Chilean firms 1995-2013[7]. Once uploaded the data, type

```
 .  xtset ID ANIO
       panel variable:  ID (unbalanced)
        time variable:  ANIO, 1995 to 2013, but with gaps
                delta:  1 unit

. prodest va, free(skilled unskilled) proxy(water ele) state(k) poly(3) met(lp)
> valueadded reps(50)
.........10.........20.........30.........40.........50

lp productivity estimator                        Cobb-Douglas PF

Dependent variable: value added                  Number of obs      =     91598
Group variable (id): ID                          Number of groups   =     17956
Time variable (t): ANIO
                                                 Obs per group: min =         1
                                                                avg =       5.1
                                                                max =        14

------------------------------------------------------------------------------
```

[7]It is a well-known and broadly used dataset in the related literature. See Petrin et al. (2004) among the others.

```
          va |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     skilled |   .2925325   .0060256    48.55   0.000     .2807225    .3043424
   unskilled |   .1793727   .0054555    32.88   0.000     .1686802    .1900652
           k |   .1457316    .003931    37.07   0.000      .138027    .1534362
------------------------------------------------------------------------------
Wald test on Constant returns to scale: Chi2 = 1152.39
                                           p = (0.00)

. prodest va, free(skilled unskilled) proxy(water ele) state(k) poly(3) met(wrdg)
> valueadded

wrdg productivity estimator                         Cobb-Douglas PF

Dependent variable: value added                     Number of obs      =     69376
Group variable (id): ID                             Number of groups   =     17956
Time variable (t): ANIO
                                                    Obs per group: min =         1
                                                                   avg =       5.1
                                                                   max =        14

------------------------------------------------------------------------------
          va |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     skilled |   .3458727   .0045537    75.95   0.000     .3369477    .3547978
   unskilled |   .1920877   .0035589    53.97   0.000     .1851125     .199063
           k |   .1385763   .0015803    87.69   0.000      .135479    .1416736
------------------------------------------------------------------------------
Wald test on Constant returns to scale: Chi2 = 2553.98
                                           p = (0.00)
```

```
. prodest va, free(skilled unskilled) proxy(water ele) state(k) poly(3) met(wrdg)
> valueadded gmm

wrdg productivity estimator gmm                 Cobb-Douglas PF

Dependent variable: value added                 Number of obs      =     69376
Group variable (id): ID                         Number of groups   =     17956
Time variable (t): ANIO
                                                Obs per group: min =         1
                                                               avg =       5.1
                                                               max =        14

------------------------------------------------------------------------------
          va |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     skilled |   .3135499   .0039656    79.07   0.000     .3057774    .3213224
   unskilled |   .2097768   .0033723    62.21   0.000     .2031673    .2163863
           k |   .1392172   .0029231    47.63   0.000      .133488    .1449464
------------------------------------------------------------------------------
Wald test on Constant returns to scale: Chi2 = 2697.60
                                           p = (0.00)
```

The output table is similar to most Stata panel commands[8] as it indicates the panel and timevar, the dependent variable, the methodology employed and the number of observations and groups.

[8]In programming the code and the output we were inspired by *levpet* both due to its user-friendly structure and to its clear output interpretation.

IV Methods

IV.1 OP, LP and WRDG with prodest

In Table (1) we report the estimates relative to various models / commands. Columns (1) and (2) refer to OLS and FE as benchmarks, while columns (3)-(6) report results of the OP methodology, without (3)-(4) and with (5)-(6) attrition. `levpet` with investment as a proxy variable yields the same results as `prodest`. Both models do not account for attrition in the data; in order to deal with the issue we show `opreg` - column (5) - and `prodest` with the attrition - column (6). In both cases there is no statistical differences between the models' estimates.[9] Again, our command proves to be faster than any other available module.

Table 1: Olley-Pakes (1996) comparison: Chilean dataset value added

	OLS	FE	Levpet	Prodest	Opreg	Prodest_exit
main						
β_k	0.116***	0.0828***	0.402***	0.402***	0.408***	0.398***
	(0.00127)	(0.00126)	(0.00904)	(0.00816)	(0.00893)	(0.0106)
β_{skil}	0.668***	0.458***	0.313***	0.313***	0.313***	0.313***
	(0.00317)	(0.00341)	(0.00746)	(0.00630)	(0.00746)	(0.00630)
β_{unskil}	0.436***	0.339***	0.224***	0.224***	0.224***	0.224***
	(0.00266)	(0.00283)	(0.00588)	(0.00551)	(0.00588)	(0.00551)
time	0.0480	1.480	77.11	59.78	455.6	196.2
N	91598	91598	60253	60253	60253	60253

Note: Column (1) reports results of a linear regression of log output - value added - on free and state variables, in column (2) we add individual fixed effects; column (3) uses the user-written command *levpet* (`levpet va, free(skilled unskilled) capital(k) proxy(inv) reps(50) valueadded`) with *investment* as proxy variable; in column (4) and (6) we perform the same exercise with *prodest* (`prodest va, free(skilled unskilled) state(k) proxy(inv) met(op) valueadded reps(50) [attrition]`), with and without the attrition; lastly, column (5) reports parameter estimates computed by the *opreg* command (`opreg va, exit(exit) free(skilled unskilled) proxy(inv) state(k)`)

Table (2), with a structure similar to (1), presents comparative results of OLS and FE models (benchmarks), `levpet` - (3) - and various methodologies implemented by `prodest`, namely *lp* (4), *lp* with attrition (5) and *wrdg* (6). Levinsohn and Petrin (2003) argue that OLS overestimate parameters of the free variables: it is the case in our application as well[10].

[9] This suggesting that the attrition is a relatively rare phenomenon in these markets

[10] The bias on state variable parameter, instead, depends on the correlation between inputs and productivity shocks; thus there exist no prior on it.

`prodest` (column 4) shows point estimates identical to `levpet` (3), but it is faster (89 vs 157 seconds). Attrition does not significantly affect results. Wooldridge methodology (column 6) yields results consistent with previous commands but with smaller standard errors.

Table 2: Levinsohn-Petrin (2004) comparison: Chilean dataset value added

	OLS	FE	Levpet	Prodest	Prodest_exit	Wooldridge
β_k	0.116***	0.0828***	0.146***	0.146***	0.147***	0.135***
	(0.00127)	(0.00126)	(0.00416)	(0.00424)	(0.00423)	(0.00157)
β_{skil}	0.668***	0.458***	0.293***	0.293***	0.293***	0.358***
	(0.00317)	(0.00341)	(0.00761)	(0.00630)	(0.00630)	(0.00449)
β_{unskil}	0.436***	0.339***	0.179***	0.179***	0.179***	0.210***
	(0.00266)	(0.00283)	(0.00641)	(0.00635)	(0.00635)	(0.00342)
time	0.0480	1.553	124.3	126.1	535.9	4.303
N	91598	91598	91598	91598	91598	69376

Note: Column (1) reports results of a linear regression of log output - value added - on free and state variables, in column (2) we add individual fixed effects; column (3) reports results using the user-written command *levpet* (`levpet va, free(skilled unskilled) capital(k) proxy(water ele) reps(50) valueadded`) with *investment* as proxy variable; in column (4) and (5) we perform the same exercise with *prodest* (`prodest va, free(skilled unskilled) state(k) proxy(water ele) met(lp) valueadded reps(50) [attrition]`), with and without the attrition; at last, column (6) reports the estimation with *prodest* using the Wooldridge method with a second order polynomial (`prodest va, free(skilled unskilled) state(k) proxy(water ele) poly(2) met(wrdg) valueadded`).

IV.2 ACF: simulations and discussion

Ackerberg et al. (2015) propose a correction to the OP and the LP methodologies. Their paper presents i) a new approach, ii) an application to real data and iii) estimates using simulated data. These have been generated according to 3 distinct DGPs: with serially correlated wages and labor input decisions set at t-δ (DGP 1), with an optimization error in labor (DGP 2) and both the elements at once (DGP 3). On top of that, they simulate data with 4 different amounts of measurement error in the intermediate input (σ_m^2 = 0,.1,.2 and .5, respectively).

In Tables (3) and (5) we report a comparison between `prodest` and `acfest` by Manjón and Mañez (2016). More specifically, we run various models on both the Chilean and a simulated dataset (DGP2) in order to test several features of both commands. We focus on table (3): *prodest* - columns (3), (6) and (9) - yields more stable and 'plausible' results. All estimates $\hat{\beta}_g^{prodest} \in [0,1], g = [skil, unskil, k]$ and - with 50 cluster bootstrap repetitions - standard errors are smaller than *acfest*'s, on average. Timing shows mixed evidence: *prodest* is faster in the gross output, 3^{rd} degree application but way slower with the value added, 2^{nd} degree. In its 3^{rd} degree version the elapsed time is very similar for the two commands.

Table 3: ACF (2015) comparison: Chilean dataset

	GO			VA - II			VA		
	LP	ACFest	Prodest	LP	ACFest	Prodest	LP	ACFest	Prodest
β_{skil}	0.268***	1.991***	0.427***	0.322***	-0.147***	0.701***	0.309***	-0.212***	0.702***
	(0.006)	(0.380)	(0.005)	(0.006)	(0.040)	(0.008)	(0.006)	(0.042)	(0.007)
β_{unskil}	0.160***	-0.528***	0.279***	0.210***	-0.089***	0.467***	0.192***	-0.161***	0.467***
	(0.006)	(0.185)	(0.006)	(0.005)	(0.032)	(0.002)	(0.005)	(0.037)	(0.002)
β_k	0.073***	0.069***	0.039***	0.139***	0.252***	0.060***	0.143***	0.269***	0.057***
	(0.003)	(0.011)	(0.003)	(0.004)	(0.008)	(0.004)	(0.004)	(0.008)	(0.004)
time	140	792	415	85	234	330	93	294	297
N	93,191	71,369	93,191	91,598	70,238	91,598	91,598	70,238	91,598

Note: In colums (1)-(3) the dependent variable is *log(gross output)* - GO - in (4)-(9) is *log(value added)* - VA. (1), (4) and (7) report the benchmark Levinsohn-Petrin estimates; (2), (5) and (8) report results obtained on Chilean data using the user-written command *acfest* with 50 bootstrap repetitions (`acfest [go/va], free(skilled unskilled) proxy(ele) state(k) nbs(50) robust [va] [second]`), whereas columns (3), (6) and (9) refer to the same models estimated with *prodest* (`prodest [go/va], free(skilled unskilled) proxy(ele) state(k) acf reps(50) [va] [poly(2)]`). Value added models have been estimated with a second degree - columns (4)-(6) - and third-degree polynomials - columns (7)-(9).

Table (5) shows various value added models implemented on a DGP3 simulated dataset, with $\beta_k^\star = .4$ and $\beta_l^\star = .6$ and 1,000 firms observed 10 times. Though constantly slower, *prodest* performs better than *acfest* both in the second and in the third order polynomial versions: in all cases, Newton-Raphson (NR) algorithm overcomes Davidon-Fletcher-Powell (DFP) and Nelder-Mead (NM) in terms of model Mean Squared Error.

Table (6) replicates Table I of ACF paper: for each DGP/measurement error couple we present the estimated parameters using ACF-corrected and LP parameters. A Monte carlo simulation of the estimates is performed with 1000 repetitions and results reported are coefficient averages with the standard deviations across the replications. The true values of β_l and β_k are .6 and .4, respectively, and ACF persistently performs better than Levinsohn-

Table 4: ACF (2015) comparison: Chilean dataset

	GO			VA - II			VA		
	LP	ACFest	Prodest	LP	ACFest	Prodest	LP	ACFest	Prodest
β_{skil}	0.268***	1.991***	0.427***	0.322***	-0.147***	0.701***	0.309***	-0.212***	0.702***
	(0.006)	(0.398)	(0.005)	(0.006)	(0.038)	(0.008)	(0.006)	(0.035)	(0.007)
β_{unskil}	0.160***	-0.528***	0.279***	0.210***	-0.089**	0.467***	0.192***	-0.161***	0.467***
	(0.006)	(0.137)	(0.006)	(0.005)	(0.037)	(0.002)	(0.005)	(0.034)	(0.002)
β_k	0.073***	0.069***	0.039***	0.139***	0.252***	0.060***	0.143***	0.269***	0.057***
	(0.003)	(0.011)	(0.003)	(0.004)	(0.007)	(0.004)	(0.004)	(0.007)	(0.004)
time	145	825	468	98	315	382	126	299	360
N	93,191	71,369	93,191	91,598	70,238	91,598	91,598	70,238	91,598

Note: In colums (1)-(3) the dependent variable is *log(gross output)* - GO - in (4)-(9) is *log(value added)* - VA. (1), (4) and (7) report the benchmark Levinsohn-Petrin estimates; (2), (5) and (8) report results obtained on Chilean data using the user-written command *acfest* with 50 bootstrap repetitions (`acfest [go/va], free(skilled unskilled) proxy(ele) state(k) nbs(50) robust [va] [second]`), whereas columns (3), (6) and (9) refer to the same models estimated with *prodest* (`prodest [go/va], free(skilled unskilled) proxy(ele) state(k) acf reps(50) [va] [poly(2)]`). Value added models have been estimated with a second degree - columns (4)-(6) - and third-degree polynomials - columns (7)-(9).

Table 5: ACF (2015) comparison: DGP3 dataset

	VA - II					VA				
	LP	ACFest	Prodest	Prodest-DFP	Prodest-NR	LP	ACFest	Prodest	Prodest-DFP	Prodest-NR
β_l	0.473***	1.009***	0.596***	0.610***	0.598***	0.473***	1.009***	0.596***	0.610***	0.597***
	(0.003)	(0.004)	(0.006)	(0.008)	(0.008)	(0.003)	(0.004)	(0.006)	(0.007)	(0.010)
β_k	0.562***	-0.012*	0.386***	0.464***	0.401***	0.562***	-0.012*	0.386***	0.464***	0.397***
	(0.016)	(0.006)	(0.007)	(0.016)	(0.019)	(0.016)	(0.006)	(0.007)	(0.024)	(0.023)
time	9	21	24	21	34	9	22	30	27	47
N	10,000	9,000	10,000	10,000	10,000	10,000	9,000	10,000	10,000	10,000

Note: (1)-(5) report results of value added estimation with a 2^{nd}-order degree polynomial, (6)-(10) with a 3^{rd}-order degree. (1) and (6) are Levinsohn-Petrin benchmark results, (2) and (7) are *acfest*'s - `acfest y, free(l) proxy(m) state(k) va nbs(50) [second] robust` - while remaining *prodest* models estimated with Nelder-Mead, Davidon-Fletcher-Powell and Newton-Raphson optimization algorithms, respectively.

Petrin, even if the routine takes longer to complete. In table (7), we report the same results in terms of $Bias$ and MSE: in all but one model (i.e. DGP2, no measurement error) ACF shows a Mean Squared Error which is persistently of an order of magnitude smaller than LP's. Results are extremely robust to various specifications of the AR(1) parameter of the productivity - i.e., ρ - and optimizers across simulations[11].

[11] See Appendix

Table 6: ACF and LP - Monte Carlo Simulations

	ACF				LP			
	β_l		β_k		β_l		β_k	
Meas. Error	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$								
0.0	0.609	0.015	0.415	0.024	-0.000	0.005	1.089	0.030
0.1	0.594	0.019	0.425	0.020	0.676	0.009	0.364	0.012
0.2	0.634	0.024	0.399	0.017	0.788	0.007	0.241	0.010
0.5	0.670	0.011	0.356	0.013	0.875	0.005	0.170	0.126
DGP2 - Optimization Error in Labor								
0.0	0.619	0.022	0.424	0.024	0.600	0.003	0.399	0.013
0.1	0.610	0.016	0.404	0.019	0.753	0.004	0.255	0.009
0.2	0.612	0.018	0.397	0.021	0.807	0.004	0.202	0.011
0.5	0.651	0.019	0.374	0.016	0.863	0.003	0.312	0.304
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)								
0.0	0.620	0.022	0.478	0.052	0.473	0.004	0.570	0.016
0.1	0.610	0.019	0.440	0.022	0.634	0.005	0.412	0.012
0.2	0.606	0.011	0.428	0.017	0.700	0.005	0.344	0.012
0.5	0.615	0.013	0.414	0.019	0.777	0.005	0.877	0.605

Note: 1000 replications. In each replication the estimated models are `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`. True values of parameters are $\beta_l = 0.6$ and $\beta_k = 0.4$. Standard deviations have been calculated among 1000 replications. ρ is set at .7 and we used Newton-Raphson optimizer.
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)

ACF methodology, however, shows serious limitations in empirical applications. Table (8) reports estimates - in terms of bias and MSE - obtained using a DGP3 simulated dataset with different starting points for the optimization routine with the Newton-Raphson optimizer. The starting points $[\beta^0_{lnl}, \beta^0_{lnk}]$ in column (1) are [0.1,(1-0.1)], in column (2) are [0.2,(1-0.2)] and up to [0.9,0.1]. It is immediate to note how results differ, even dramatically, when

Table 7: ACF and LP Bias and MSE - Monte Carlo Simulations

Meas. Error	ACF			LP		
	Bias_l	Bias_k	MSE	Bias_l	Bias_k	MSE
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$						
0.0	0.009	0.015	0.001	-0.600	0.689	0.418
0.1	-0.006	0.025	0.001	0.076	-0.036	0.004
0.2	0.034	-0.001	0.001	0.188	-0.159	0.030
0.5	0.070	-0.044	0.004	0.275	-0.230	0.072
DGP2 - Optimization Error in Labor						
0.0	0.019	0.024	0.001	-0.000	-0.001	0.000
0.1	0.010	0.004	0.000	0.153	-0.145	0.022
0.2	0.012	-0.003	0.000	0.207	-0.198	0.041
0.5	0.051	-0.026	0.002	0.263	-0.088	0.085
DGP3 - Opt Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)						
0.0	0.020	0.078	0.005	-0.127	0.170	0.023
0.1	0.010	0.040	0.001	0.034	0.012	0.001
0.2	0.006	0.028	0.001	0.100	-0.056	0.007
0.5	0.015	0.014	0.000	0.177	0.477	0.312

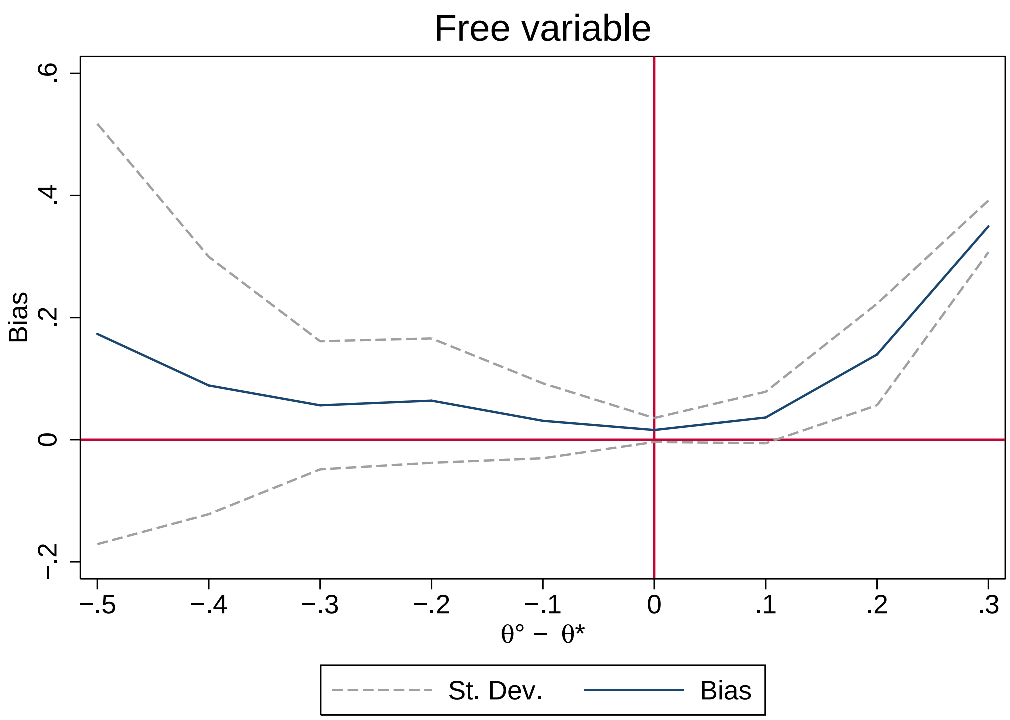
Note: 1000 replications. In each replication the estimated models are `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`. True values of parameters are $\beta_l = 0.6$ and $\beta_k = 0.4$. ρ is set at .7 and we used Newton-Raphson optimizer. The definition of $\text{Bias}_{ig} = \hat{\beta}_{ig} - \beta^{\star}_{ig}$, where $g \in [l,k]$ stands for state and free variables and $i \in [1,..,1000]$ identifies each Monte Carlo replication. Bias_k and Bias_k are averaged across replications. MSE is defined as the average across replications of $MSE_i = (\text{Bias}^2_{ik} + \text{Bias}^2_{il})/2$.
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)

starting points depart from the true values - in column (6), in blue, the starting points are fixed at the true values.[12] This is true in particular for extreme values of the starting points (columns 1 and 9)[13] when the model yields to estimates greater than 1, non-significant or

[12]Apart from the optimization starting points the data and the command are the same: `prodest lny, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) va reps(50) acf init("‘starting_points’")`

[13]In our exercises we fixed the sum of starting points to 1. The bias is even worse when starting from

Figure 1: Bias of ACF estimates with respect to starting points - Free variable



Note: The figure reports on the y axis the average bias (continuous line) with its confidence interval (dotted lines) for each difference level between the starting point and the real parameter value (x axis).

basically 0.

Figure (1) reports the average bias on the y axis - we define bias as $|E(\hat{\theta}) - \theta^{\star}|$ - and the distance of the starting point from the true parameter value on the x axis - i.e., $\theta^0 - \theta^{\star}$. It is straightforward to note that both the bias and its standard deviation reach their minimum at $\theta^0 = \theta^{\star}$ (that is, when the starting point is at the true value); however, the bias pattern differs depending on whether the starting point is above of below the true θ. Lower starting points lead to very noisy but not much biased results, while when $\theta^0 > \theta^{\star}$ the bias increases and is statistically significant.

Figure (2) is the heat map of the MSE (defined as $\sum_j (\hat{\beta}_j - \beta_j^{\star})^2$, with $j = [k, l]$) of 6,400 models estimated with starting points in a $\pm .4$ range around the true parameter value (step

larger values (in absolute terms).

Table 8: ACF (DGP3) Bias and MSE - Monte Carlo Simulations

Panel (a): Meas. Err. 0

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Bias_l	0.277	0.030	0.041	0.020	0.007	0.004	0.016	0.156	0.397
	(0.177)	(0.057)	(0.042)	(0.061)	(0.006)	(0.004)	(0.017)	(0.069)	(0.009)
Bias_k	0.301	0.089	0.113	0.053	0.020	0.026	0.085	0.925	0.382
	(0.184)	(0.141)	(0.103)	(0.063)	(0.016)	(0.020)	(0.085)	(0.930)	(0.023)
MSE	0.116	0.016	0.013	0.005	0.000	0.001	0.008	0.875	0.152
	(0.096)	(0.109)	(0.048)	(0.025)	(0.001)	(0.001)	(0.013)	(1.863)	(0.012)

Panel (b): Meas. Err. 0.1

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Bias_l	0.053	0.073	0.034	0.011	0.010	0.004	0.010	0.106	0.409
	(0.088)	(0.178)	(0.091)	(0.024)	(0.028)	(0.004)	(0.006)	(0.108)	(0.053)
Bias_k	0.146	0.194	0.056	0.038	0.030	0.029	0.091	0.251	0.480
	(0.249)	(0.334)	(0.092)	(0.026)	(0.026)	(0.015)	(0.069)	(0.501)	(0.093)
MSE	0.047	0.093	0.011	0.001	0.001	0.001	0.007	0.168	0.205
	(0.170)	(0.354)	(0.036)	(0.008)	(0.009)	(0.001)	(0.007)	(0.924)	(0.075)

Panel (c): Meas. Err. 0.2

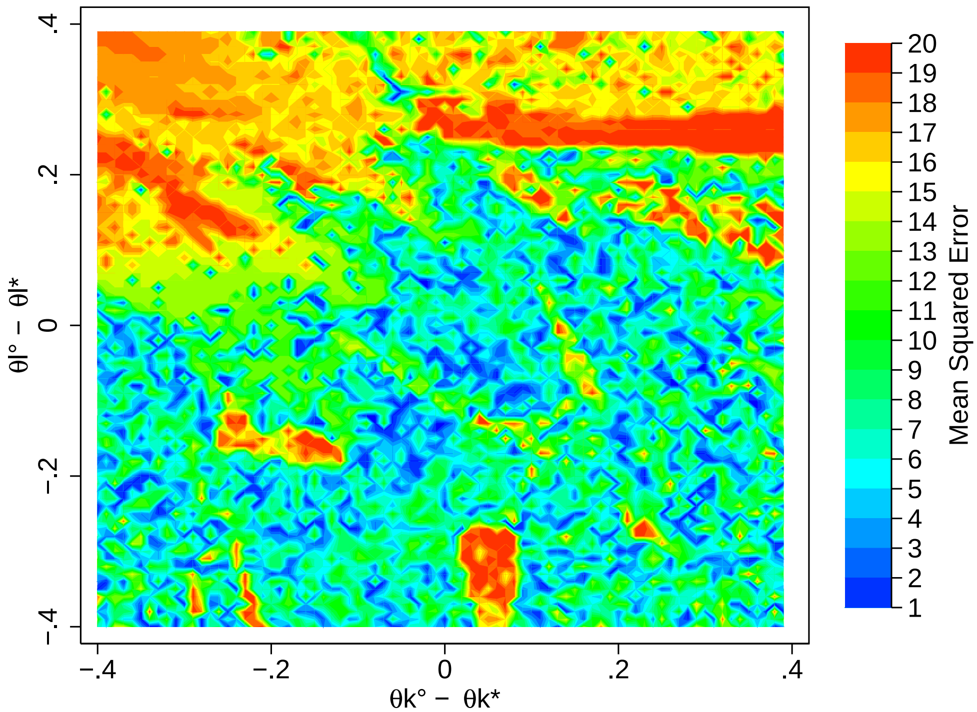
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Bias_l	0.054	0.032	0.036	0.009	0.019	0.006	0.011	0.140	0.329
	(0.108)	(0.101)	(0.073)	(0.020)	(0.057)	(0.005)	(0.007)	(0.115)	(0.009)
Bias_k	0.115	0.068	0.103	0.028	0.036	0.027	0.070	0.246	0.332
	(0.276)	(0.152)	(0.166)	(0.023)	(0.052)	(0.013)	(0.054)	(0.533)	(0.016)
MSE	0.052	0.019	0.022	0.001	0.004	0.000	0.004	0.189	0.109
	(0.440)	(0.156)	(0.067)	(0.006)	(0.018)	(0.000)	(0.004)	(0.756)	(0.009)

Panel (d): Meas. Err. 0.5

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Bias_l	0.052	0.052	0.033	0.024	0.040	0.014	0.021	0.229	0.296
	(0.097)	(0.116)	(0.048)	(0.035)	(0.076)	(0.009)	(0.015)	(0.057)	(0.006)
Bias_k	0.057	0.057	0.046	0.034	0.038	0.019	0.043	0.215	0.285
	(0.114)	(0.129)	(0.120)	(0.056)	(0.068)	(0.012)	(0.045)	(0.269)	(0.012)
MSE	0.014	0.018	0.010	0.003	0.007	0.000	0.002	0.087	0.085
	(0.105)	(0.158)	(0.107)	(0.012)	(0.021)	(0.000)	(0.003)	(0.176)	(0.005)

Note: 1000 replications. In columns (1) to (9) we report the Bias in ACF estimates with different optimization starting points from [0.1,(1-0.1)] to [0.9,(1-0.9)] and the relative MSE. In blue - column (6) - results with the starting points equal to the true values. All results are averaged across 1000 Monte Carlo simulations.

Figure 2: Mean Squared Error and Starting Points in ACF estimates



Note: heat map of the Mean Squared Error (defined as $\sum_j(\hat{\beta}_j - \beta_j^\star)^2$) of ACF models with Newton-Raphson optimizer.

= 0.01). More specifically, we use a simulated dataset (DGP3, measurement error = 0.2) and for each pair $[\theta_l^0, \theta_k^0]$, $\theta_l^0 \in [0.2, 1]$ and $\theta_k^0 \in [0, 0.8]$, we run the command `prodest lny, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded acf init("`$[\theta_l^0, \theta_k^0]$`")`. The plot highlights the limitations of ACF methodology with respect to optimization procedures: it is easy to note how the MSE increases as the starting point for the free variable outweighs the true value. Apart from this element, however, no clear pattern emerges as we find a patchwork of high-MSE regions throughout the starting points' domain and single models yielding correct values within regions of severely biased estimates. In turn, this indicates that the optimization procedures (Newton-Raphson in the present case) are often trapped in local maxima.

In the spirit of Knittel and Metaxoglu (2014), who show how often numerical convergence in nonlinear GMM models leads to severely biased results[14], in the Appendix we have repeated all the above exercises with different choices of optimizers. More specifically, all results hold with *Davidon-Fletcher-Powell* (table A.4), *Broyden-Fletcher-Goldfarb-Shanno* (table A.3) and *Berndt-Hall-Hall-Hausman* (table A.12).

IV.3 Translog Production Function - ACF

Firstly introduced by Kmenta (1967), the translog production function has been proposed as a feasible approximation of CES production functions through a 2^{nd} order Taylor expansion. Unlike the Cobb-Douglas, the Translog does not require the assumption of smooth substitution between production factors.

In the present section, we briefly describe the models estimated via the `translog` option of ACF methods in `prodest`.[15] The translog production function

$$y_{it} = \mathbf{w}_{it}\beta_w + \mathbf{x}_{it}\beta_x + \mathbf{w}_{it}^2\beta_{ww} + \mathbf{x}_{it}^2\beta_{xx} + \sum_{j,k}\beta_{wx}w^k x^j + \omega_{it} + \epsilon_{it} \tag{35}$$

The first stage equation, which is the same in the value added and gross output cases, turns into

$$y_{it} = \mathbf{w}_{it}\beta_w + \mathbf{x}_{it}\beta_x + \mathbf{w}_{it}^2\beta_{ww} + \mathbf{x}_{it}^2\beta_{xx} + \sum_{j,k}\beta_{wx}w^k x^j + h(p_{it}, \mathbf{x}_{it}, \mathbf{w}_{it}) + \epsilon_{it} = \Phi(p_{it}, \mathbf{x}_{it}, \mathbf{w}_{it}) + \epsilon_{it} \tag{36}$$

Once obtained $\hat{\Phi}$ the estimation of productivity terms $\theta = (\beta_w, \beta_x, \beta_{ww}, \beta_{xx}, \beta_{wx})$ in (35) follows as in the usual ACF case.[16] The interpretation of translog parameters, though, differs with respect to that of the Cobb-Douglas ACF. Indeed, the elasticities for free and

[14] In the paper they deal with models *à la* Berry, Levinsohn and Pakes.

[15] See De Loecker and Warzynski (2012) and Gandhi et al. (2011) for a review of the models presented here.

[16] Exploiting the fact that $\hat{\omega}_{it}(\theta) = \hat{\Phi}_{it} - \mathbf{w}_{it}\beta_w - \mathbf{x}_{it}\beta_x - \mathbf{w}_{it}^2\beta_{ww} - \mathbf{x}_{it}^2\beta_{xx} - \sum_{j,k}\beta_{wx}w^k x^j$ is possible to regress it on its past values and recover the productivity shocks ξ_{it} and proceed with the GMM estimation.

state variables are given by:

$$\bar{\beta}_{translog} = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T}(\hat{\beta}_w + 2\hat{\beta}_{ww}\mathbf{w}_{it} + \hat{\beta}_{wx}\mathbf{x}_{it})}{N \times T}$$

$$\bar{\gamma}_{translog} = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T}(\hat{\beta}_x + 2\hat{\beta}_{xx}\mathbf{x}_{it} + \hat{\beta}_{wx}\mathbf{w}_{it})}{N \times T}$$

Table (9) reports results of Cobb-Douglas (odd columns) and Translog production functions (even columns) both for gross output and value added models. The number of translog parameters are obviously higher, and increasing the number of parameters implies that the GMM optimization takes longer to complete.

Table 9: Translog Production Function comparison: Chilean dataset

	VA		GO	
	Cobb-Douglas	Translog	Cobb-Douglas	Translog
β_{skil}	0.617***	0.512***	0.515***	0.816***
	(0.027)	(0.000)	(0.000)	(0.002)
β_{unskil}	0.603***	0.525***	0.523***	0.779***
	(0.018)	(0.000)	(0.000)	(0.002)
β_{k}	0.053***	0.726***	0.008***	0.573***
	(0.002)	(0.001)	(0.003)	(0.000)
β_{water}			0.724***	0.569***
			(0.001)	(0.000)
β_{skil^2}		0.639***		0.603***
		(0.000)		(0.001)
$\beta_{skil,unskil}$		0.542***		0.540***
		(0.000)		(0.000)
$\beta_{skil,k}$		0.548***		0.529***
		(0.000)		(0.000)
β_{unskil^2}		0.499***		0.815***
		(0.000)		(0.002)
$\beta_{unskil,k}$		0.492***		0.433***
		(0.000)		(0.000)
β_{k^2}		-0.302***		-0.472***
		(0.002)		(0.007)
$\beta_{skil,water}$				0.713***
				(0.001)
$\beta_{unskil,water}$				0.551***
				(0.000)
$\beta_{k,water}$				0.134***
				(0.002)
β_{water^2}				0.441***
				(0.000)
time	44	202	77	294
N	91,598	91,598	93,191	93,191
# Pars	3	9	4	14

Note: in columns (1)-(2) the parameters of Cobb-Douglas and Translog production function on value added models - VA - while in columns (3)-(4) results of gross output models - GO. All models have been estimated with `prodest [va/go], free(skilled unskilled) proxy(water) state(k) acf [va] reps(5) init(".5,.5,.5")`.

IV.4 The `endogenous` option

The *endogeous()* option in `prodest` allows the user to specify one or more variables which endogenously affect the dynamics of productivity ω. In particular, if any variable $a_{i,t}$ has an effect on productivity level at time t, the law of motion (2) should read $\omega_{it} = g(\omega_{it-1}, a_{i,t-1}) + \xi_{i,t}$ - i.e., ω_{it} follows a first-order markov chain process and $g(.)$ is a non-parametric function of ω_{it-1} and a_{it-1}. Such model is able to capture productivity changes conditional on the level of the endogenous variables: i.e., it accounts for the fact that firms update their expectation of the productivity level and adjust their investment based on the optimal level of the endogenous variable.

Models with endogenous variables have been implemented, among the others, by De Loecker (2007) - who used the lagged export quotas as drivers of productivity dynamics - and Doraszelski and Jaumandreu (2013) - who account for the R&D expenditure in estimating ω. Finally, Konings and Vanormelingen (2015) evaluate the impact of workforce training on both output and firm's productivity. Using a subsample of their data, in table 10 we report two examples of the `endogenous()` option on both *LP* and *ACF* models.

Table 10: OLS, LP and ACF models with and without the `endogenous` option

	OLS		LP		ACF	
			Plain	End	Plain	End
Labour	0.617***	0.509***	0.615***	0.615***	0.667***	0.692***
	(0.00907)	(0.0197)	(0.0200)	(0.0200)	(0.0216)	(0.0171)
Capital	0.191***	0.104***	0.0969***	0.0989***	0.135***	0.125***
	(0.00484)	(0.0103)	(0.0259)	(0.0263)	(0.0320)	(0.0315)
Training	0.181***	0.0405	0.0917**	0.110***	0.202***	0.194***
	(0.0535)	(0.0313)	(0.0358)	(0.0384)	(0.0593)	(0.0588)
N	2651	2651	2651	2651	2651	2651
FE		✓				

Note: OLS and OLS with fixed effects (columns 1 and 2, respectively) are compared to LP (columns 3-4) and ACF (columns 5-6). In columns *Plain* models without the endogenous variable, which is instead added in *End* columns. The model estimated is `prodest y if e(sample), free( Labour ) state( Capital Training ) proxy( Materials ) va met(lp) opt(dfp) reps(50) [acf] [endogenous(lagTraining)]`

IV.5 Wooldridge and MrEst

Introducing dynamic panel instruments could be useful in the estimation of "large N, small T" panel datasets. The overidentification helps improving estimation fit by increasing lags - and moment conditions. Table (11) reports the results of the estimation of LP, Wooldridge and MrEst with 2, 3 and 4 lags on subsets of Chilean data. In particular, we report the $\hat{\beta}_{sk}$, $\hat{\beta}_{unsk}$ and $\hat{\beta}_k$[17], estimated with the LP methodology and averaged across 61 industrial sectors (CIIU2) with various sample sizes - panel (a). We define $Bias_j = \hat{\beta}_j - \beta_j^{lp}$, $\forall j \in [sk, unsk, k]$ and $MSE = \mathrm{E}(Bias_j^2)$ (i.e. we test Wooldridge and MrEst estimator performance with respect to the benchmark LP method) and report their average values across sectors.

MrEst consistently performs better than Wooldridge in terms of Mean Squared Error - even if not always in terms of bias. MrEst models are particularly time intensive but the computational time does not increase dramatically with the number of lags required for each dynamic panel instrument, whereas it increases consistency and precision.

In table (12) we report the MSE of MrEst - with 2, 3 or 4 lags - on simulated data (DGP2 , no measurement error) as n increases with T fixed - panel (a) - and as T increases with n fixed. In particular, increasing the sample size leads to lower MSE and, as expected, adding lags increases estimate precision. Increasing the time dimension keeping n fixed, instead, does not appear to have clear effects on the Mean Squared Error.

[17]Skilled labor, unskilled labor and capital, respectively.

Table 11: Wooldridge and MrEst - various DGP

Panel (a): Levinsohn-Petrin

	$\hat{\beta}_{sk}$	$\hat{\beta}_{unsk}$	$\hat{\beta}_k$	MSE
Levinsohn-Petrin	0.303	0.228	0.039	0.000
	(0.121)	(0.086)	(0.045)	(0.000)

Panel (b): Wrdg and MrEst: Bias + MSE

	$Bias_{sk}$	$Bias_{unsk}$	$Bias_k$	MSE
Wooldridge	-0.007	-0.014	-0.003	0.002
	(0.050)	(0.041)	(0.021)	(0.002)
MrEst - 2 lags	-0.025	-0.017	-0.004	0.001
	(0.042)	(0.036)	(0.016)	(0.002)
MrEst - 3 lags	-0.025	-0.014	-0.002	0.001
	(0.043)	(0.034)	(0.013)	(0.002)
MrEst - 4 lags	-0.026	-0.014	-0.004	0.001
	(0.044)	(0.033)	(0.012)	(0.002)

Note: in panel (a) we report the average $\hat{\beta}$ value of Levinsohn and Petrin estimation on 60 subsets (i.e. industry sectors, according to the CIIU2 variable) of Chilean firm-level data. These are the benchmark values: we define $Bias_j = \hat{\beta}_j - \beta_j^{lp}$, $\forall j \in [sk, unsk, k]$ and $MSE = \mathrm{E}(Bias_j^2)$. Panel (b) reports the average bias and the MSE, with their standard deviations, of Wooldridge and MrEst models (various lags).

Table 12: MrEst - MSE with simulated data (DGP2)

Panel (a): $n \to \infty$, fixed T

	(1)	(2)	(3)	(4)	(5)	(6)
MrEst - 2 lags	0.184	0.111	0.067	0.026	0.000	0.005
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
MrEst - 3 lags	0.179	0.106	0.064	0.025	0.000	0.004
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
MrEst - 4 lags	0.175	0.099	0.062	0.025	0.000	0.004
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
N	1500	3000	5000	6500	8000	10000

Panel (b): increasing T, fixed n

	(1)	(2)	(3)	(4)	(5)	(6)
MrEst - 2 lags	0.032	0.080	0.070	0.040	0.052	0.067
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
MrEst - 3 lags	0.029	0.076	0.067	0.037	0.049	0.064
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
MrEst - 4 lags	0.028	0.075	0.065	0.036	0.048	0.062
	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
N	2500	3000	3500	4000	4500	5000

Note: MSE of MrEst with 2,3 and 4 lags on simulated data - DGP3, no measurement error - with increasing number of firms in the sample and T = 10 fixed.

V Conclusions

There are three main approaches in the literature of production function estimation, namely Instrumental Variable, Fixed Effects and Control Function. Olley and Pakes, Levinsohn and Petrin, Ackerberg, Caves and Frazer all decisively contributed to the latter strand by developing widely used methodologies. Wooldridge showed how to implement the control function approach in a system GMM framework. We build a new estimator, MrEst, based on his results, adding dynamic panel instruments in order to improve efficiency and gain predictive power. Our estimator proves to be consistent and to perform better than Wooldridge's as the number of individuals increases.

Moreover, we provide evidence that non-linear GMM models in general, and OP/LP models with ACF correction in particular, have to be handled with care in empirical applications. In our Monte Carlo simulations - based on the ACF data generating process - results change dramatically depending on the starting points passed to the optimization algorithm; this result is robust to different choices of optimizer, model, and sample.

Furthermore, we have introduced a new Stata module, `prodest`, aimed at implementing all the methods listed above in a user-friendly and effective way. It performs well in comparison with other user-written commands on several datasets and introduces new estimation methods. It features a number of options for expert users aimed at controlling the optimization procedures, the model specification and results handling.

VI Saved Results

`prodest` saves in `e()`:

Scalars

`e(N)`	number of observations
`e(N_g)`	number of panel IDs
`e(tmin)`	min number of periods
`e(tmean)`	average number of periods
`e(tmax)`	max number of periods

Macros

`e(cmd)`	prodest
`e(depvar)`	depvar
`e(free)`	free variable(s)
`e(state)`	state variable(s)
`e(proxy)`	proxy variable(s)
`e(control)`	control variable(s)
`e(endogenous)`	endogenous variable(s)
`e(technique)`	optimization technique
`e(idvar)`	*panelvar*
`e(timevar)`	*timevar*

`e(method)`	estimation method
`e(model)`	va or go
`e(correction)`	correction - ACF
`e(hans_j)`	Hansen's J - Wrdg
`e(hans_p)`	Hansen's J p-value
`e(waldT)`	Wald Test on constant returns to scale
`e(waldP)`	Wald Test P-value

Matrices

`e(b)`	coefficient vector
`e(V)`	var-covar matrix

Functions

`e(sample)`	estimation sample

VII Acknowledgments

Special thanks go to Francesco Decarolis who supervised the whole project, providing expertise and support. Also, we thank Daniel Ackerberg, Federico Belotti, Sara Calligaris, Leonardo Giuffrida, Andrea Pozzi, Lorenzo Rovigatti as well as several `prodest` users for comments that improved the module and the manuscript. We also thank an anonymous referee for precious insights on command implementation. Any errors are our own.

VIII References

Ackerberg, D., K. Caves, and G. Frazer. 2015. Identification Properties of Recent Production Function Estimators. *Econometrica* 83(6): 2411–2451.

Blundell, R., and S. Bond. 1998. Initial conditions and moment restrictions in dynamic panel data models. *Journal of Econometrics* 87: 115–143.

Bond, S., and M. Soderbom. 2005. Adjustment Costs and the Identification of Cobb Douglas Production Functions. *Working Paper* .

De Loecker, J. 2007. Do exports generate higher productivity? Evidence from Slovenia. *Journal of international economics* 73(1): 69–98.

De Loecker, J., and F. Warzynski. 2012. Markups and firm-level export status. *The American Economic Review* 102(6): 2437–2471.

Doraszelski, U., and J. Jaumandreu. 2013. R&D and productivity: Estimating endogenous productivity. *Review of Economic Studies* 80(4): 1338–1383.

Gandhi, A., S. Navarro, and D. A. Rivers. 2011. On the identification of production functions: How heterogeneous is productivity? *Report, Winsconsin-Madison* .

Hopenayn, H. A. 1992. Entry, Exit and Firm Dynamics in Long Run Equilibrium. *Econometrica* 5(60): 1127–1150.

Kmenta, J. 1967. On estimation of the CES production function. *International Economic Review* 8(2): 180–189.

Knittel, C. R., and K. Metaxoglu. 2014. Estimation of Random-Coefficient Demand Models: Two Empiricists' Perspective. *The Review of Economics and Statistics* 96(1): 34–59.

Konings, J., and S. Vanormelingen. 2015. The impact of training on productivity and wages: firm-level evidence. *Review of Economics and Statistics* 97(2): 485–497.

Levinsohn, J., and A. Petrin. 2003. Estimating Production Functions Using Inputs to Control for Unobservables. *The Review of Economic Studies* 70(2): 317–341.

Manjón, M., and J. Mañez. 2016. Production function estimation in Stata using the Ackerberg-Caves-Frazer method. *Stata Journal* 16(4): 900–916(17).

Melitz, M. J. 2003. The Impact of Trade on Intra-Industry Reallocations and Aggregate Industry Productivity. *Econometrica* 71(6): 1695–1725.

Olley, S. G., and A. Pakes. 1996. The Dynamics of Productivity in the Telecommunications Equipment Industry. *Econometrica* 64(6): 1263–1297.

Petrin, A., B. P. Poi, and J. Levinsohn. 2004. Production function estimation in Stata using inputs to control for unobservables. *The Stata Journal* 4(2): 113–123.

Robinson, P. M. 1988. Root-N-consistent semiparametric regression. *Econometrica: Journal of the Econometric Society* 931–954.

Wooldridge, J. M. 1996. Estimating systems of equations with different instruments for different equations. *Journal of Econometrics* 74: 387–405.

———. 2009. On estimating firm-level production functions using proxy variables to control for unobservables. *Economics Letters* 104: 112–114.

Yasar, M., R. Raciborski, and B. Poi. 2008. Production function estimation in Stata using Olley and Pakes method. *The Stata Journal* 8(2): 221–231.

About the authors

Vincenzo Mollisi is a PhD candidate in Law, Economics and Institutions at Tor Vergata University in Rome, Italy and visiting researcher at the University of Bolzano.

Gabriele Rovigatti graduated in Economics and Finance at Tor Vergata University in Rome, Italy, and is a post-doc visiting researcher at Chicago Booth University and research scholar at the Einaudi Institute for Economics and Finance in Rome, Italy.

A Online Appendix

Table A.1: ACF and LP - $\rho = .5$

Panel (a): Meas. Err. 0

	DGP1		DGP2		DGP3	
	ACF	LP	ACF	LP	ACF	LP
lnl	0.596***	-0.00474	0.572***	0.598***	0.582***	0.431***
	(0.0169)	(0.00442)	(0.0219)	(0.00266)	(0.0116)	(0.00361)
lnk	0.394***	1.025***	0.431***	0.407***	0.418***	0.579***
	(0.0202)	(0.0183)	(0.0282)	(0.0107)	(0.0163)	(0.0145)
time	37.85	6.864	28.27	6.329	36.31	6.618

Panel (b): Meas. Err. 0.1

	ACF	LP	ACF	LP	ACF	LP
lnl	0.583***	0.535***	0.576***	0.742***	0.582***	0.564***
	(0.0334)	(0.00666)	(0.0466)	(0.00227)	(0.0106)	(0.00446)
lnk	0.425***	0.479***	0.433***	0.268***	0.426***	0.452***
	(0.0219)	(0.0119)	(0.0314)	(0.00804)	(0.0157)	(0.0130)
time	33.21	6.599	27.80	6.438	37.86	8.566

Panel (c): Meas. Err. 0.2

	ACF	LP	ACF	LP	ACF	LP
lnl	0.660***	0.675***	0.653***	0.801***	0.584***	0.637***
	(0.0391)	(0.00606)	(0.0422)	(0.00240)	(0.0138)	(0.00485)
lnk	0.378***	0.337***	0.379***	0.211***	0.427***	0.380***
	(0.0291)	(0.0102)	(0.0342)	(0.00634)	(0.0134)	(0.0113)
time	33.81	6.211	24.89	9.686	43.48	11.68

Panel (d): Meas. Err. 0.5

	(1)	(2)	(3)	(4)	(5)	(6)
	ACF	LP	ACF	LP	ACF	LP
lnl	0.626***	0.803***	0.660***	0.856***	0.598***	0.713***
	(0.0263)	(0.00468)	(0.0234)	(0.00191)	(0.0234)	(0.00452)
lnk	0.388***	0.203***	0.357***	0.157***	0.414***	0.306***
	(0.0177)	(0.00871)	(0.0183)	(0.00610)	(0.0164)	(0.0107)
time	23.97	6.225	17.09	9.362	36.13	11.15
N	10000	10000	10000	10000	10000	10000

Note: models have been estimated through `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`
DGP1 - Serially Correlated Wages and Labor Set at Time $t - b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t - b$ (DGP1 plus DGP2)

Table A.2: ACF (nm) - various starting points - DGP3

Panel (a): Meas. Err. 0

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
lnl	0.814***	0.830***	0.804***	0.595***	0.596***	0.600***	0.595***	0.974***
	(0.234)	(0.0892)	(0.00728)	(0.00617)	(0.00618)	(0.0000167)	(0.129)	(0.00458)
lnk	2.241***	2.701***	1.876***	0.369***	0.386***	0.400***	0.373***	0.114***
	(0.440)	(0.269)	(0.0170)	(0.00735)	(0.00737)	(0.0000167)	(0.0620)	(0.000868)
time	20.46	20.42	11.31	19.25	17.70	4.591	20.97	17.72

Panel (b): Meas. Err. 0.1

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
lnl	0.100	0.200	0.861***	0.400	0.500	0.600***	0.849***	0.931***
	(.)	(.)	(0.00741)	(.)	(.)	(0.0000170)	(0.00238)	(0.00306)
lnk	0.900	0.800	2.010***	0.600	0.500	0.400***	0.461***	0.106***
	(.)	(.)	(0.0173)	(.)	(.)	(0.0000166)	(0.0126)	(0.000592)
time	4.719	4.586	10.90	4.598	4.601	4.586	18.72	16.00

Panel (c): Meas. Err. 0.2

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
lnl	0.100	0.200	0.884***	0.400	0.500	0.600***	0.859***	0.915***
	(.)	(.)	(0.00698)	(.)	(.)	(0.0000170)	(0.0256)	(0.00355)
lnk	0.900	0.800	2.062***	0.600	0.500	0.400***	0.520***	0.103***
	(.)	(.)	(0.0163)	(.)	(.)	(0.0000164)	(0.138)	(0.000645)
time	4.609	4.747	15.12	5.331	4.512	4.501	12.69	14.14

Panel (d): Meas. Err. 0.5

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
lnl	0.100	0.200	0.906***	0.400	0.500	0.600	0.800***	0.900***
	(.)	(.)	(0.00668)	(.)	(.)	(.)	(0.0000106)	(0.0000175)
lnk	0.900	0.800	2.114***	0.600***	0.500***	0.400***	0.200	0.100
	(.)	(.)	(0.0156)	(0.00000839)	(0.00000840)	(0.0000174)	(.)	(.)
time	4.510	4.548	11.42	4.706	4.529	4.535	4.611	4.527

Note: in column (1) to (8) we report ACF estimates with different optimization starting points from [0.1,(1-0.1)] to [0.8,(1-0.8)], respectively.

Table A.3: ACF (bfgs) - various starting points - DGP3

Panel (a): Meas. Err. 0

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.429***	0.616***	0.591***	0.591***	0.607***	0.588***	0.590***	1.011***
	(0.0989)	(0.134)	(0.0269)	(0.0226)	(0.00778)	(0.00509)	(0.0599)	(0.00401)
lnk	0.536	0.499***	0.425***	0.424***	0.483***	0.402***	0.424***	0.000360
	(0.371)	(0.140)	(0.0658)	(0.0461)	(0.0228)	(0.000644)	(0.136)	(0.00677)
time	5.067	4.465	4.371	4.041	2.438	2.035	4.338	4.008

Panel (b): Meas. Err. 0.1

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.590	0.593***	0.593***	0.591***	0.594***	0.584***	0.858***	0.952***
	(0.618)	(0.0688)	(0.0325)	(0.00912)	(0.0116)	(0.00926)	(0.110)	(0.00306)
lnk	0.427	0.440**	0.439***	0.443***	0.440***	0.406***	1.891***	0.0654***
	(0.738)	(0.214)	(0.159)	(0.0185)	(0.0148)	(0.0181)	(0.406)	(0.00636)
time	5.277	4.787	4.107	3.568	2.814	2.335	5.625	3.518

Panel (c): Meas. Err. 0.2

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.597***	0.597**	0.596***	0.591***	0.597***	0.596***	0.865***	0.922***
	(0.0327)	(0.269)	(0.0453)	(0.0436)	(0.0177)	(0.0120)	(0.116)	(0.00419)
lnk	0.444***	0.443	0.442**	0.437**	0.444***	0.444***	0.336	0.108***
	(0.100)	(0.367)	(0.189)	(0.185)	(0.0149)	(0.0200)	(0.351)	(0.0126)
time	5.608	4.769	4.843	3.578	3.250	2.786	4.315	2.543

Panel (d): Meas. Err. 0.5

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	2.772***	0.597***	0.596***	0.598***	0.597***	0.572***	0.832***	0.906***
	(0.716)	(0.0530)	(0.0366)	(0.00870)	(0.0224)	(0.0117)	(0.130)	(0.00473)
lnk	-1.894**	0.445***	0.444***	0.444***	0.445***	0.447***	0.295*	0.103***
	(0.757)	(0.105)	(0.0999)	(0.0102)	(0.0253)	(0.0100)	(0.169)	(0.0112)
time	4.730	5.465	4.652	4.757	3.320	2.754	3.806	2.009

Note: in column (1) to (8) we report ACF estimates with different optimization starting points from [0.1,(1-0.1)] to [0.8,(1-0.8)], respectively.

Table A.4: ACF (dfp) - various starting points - DGP3

Panel (a): Meas. Err. 0

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.566***	1.039***	0.634***	0.613***	0.607***	0.598***	0.771***	1.002***
	(0.0937)	(0.0622)	(0.0155)	(0.0192)	(0.00945)	(0.00706)	(0.0810)	(0.00801)
lnk	-0.232	-0.0935	0.499***	0.481***	0.454***	0.400***	0.942	0.000845
	(0.341)	(0.180)	(0.0405)	(0.0490)	(0.0207)	(0.00214)	(0.814)	(0.0170)
time	27.46	19.23	19.77	28.44	17.99	10.20	121.8	55.95

Panel (b): Meas. Err. 0.1

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.586***	0.601***	0.709***	0.602***	0.609***	0.596***	0.887***	0.931***
	(0.219)	(0.107)	(0.0733)	(0.0741)	(0.0106)	(0.00988)	(0.0886)	(0.00820)
lnk	0.386	0.418*	-0.157	0.418***	0.436***	0.402***	0.253	0.104***
	(0.336)	(0.230)	(0.216)	(0.122)	(0.0160)	(0.00401)	(0.161)	(0.0202)
time	67.36	59.37	48.03	30.33	25.09	17.72	141.2	42.42

Panel (c): Meas. Err. 0.2

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.785***	0.572***	0.630	0.708***	0.613***	0.613***	0.866***	0.915***
	(0.0856)	(0.113)	(0.493)	(0.0604)	(0.0124)	(0.0107)	(0.0153)	(0.00322)
lnk	0.451***	0.268	0.240	0.0225	0.420***	0.411***	0.297	0.105***
	(0.168)	(0.203)	(0.561)	(0.165)	(0.0225)	(0.0119)	(0.204)	(0.00356)
time	85.00	77.39	57.00	40.90	21.99	15.33	24.41	15.49

Panel (d): Meas. Err. 0.5

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.612*	0.632***	0.612***	0.633***	0.636***	0.631***	0.837***	0.900***
	(0.317)	(0.152)	(0.102)	(0.0662)	(0.0232)	(0.00954)	(0.0390)	(0.00404)
lnk	0.435	0.397**	0.399***	0.397***	0.397***	0.401***	0.274***	0.100***
	(0.350)	(0.188)	(0.125)	(0.123)	(0.0671)	(0.0154)	(0.0619)	(0.00657)
time	85.20	88.25	71.45	80.25	52.01	16.64	20.20	14.42

Note: in column (1) to (8) we report ACF estimates with different optimization starting points from [0.1,(1-0.1)] to [0.8,(1-0.8)], respectively.

Table A.5: ACF (bhhh) - various starting points - DGP3

Panel (a): Meas. Err. 0

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.979***	0.621***	0.588***	0.634***	0.590***	0.590***	0.593***	0.980***
	(0.198)	(0.0290)	(0.0441)	(0.0773)	(0.0552)	(0.00409)	(0.0136)	(0.0221)
lnk	0.168	0.0769	0.414	-0.0165	0.421	0.419***	0.258*	0.192***
	(0.160)	(0.298)	(0.813)	(0.239)	(0.460)	(0.0140)	(0.146)	(0.0306)
time	466.5	853.2	1895.1	918.6	1343.5	282.6	587.3	300.9

Panel (b): Meas. Err. 0.1

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.204	0.923***	0.597***	0.596***	0.591***	0.589***	0.940***	0.937***
	(0.196)	(0.251)	(0.132)	(0.0767)	(0.0348)	(0.00728)	(0.137)	(0.00881)
lnk	0.812***	0.277*	0.446	0.442**	0.437***	0.430***	0.103	0.118***
	(0.171)	(0.168)	(0.757)	(0.191)	(0.0596)	(0.0183)	(0.0696)	(0.0373)
time	295.7	649.5	1367.6	535.6	292.9	282.4	284.7	281.6

Panel (c): Meas. Err. 0.2

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.0917	0.874***	0.568***	0.564***	0.595***	0.587***	0.852***	0.913***
	(0.272)	(0.263)	(0.128)	(0.0737)	(0.0382)	(0.0109)	(0.134)	(0.00635)
lnk	0.893***	1.553***	0.433***	0.433***	0.443***	0.423***	0.965***	0.134***
	(0.279)	(0.209)	(0.120)	(0.113)	(0.0284)	(0.0205)	(0.119)	(0.0220)
time	285.9	478.6	274.4	293.8	282.6	274.0	352.1	281.6

Panel (d): Meas. Err. 0.5

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
	lny	lny	lny	lny	lny	lny	lny	lny
lnl	0.906***	0.248	0.326**	0.592***	0.509***	0.601***	0.921***	0.904***
	(0.137)	(0.172)	(0.141)	(0.0851)	(0.0277)	(0.0221)	(0.120)	(0.00167)
lnk	0.366***	0.746***	0.676***	0.446***	0.488***	0.442***	0.313**	0.109***
	(0.116)	(0.125)	(0.115)	(0.0629)	(0.0248)	(0.0190)	(0.122)	(0.00741)
time	445.7	285.0	278.4	280.1	280.9	272.8	282.4	423.7

Note: in column (1) to (8) we report ACF estimates with different optimization starting points from [0.1,(1-0.1)] to [0.8,(1-0.8)], respectively.

Table A.6: ACF and LP - Newton-Raphson with $\rho = .6$

	ACF				LP			
	β_l		β_k		β_l		β_k	
Meas. Error	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.
DGP1 - Serially Correlated Wages and Labor Set at Time $t - b$								
0.0	0.598	0.012	0.396	0.021	-0.000	0.005	1.048	0.025
0.1	0.611	0.031	0.417	0.018	0.608	0.009	0.420	0.012
0.2	0.637	0.031	0.396	0.021	0.737	0.008	0.283	0.010
0.5	0.668	0.017	0.358	0.014	0.845	0.006	0.168	0.008
DGP2 - Optimization Error in Labor								
0.0	0.603	0.015	0.404	0.017	0.600	0.003	0.400	0.011
0.1	0.608	0.019	0.397	0.020	0.750	0.004	0.254	0.007
0.2	0.633	0.024	0.384	0.018	0.805	0.004	0.200	0.007
0.5	0.657	0.018	0.361	0.015	0.862	0.003	0.145	0.025
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t - b$ (DGP1 plus DGP2)								
0.0	0.599	0.014	0.422	0.029	0.452	0.004	0.573	0.015
0.1	0.597	0.012	0.420	0.014	0.603	0.005	0.424	0.011
0.2	0.598	0.011	0.419	0.016	0.671	0.005	0.355	0.011
0.5	0.620	0.019	0.407	0.016	0.752	0.005	0.271	0.032

Note: models have been estimated through `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`
DGP1 - Serially Correlated Wages and Labor Set at Time $t - b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t - b$ (DGP1 plus DGP2)

Table A.7: ACF and LP - DFP with $\rho = .6$

	ACF				LP			
	β_l		β_k		β_l		β_k	
Meas. Error	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$								
0.0	0.603	0.012	0.402	0.015	-0.000	0.005	1.047	0.025
0.1	0.638	0.014	0.409	0.015	0.607	0.010	0.420	0.012
0.2	0.654	0.005	0.388	0.011	0.737	0.008	0.283	0.010
0.5	0.658	0.003	0.366	0.008	0.845	0.006	0.168	0.009
DGP2 - Optimization Error in Labor								
0.0	0.593	0.013	0.379	0.008	0.600	0.003	0.400	0.011
0.1	0.619	0.017	0.398	0.017	0.751	0.004	0.253	0.007
0.2	0.647	0.006	0.380	0.013	0.805	0.004	0.200	0.007
0.5	0.656	0.004	0.362	0.008	0.862	0.003	0.145	0.024
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)								
0.0	0.585	0.008	0.373	0.008	0.452	0.004	0.573	0.015
0.1	0.587	0.011	0.403	0.017	0.604	0.005	0.424	0.011
0.2	0.594	0.013	0.416	0.014	0.672	0.005	0.355	0.011
0.5	0.628	0.016	0.406	0.016	0.753	0.005	0.276	0.073

Note: models have been estimated through `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)

Table A.8: ACF and LP - DFP with $\rho = .7$

	ACF				LP			
	β_l		β_k		β_l		β_k	
Meas. Error	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$								
0.0	0.594	0.010	0.373	0.005	-0.000	0.005	1.087	0.029
0.1	0.590	0.022	0.422	0.013	0.676	0.009	0.363	0.012
0.2	0.634	0.016	0.400	0.017	0.788	0.007	0.240	0.010
0.5	0.656	0.003	0.364	0.009	0.875	0.005	0.260	0.239
DGP2 - Optimization Error in Labor								
0.0	0.596	0.009	0.362	0.004	0.600	0.003	0.400	0.012
0.1	0.598	0.013	0.378	0.011	0.753	0.004	0.256	0.008
0.2	0.606	0.019	0.393	0.015	0.807	0.004	0.203	0.010
0.5	0.649	0.012	0.377	0.016	0.863	0.003	0.370	0.320
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)								
0.0	0.589	0.006	0.360	0.002	0.473	0.003	0.571	0.016
0.1	0.588	0.007	0.370	0.010	0.634	0.005	0.413	0.012
0.2	0.589	0.010	0.380	0.015	0.701	0.005	0.344	0.012
0.5	0.598	0.015	0.393	0.016	0.777	0.005	1.444	0.123

Note: models have been estimated through `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)

Table A.9: ACF and LP - BFGS with $\rho = .6$

	ACF				LP			
	β_l		β_k		β_l		β_k	
Meas. Error	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$								
0.0	0.603	0.012	0.401	0.015	-0.000	0.005	1.047	0.025
0.1	0.631	0.022	0.410	0.015	0.607	0.009	0.420	0.012
0.2	0.652	0.011	0.388	0.012	0.737	0.008	0.283	0.010
0.5	0.658	0.004	0.366	0.008	0.845	0.006	0.168	0.008
DGP2 - Optimization Error in Labor								
0.0	0.594	0.013	0.379	0.010	0.600	0.003	0.399	0.011
0.1	0.619	0.016	0.398	0.017	0.751	0.004	0.253	0.007
0.2	0.647	0.007	0.380	0.013	0.805	0.004	0.200	0.007
0.5	0.655	0.004	0.362	0.008	0.862	0.003	0.145	0.029
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)								
0.0	0.586	0.007	0.374	0.011	0.452	0.004	0.572	0.014
0.1	0.589	0.009	0.405	0.019	0.604	0.005	0.424	0.010
0.2	0.596	0.010	0.417	0.015	0.672	0.005	0.355	0.010
0.5	0.627	0.016	0.406	0.015	0.753	0.005	0.275	0.072

Note: models have been estimated through `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)

Table A.10: ACF and LP - BFGS with $\rho = .7$

	ACF				LP			
	β_l		β_k		β_l		β_k	
Meas. Error	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.
	DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$							
0.0	0.594	0.010	0.373	0.006	0.000	0.005	1.088	0.029
0.1	0.591	0.020	0.422	0.014	0.677	0.009	0.363	0.012
0.2	0.633	0.017	0.401	0.017	0.788	0.007	0.240	0.010
0.5	0.656	0.003	0.365	0.010	0.875	0.005	0.258	0.237
	DGP2 - Optimization Error in Labor							
0.0	0.596	0.010	0.362	0.005	0.600	0.003	0.400	0.013
0.1	0.599	0.013	0.378	0.012	0.753	0.004	0.256	0.008
0.2	0.606	0.019	0.393	0.015	0.807	0.004	0.203	0.010
0.5	0.650	0.011	0.376	0.015	0.863	0.003	0.374	0.321
	DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)							
0.0	0.590	0.006	0.360	0.002	0.473	0.004	0.570	0.016
0.1	0.589	0.008	0.374	0.019	0.634	0.005	0.412	0.012
0.2	0.594	0.011	0.393	0.028	0.701	0.005	0.344	0.033
0.5	0.603	0.015	0.398	0.019	0.777	0.005	1.449	0.127

Note: models have been estimated through `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)

Table A.11: ACF and LP - BHHH with $\rho = .7$

	ACF				LP			
	β_l		β_k		β_l		β_k	
Meas. Error	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.	Coeff.	St. Dev.
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$								
0.0	0.611	0.009	0.414	0.018	0.003	0.004	1.099	0.021
0.1	0.579	0.023	0.424	0.023	0.683	0.011	0.360	0.015
0.2	0.643	0.018	0.399	0.011	0.792	0.008	0.238	0.009
0.5	0.669	0.007	0.360	0.004	0.877	0.007	0.142	0.009
DGP2 - Optimization Error in Labor								
0.0	0.607	0.014	0.416	0.025	0.600	0.003	0.392	0.015
0.1	0.608	0.013	0.407	0.016	0.755	0.004	0.248	0.009
0.2	0.602	0.019	0.403	0.017	0.809	0.003	0.197	0.009
0.5	0.642	0.025	0.376	0.026	0.866	0.003	0.713	0.066
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)								
0.0	0.617	0.014	0.476	0.033	0.475	0.004	0.571	0.019
0.1	0.612	0.006	0.443	0.011	0.634	0.005	0.415	0.013
0.2	0.606	0.008	0.423	0.019	0.702	0.004	0.344	0.009
0.5	0.610	0.017	0.413	0.016	0.775	0.006	1.462	0.048

Note: models have been estimated through `prodest lnva, free(lnl) proxy(lnm) state(lnk) poly(3) met(lp) valueadded reps(50) [acf]`
DGP1 - Serially Correlated Wages and Labor Set at Time $t-b$
DGP2 - Optimization Error in Labor
DGP3 - Optimization Error in Labor and Serially Correlated Wages and Labor Set at Time $t-b$ (DGP1 plus DGP2)

Table A.12: ACF (nr) - Monte Carlo - DGP2

Panel (a): Meas. Err. 0

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
β_l	0.656	0.636	0.641	0.609	0.630	0.601	0.636	0.644	0.981
	(0.086)	(0.079)	(0.107)	(0.014)	(0.096)	(0.008)	(0.051)	(0.019)	(0.065)
β_k	0.384	0.336	0.345	0.416	0.387	0.401	0.433	0.175	0.261
	(0.189)	(0.164)	(0.140)	(0.021)	(0.095)	(0.018)	(0.031)	(0.076)	(0.622)

Panel (b): Meas. Err. 0.1

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
β_l	0.628	0.620	0.615	0.611	0.628	0.607	0.613	0.695	0.957
	(0.111)	(0.084)	(0.071)	(0.041)	(0.086)	(0.012)	(0.018)	(0.031)	(0.032)
β_k	0.377	0.378	0.382	0.376	0.384	0.403	0.382	0.149	0.148
	(0.124)	(0.110)	(0.104)	(0.081)	(0.086)	(0.017)	(0.049)	(0.074)	(0.134)

Panel (c): Meas. Err. 0.2

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
β_l	0.553	0.571	0.583	0.590	0.624	0.613	0.620	0.715	0.943
	(0.189)	(0.140)	(0.103)	(0.064)	(0.062)	(0.014)	(0.015)	(0.027)	(0.016)
β_k	0.442	0.424	0.421	0.412	0.389	0.400	0.356	0.155	0.139
	(0.181)	(0.142)	(0.121)	(0.064)	(0.060)	(0.018)	(0.050)	(0.059)	(0.084)

Panel (d): Meas. Err. 0.5

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
β_l	0.307	0.382	0.447	0.521	0.610	0.628	0.675	0.731	0.927
	(0.239)	(0.196)	(0.145)	(0.099)	(0.057)	(0.020)	(0.042)	(0.034)	(0.022)
β_k	0.674	0.603	0.544	0.477	0.404	0.389	0.344	0.181	0.128
	(0.224)	(0.183)	(0.132)	(0.086)	(0.046)	(0.022)	(0.040)	(0.055)	(0.093)

Note: in column (1) to (8) we report ACF estimates with different optimization starting points from [0.1,(1-0.1)] to [0.8,(1-0.8)], respectively.

Table A.13: ACF (dfp) - Monte Carlo - DGP3

Panel (a): Meas. Err. 0

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
β_l	0.595	0.626	0.622	0.620	0.607	0.592	0.589	0.739	1.006
	(0.116)	(0.101)	(0.024)	(0.019)	(0.009)	(0.006)	(0.005)	(0.088)	(0.007)
β_k	0.108	0.401	0.487	0.487	0.466	0.401	0.325	0.942	-0.00786
	(0.324)	(0.158)	(0.053)	(0.049)	(0.022)	(0.001)	(0.006)	(0.739)	(0.013)

Panel (b): Meas. Err. 0.1

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
β_l	0.645	0.657	0.642	0.610	0.613	0.591	0.590	0.904	0.943
	(0.099)	(0.226)	(0.210)	(0.042)	(0.010)	(0.009)	(0.008)	(0.112)	(0.009)
β_k	0.315	0.241	0.226	0.425	0.447	0.405	0.343	0.234	0.0694
	(0.224)	(0.322)	(0.293)	(0.059)	(0.015)	(0.004)	(0.021)	(0.303)	(0.021)

Panel (c): Meas. Err. 0.2

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
β_l	0.651	0.676	0.672	0.623	0.607	0.593	0.593	0.868	0.917
	(0.121)	(0.274)	(0.256)	(0.053)	(0.013)	(0.012)	(0.009)	(0.038)	(0.004)
β_k	0.319	0.281	0.241	0.320	0.428	0.412	0.347	0.304	0.102
	(0.186)	(0.325)	(0.319)	(0.162)	(0.026)	(0.009)	(0.021)	(0.160)	(0.007)

Panel (d): Meas. Err. 0.5

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
β_l	0.610	0.660	0.656	0.632	0.608	0.606	0.605	0.841	0.901
	(0.308)	(0.215)	(0.097)	(0.050)	(0.023)	(0.013)	(0.015)	(0.035)	(0.003)
β_k	0.373	0.324	0.321	0.351	0.413	0.419	0.353	0.305	0.101
	(0.327)	(0.241)	(0.141)	(0.114)	(0.028)	(0.013)	(0.029)	(0.088)	(0.003)

Note: in column (1) to (8) we report ACF estimates with different optimization starting points from [0.1,(1-0.1)] to [0.8,(1-0.8)], respectively.